ZCP 7.2 (build 48988)

Zarafa Collaboration Platform

Manual do Administrador

ZCP 7.2 (build 48988) Zarafa Collaboration Platform
Manual do Administrador
Edição 7.2

O Zarafa Collaboration Platform (ZCP) combina a usabilidade do Outlook com a estabilidade e flexibilidade de um servidor Linux. Além de dispor de uma rica interface web, o WebAccess Zarafa, o ZCP também oferece brilhantes opções de integração com todos os tipos de clientes, incluindo todas as plataformas móveis mais populares.

Most components of ZCP are open source, licensed under the *AGPLv3*[1], can therefore be downloaded freely as *ZCP's Community Edition*[2].

Existem alguns componentes de código-fechado, dentre eles os mais relevantes são:

[4] http://creativecommons.org/licenses/by-sa/3.0/
[1] http://www.gnu.org/licenses/agpl-3.0.html
[2] http://community.zarafa.com

- o Zarafa Windows Cliente proporcionando integração com o Outlook,
- o Zarafa BES proporcionando integração com o servidor Blackberry Enterprise,
- o Plugin Zarafa ADS proporcionando integração como o Active Directory, e
- as ferramentas de Backup do Zarafa.

Estes componentes, bem como vários recursos avançados para grandes configurações e serviços de hospedagem, somente estão disponíveis em combinação com um contrato de suporte como parte de *edições comerciais do Zarafa*[3].

Como alternativa, há disponível uma vasta seleção de ofertas de hospedagem no Zarafa.

O Manual do Administrador, descreve como instalar, atualizar, configurar e manter o Zarafa em seu servidor Linux. Além disso, são discutidas várias configurações avançadas e opções de integração.

[3] http://www.zarafa.com/content/editions

1. Introdução **1**
1.1. Público Alvo ... 1
1.2. Arquitetura ... 1
1.3. Componetes ... 3
1.4. Protocolos e conexões ... 4
1.4.1. SOAP ... 4
1.4.2. HTTP seguro (HTTPS) ... 4
1.5. Versões e licenciamento das Versões do Zarafa ... 4
1.5.1. A subscrição de avaliação ... 4
1.5.2. A Versão Community do Zarafa ... 5
1.5.3. Edições comerciais do Zarafa ... 5
1.5.4. Usuários ativos e não-ativos ... 5

2. Instalando **7**
2.1. Requisitos de Sistema ... 7
2.1.1. Recomendções de Hardware ... 7
2.1.2. Connection/bandwidth Recommendation ... 8
2.1.3. Plataformas suportadas ... 8
2.1.4. Dependências ... 9
2.2. Instalação ... 11
2.2.1. Instalar usando o script de instalação ... 11
2.2.2. Instalando os pacotes manualmente ... 11
2.3. Solucionando problemas de instalação ... 16
2.3.1. Processos do servidor ... 16
2.3.2. WebAccess & WebApp ... 16
2.4. Removing Zarafa ... 17

3. Atualizando a versão **19**
3.1. Preparação ... 19
3.2. Criando backups ... 20
3.3. Dependências da versão 7 do Zarafa ... 20
3.4. Realizando a atualização em RPM baseada em distribuições ... 21
3.5. Realizando a atualização em Debian baseado em distribuições ... 21
3.5.1. Etapas de pré atualização da versão 6.40 do Zarafa ... 22
3.5.2. Da versão 6.40 para 7.0.0 e posterior ... 23
3.5.3. From 7.0 to 7.1.0 and higher ... 25
3.6. Finalizando a atualização ... 26

4. Configurar Componentes ZCP **29**
4.1. Configurando o Servidor Zarafa ... 29
4.2. Configurar a linguagem em distribuições baseadas em RPM ... 30
4.3. Configurar a linguagem em distribuições baseadas em Debian ... 30
4.4. Autenticação de Usuário ... 31
4.4.1. Autenticação via plugin DB ... 32
4.4.2. Autenticação via plugin Unix ... 32
4.4.3. Autenticação via plugin LDAP ... 33
4.5. Auto-respondedor ... 33
4.6. Armazenando anexos fora do banco de dados ... 35
4.7. Conexões SSL e certificados ... 36
4.8. Configurar o Gerenciador de Licensa ... 37
4.9. Configurando o Spooler do Zarafa ... 38
4.9.1. Configuração ... 38
4.10. Configurar o Zarafa Caldav ... 39
4.10.1. SSL/TLS ... 40
4.11. Configurando o Zarafa Gateway (IMAP and POP3) ... 41

4.11.1. SSL/TLS 43
4.11.2. Informações importantes 43
4.12. Configurando o Gerenciador de Quota do Zarafa 43
4.12.1. Configurando a quota de todo do servidor 43
4.12.2. Definir a quota por usuário 44
4.12.3. Monitorando quando a quota excede 44
4.12.4. Modelos de alerta de quota 44
4.13. Configure Zarafa Search 45
4.13.1. Enabling the search service 45
4.13.2. Search configuration 45
4.13.3. Anexos 46
4.14. Configure Zarafa WebAccess 47
4.15. Configure Zarafa WebApp 47

5. Configurar componentes de terceiros 49
5.1. Configurar o servidor web 49
5.1.1. Configurar PHP 49
5.1.2. Configurar o Apache 49
5.1.3. Apache como um proxy HTTP 51
5.2. Configurar a integração do ZCP OpenLDAP 52
5.2.1. Configuring OpenLDAP to use the Zarafa schema 52
5.2.2. LDAP indices 52
5.2.3. Configurando o ZCP para OpenLDAP 53
5.2.4. Configuração do usuário 54
5.2.5. Configuração de grupos 55
5.2.6. Configuração de lista de endereços 56
5.2.7. Testando a configuração do LDAP 56
5.3. Configurar a integração do Active Directory do ZCP 57
5.3.1. Instalando o Plugin Zarafa ADS e arquivos de esquema 57
5.3.2. Configurando o ZCP para ADS 59
5.3.3. Configuração do usuário 60
5.3.4. Configuração de grupos 61
5.3.5. Configuração de lista de endereços 61
5.3.6. Testando a configuração do Active Directory 62
5.4. Integração do ZCP Postfix 62
5.4.1. Configurar a integração do ZCP Postfix com OpenLDAP 63
5.4.2. Configurar integração do Postfix do ZCP com o Active Directory 64
5.4.3. Configurar a integração do Postfix do ZCP com usuários virtuais 66
5.4.4. Configure ZCP Postfix integration with the DB plugin 67
5.5. Configurar Z-Push (ActiveSync remoto para dispositivos móveis) 68
5.5.1. Compatibilidade 69
5.5.2. Segurança 69
5.5.3. Instalação 69
5.5.4. Gerenciamento de dispositivo móvel 71
5.5.5. Atualização 71
5.5.6. S/MIME 72
5.6. Configuring SSL for Windows Mobile and Windows Phone 73
5.7. Solução de problemas 74

6. Configurações avançadas 77
6.1. Rodando componentes do ZCP além da hospedagem local 77
6.2. Configurações de Multi-locação 78
6.2.1. Plugins de usuário suportados 78
6.2.2. Configurando o servidor 78
6.2.3. Admininstrando espaços do locatário (companhia) 81

6.2.4. Administrando usuários e grupos 82
6.2.5. Níveis de quota 82
6.2.6. Usuários Administradores 83
6.3. Configuração de Multi-servidor 84
6.3.1. Introdução 84
6.3.2. Preparando / definindo o servidor LDAP para configuração multi-servidor 86
6.3.3. Configurando os servidores 87
6.3.4. Criando certificados SSL 88
6.4. Atualizador do Zarafa Windows Client 90
6.4.1. Configuração do servidor de apoio 91
6.4.2. Configuração do servidor de apoio 92
6.4.3. Opções MSI 94
6.5. Armazenagem de anexos em caso singular 95
6.5.1. Armazenagem de anexos em caso singular e LMTP 95
6.6. Rodando o ZPC Services com privilégios de usuário regulares 95
6.7. Single Sign On com o ZCP 96
6.7.1. NTLM SSO com ADS 96
6.7.2. NTLM SSO with Samba 3 99
6.7.3. SSO com Kerberos 99
6.7.4. Funcionando 104
6.8. Rastreando mensagens com o Zarava Archive 104
6.8.1. Arquivo na entrega 104
6.8.2. Arquivar ao enviar 105
6.9. Zarafa Python plugin framework 105
6.9.1. How it works 105
6.9.2. General Options 105
6.9.3. How to use 106
6.9.4. Zarafa-DAgent plugins 106
6.9.5. Zarafa-Spooler plugins 107
6.9.6. Troubleshooting 107
6.10. Running ZCP multi-server behind Reverse Proxy 108
6.10.1. Description of redirection problem 109
6.10.2. Setup Prerequisites 110
6.10.3. Example Setup with Apache 110

7. Gerenciando os serviços do Zarafa **113**
7.1. Iniciando os serviços 113
7.1.1. Interrompendo os serviços 113
7.1.2. Recarregando configurações de serviços 114
7.2. Opções de acesso 114
7.3. Registro de segurança 115
7.3.1. Itens de registro 115
7.3.2. Configuração 118
7.4. Monitoramento estatístico do Zarafa 118
7.5. Sistema de exclusão 119

8. Gerenciamento dos Usuários **121**
8.1. Pasta Pública 121
8.2. Uso geral da ferramenta Zarafa-admin 121
8.3. Gerenciamento de usuários com o plugin DB 123
8.3.1. Criando usuários com o plugin DB 123
8.3.2. Usuários não-ativos 124
8.3.3. Atualizando informação do usuário com o plugin DB 124
8.3.4. Deletando usuários com o plugin DB 124
8.3.5. Configurando permissões de "Send as" 124

8.3.6. Grupos 125
8.4. Gerenciamento de usuários com o plugin UNIX 126
8.4.1. Criando usuários com o plugin Unix 126
8.4.2. Usuários não-ativos 126
8.4.3. Atualizando informação do usuário com o plugin Unix 127
8.4.4. Deletando usuários com o plugin Unix 127
8.4.5. Configurando permissões de "Send as" 127
8.4.6. Grupos com o plugin Unix 128
8.5. Gestão do Usuário com o LDAP ou Active Directory 129
8.5.1. O princípio da sincronização do usuário do Zarafa 129
8.5.2. Manuseio de usuário à partir do ADS 132
8.5.3. Manuseio de usuário à partir do OpenLDAP 136
8.6. Exemplos de Condição do LDAP 138
8.7. Manuseio do Zarafa Feature 139
8.7.1. Habilitando características globalmente 139
8.7.2. Habilitando ou desabilitando características por usuário 139
8.8. Configuração de recurso 141
8.8.1. Metodos de reserva de recurso 142
8.8.2. Reserva de requisição de reunião (MR) 143
8.8.3. Definindo o método de reserva de recurso 144
8.9. Out of office management 144
8.10. Realocador de estoque da caixa de correio 145
8.10.1. Pré-requisitos 145
8.10.2. Invocação 145
8.10.3. Atualizando o LDAP/ADS 146
8.10.4. Configuração 146
8.10.5. Post migration steps 147

9. Ajuste de desempenho **149**
9.1. Considerações sobre hardwares 149
9.1.1. Uso da memória 149
9.1.2. Considerações sobre Hardwares 150
9.1.3. Mais memório significa mais velocidade 150
9.1.4. RAID 1/10 é mais rápdo que RAID 5 150
9.1.5. Alta velocidade de rotação (RPM) para um melhor desempenho do banco de dados 150
9.1.6. Hardware RAID 150
9.2. Configuração do uso de memória 150
9.2.1. Cache do Zarafa em relação a telefones celulares (**cache_cell_size**) 151
9.2.2. Cache dos objetos do Zarafa (**cache_object_size**) 151
9.2.3. Zarafa indexedobject cache (**cache_indexedobject_size**) 152
9.2.4. MySQL **innodb_buffer_pool_size** 152
9.2.5. MySQL **innodb_log_file_size** 152
9.2.6. MySQL **innodb_log_buffer_size** 152
9.2.7. MySQL **query_cache_size** 152
9.2.8. MySQL **innodb_buffer_pool_size** 152
9.2.9. MySQL **max_allowed_packet** 153
9.3. Configuração de módulos em diferentes servidores 153

10. Backup & Restauração **155**
10.1. Softdelete restore 155
10.2. Despejo de memória do banco de dados completo 155
10.2.1. Despejo do SQL através do mysqldump 156
10.2.2. Despejo de dados binários via LVM Snapshotting 156
10.2.3. Backup de anexos 156

10.3. Backups Brick-level 157
10.3.1. Formatar Backup 157
10.3.2. Processo de Backup 157
10.3.3. Processo de restauração 158

11. Apêndice A; estratégias de atualização pré-5.2x **161**
11.1. Atualizações do banco de dados à partir de 4.1 ou 4.2 161
11.2. Atualização da versão 5.0 para as versões 5.1x e posteriores 162
11.3. Mudanças importantes a partir das versões 4.x e 5.x 162

12. Apêndice B; descrição dos atributos do LDAP **163**

13. Appendix C: Example LDIF **171**

Introdução

Zarafa Collaboration Platform (ZCP) is an open source software suite capable of replacing Microsoft Exchange. It's architecture is very modular, makes use of standards wherever possible, and integrates with common open source components.

Este documento explica como executar as tarefas administrativas mais comuns com a solução Zarafa.

Importante

Embora nós, da Zarafa, façamos o nosso melhor para manter as informações contidas neste manual tão precisas quanto possível, nos reservamos o direito de modificar estas informações a qualquer momento, sem aviso prévio.

1.1. Público Alvo

Este manual é recomendado para administradores de sistema responsáveis pela instalação, manutenção e o suporte do sistema ZCP. Assumimos que o leitor deste manual tenha conhecimentos profundos em:

- Conceitos e tarefas da administração de sistemas Linux
- Padrões de comunicação de e-mail
- Conceitos de segurança
- Serviços de diretório
- Gerenciamento de banco de dados

1.2. Arquitetura

Em conformidade com a filosofia UNIX, a Plataforma de Colaboração Zarafa é constituída por componentes com tarefas bem definidas. Veja a*Figura 1.1, "Diagrama da arquitetura do Pacote de Colaboração Zarafa."* que descreve o relacionamento entre componenes e os protocolos usados. Este diagrama descreve uma configuração simples, usada pela maioria dos nossos clientes. Apenas os componentes mais comumente usados são mostrados.

A parte superior do diagrama mostra os clientes: aparelhos e respectivos softwares pelos quais os usuários acessam seus dados. Alguns são aplicativos de desktop, outros por sua vez são aplicativos móveis.

Entre "A Internet" e "o Servidor Zarafa" encontram-se os componentes de infra-estrutura do Zarafa (azul) e alguns dos componentes de infra-estrutura mais comuns (cinza) Estes componentes são necessários para facilitar a comunicação entre o Servidor Zarafa e vários clientes. O Microsoft Outlook não precisa de nenhuma infra-estrutura especial, pois se comunica diretamente com o servidor Zarafa usando o Zarafa Windows Client.

O servidor Zarafa atua basicamente como um servidor de chamadas MAPI, armazena dados em um banco de dados MySQL. É possível fazer a autenticação de usuário por vários métodos (que são discutidos neste documento), os mais comuns são os servidores que implementam LDAP (por exemplo: OpenLDAP, ou O Microsoft Active Directory).

A próxima seção descreve brevemente cada um dos componentes da solução Zarafa.

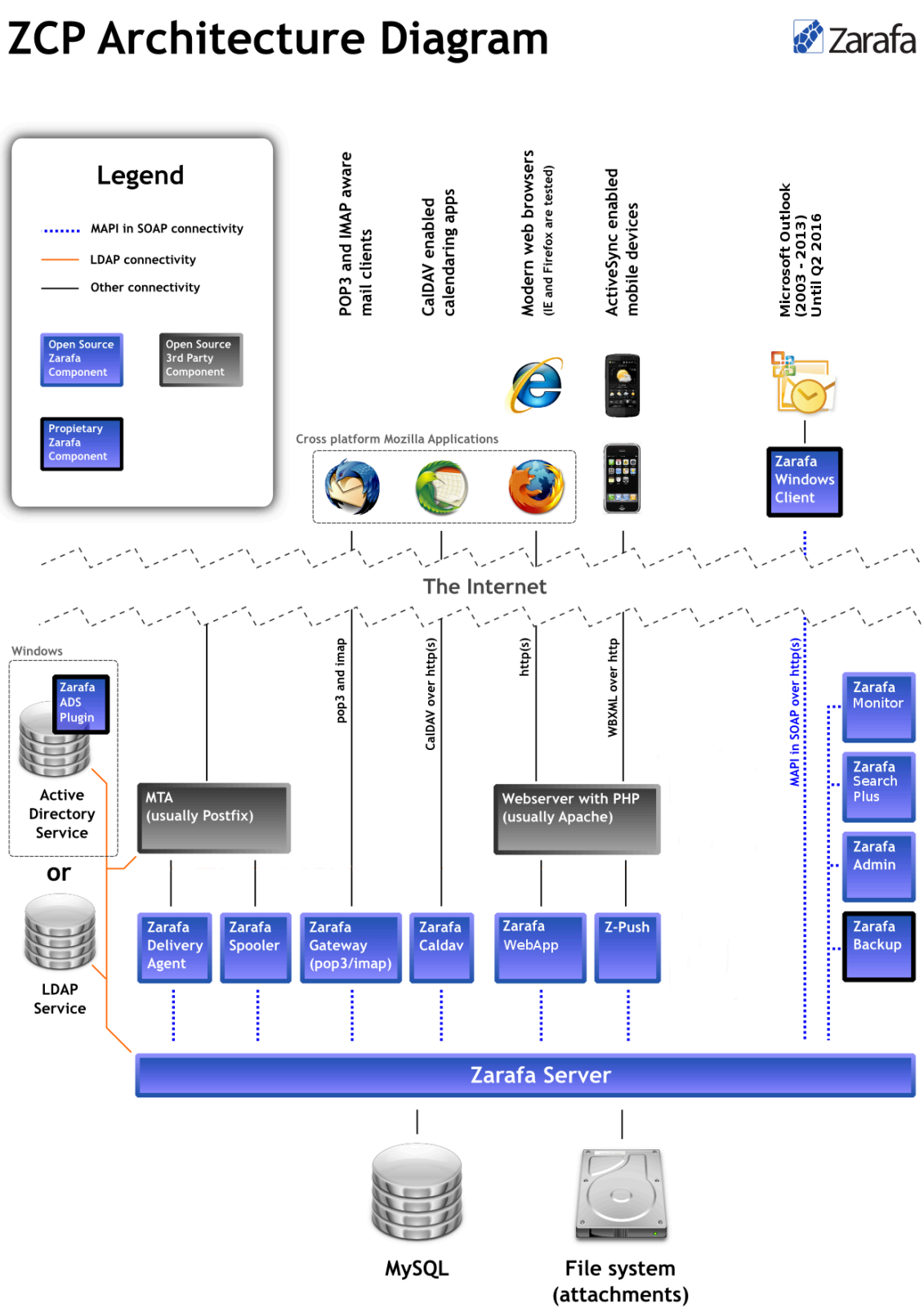


Figura 1.1. Diagrama da arquitetura do Pacote de Colaboração Zarafa.

1.3. Componetes

As instalações da Plataforma de Colaboração Zarafa (ZCP) em geral é composta pelos seguintes componentes:

- **Servidor Zarafa** (**`zarafa-server`** — O Servidor aceita conexões para todos os clientes através de SOAP (HTTP), e armazena os dados em um banco de dados SQL.
- **Manager do Zarafa License** (**`zarafa-licensed`**) — O processo licensed verificará quais funcionalidades estarão disponíveis dependendo da subscrição para a edição Small Business, Professional ou Enterprise.
- **Zarafa Windows Client** — O Zarafa Client provê a integração como o Outlook através de uma interface conhecida como MAPI. As conexões com o servidor são manipuladas pelo SOAP.
- **Zarafa WebAccess** (**`zarafa-webaccess`**) — Uma interface web cheia de recursos (com um "Look and Feel" do Outlook) que cria um ambiente colaborativo virtual em que os usuários podem se conectar a partir de qualquer computador com conexão à internet.
- **Zarafa WebApp** (**`zarafa-webapp`**) — The next generation collaboration web client, which offers integration with chat, presence and video conferencing.
- **Zarafa Delivery Agent** e **Zarafa Spooler** (**`zarafa-dagent`**, **`zarafa-spooler`**) — São as ferramentas que possibilitam a comunicação via e-mail com o mundo exterior. O agente entrega email a partir do Mail Transport Agent (MTA) para um usuário Zarafa. O spooler envia mensagens aguardando na fila de saída para o MTA especificado.
- **Zarafa Admin** (**`zarafa-admin`**) — Trata-se da ferramenta cuja função é gerenciar os usuários informações de usuário e grupos.
- **Zarafa Gateway** (**`zarafa-gateway`**) — Consiste num serviço opcional que proporciona aos usuários do Zarafa, acesso a POP3 e IMAP.
- **Zarafa Monitor** (**`zarafa-monitor`**) — Serviço que monitora as contas dos usuários no que se refere às cotas de armazenamento.
- **Zarafa Caldav** (**`zarafa-caldav`**) — Serviço opcional que proporciona suporte iCal e CalDAV aos usuários do Zarafa. O CalDAV é recomendado no caso de baixa velocidade e menor transferência de dados.
- **Zarafa Backup Tools** (**`zarafa-backup`**, **`zarafa-restore`**) — Trata-se de uma ferramentas de backup (brick-level) que serve para criar cópias de segurança simples de armazenamentos e restaurar tais backups em um momento posterior. Esta ferramenta somente está disponível em edições comerciais do Zarafa.
- **Zarafa search** — Optional service to provide full text indexing. This offers fast searching through email and attachments.
- **Apache** — Atende páginas web do WebAccess do navegador dos usuários.
- **PHP** — O WebAccess está escrito nesta linguagem de programação.
- **PHP-MAPI extension** — Módulo para PHP que possibilita o uso da camada de MAPI. Através deste módulo, as funções de MAPI são acessíveis para os desenvolvedores de PHP. Isso efetivamente significa que os clientes MAPI web podem ser escritos. O WebAccess é um desses clientes.

- **Python-MAPI extension** — Módulo para Python que possibilita o uso da camada de MAPI. Através deste módulo, as funções de MAPI são acessíveis para os desenvolvedores de Python.

Para conectividade com dispositivos móveis recomendamos o uso *Z-Push*[1] (see *Seção 5.5, "Configurar Z-Push (ActiveSync remoto para dispositivos móveis)"*), uma implementação open-source do protocolo ActiveSync. Para mais dispositivos móveis e dispositivos móveis que não suportam o protocolo ActiveSync nós enviamos os **Zarafa WebAccess Mobile** (**`zarafa-webaccess-mobile`**), que fornece interface web básica com funcionalidade limitada. Observe que este componente é obsoleto e provavelmente será removido da próxima versão do Zarafa.

1.4. Protocolos e conexões

Todos os aplicativos que se conectam diretamente ao Servidor Zarafa utilizam MAPI em SOAP para fazê-lo. Até mesmo o WebAccess usa a MAPI em SOAP (fornecido pela extensão PHP-MAPI) para se conectar ao servidor Zarafa.

O Zarafa Windows Client é um padrão Microsoft Windows compatível com provedor MAPI. Ele se conecta ao servidor (MAPI em SOAP) sobre o protocolo HTTP (S).

1.4.1. SOAP

SOAP é uma abreviatura de Simple Object Access Protocol. É um protocolo para troca de dados e realização de chamadas de procedimento remoto entre aplicativos através de uma rede ou da Internet.

O SOAP é baseado em XML e HTTP 1.1 (porta **`80`**, ou **`443`** no caso de HTTPS). Por causa desses padrões é possível se conectar de forma transparente através de proxys, permitindo conectividade na maioria das redes sem modificações.

1.4.2. HTTP seguro (HTTPS)

O Zarafa Windows Client tem a possibilidade de se conectar ao servidor através de HTTP seguro com SSL (HTTPS). Quando um perfil MAPI do Outlook é criado, é possível configurar a conexão para usar HTTPS. Todas as conexões através da rede será criptografada, fazendo escutas praticamente impossível.

O servidor Zarafa deve ser configurado para também aceitar conexões SSL. Por padrão é desabilitado, porque exige a criação de certificados SSL. Quando o certificado de servidor é criado, as conexões SSL podem ser aceitas diretamente de um cliente. Assim como uma opção extra de outros componentes do Zarafa (como o Zarafa Delivery Agent e o Zarafa Spooler) também pode se conectar via HTTPS para o servidor e autenticar usando a chave privada do servidor Zarafa.

1.5. Versões e licenciamento das Versões do Zarafa

1.5.1. A subscrição de avaliação

Quando se obtém uma versão de avaliação, um período de tempo é disponibilizado para testar o Zarafa com funcionalidade completa. É possível continuar usando o banco de dados atual quando após efetuar uma assinatura comercial válida.

Uma versão de avaliação pode ser solicitada em *http://www.zarafa.com/serial_request*.

[1] http://z-push.sourceforge.net

1.5.2. A Versão Community do Zarafa

está licenciado sob a Affero GPLv3. Esta edição pode ser usado com até três usuários com a proprietária do Windows Client do Zarafa (para conexão com o Microsoft Outlook). O WebAccess, o IMAP gateway e a sincronização móvel podem ser utilizado por usuários ilimitados.

Nota

Para ter suporte do Outlook na versão Communidade o Gestor de Licença do proprietário deve estar em execução, embora a assinatura não seja necessária.

1.5.3. Edições comerciais do Zarafa

Edições Small Business, Professional, Enterprise e Hosted necessitam uma subscrição comercial. Neste documento será explicitamente mencionado se uma funcionalidade ou componente está disponível somente em uma edição comercial.

1.5.4. Usuários ativos e não-ativos

Ao fazer uma assinatura da solução Zarafa é fixado um número básico de usuários. A assinatura básica é para um número fixo de usuários, mas pode ser estendida através da adição de Licenças de acesso de clientes extras, ou seja, ter uma assinatura de base para 10 usuários e uma CAL (Client Access Licenses) para 10 usuários, é funcionalmente equivalente a ter uma assinatura base de 20 usuários .

As assinaturas são baseados em usuários nomeados, ou seja, 10 usuários nomeados podem ser adicionados em um sistema com 10 usuários licenciados. No entanto, há também os usuários que não entra nesta contagem, estes são os chamados usuários "não-activos": eles não podem efetuar login. Um exemplo de um usuário não-activo é um usuário 'info' ou 'helpdesk'. Trata-se de um usuário que pode receber e-mails e tem todas as pastas padrão, mas não está autorizado a fazer login. Outros usuários não podem acessar uma conta de usuário 'info' como uma conta de delegado e enviar e-mails de lá.

Cada nova assinatura automaticamente implica na adição de uma quantidade extra de usuários não-ativos. A quantidade desses usuários é o equivalente a 150% da contagem do "usuários ativos" permitidos pela assinatura, com um mínimo de 20 usuários não-ativos. O número de usuários não-ativos foi aumentado a partir das versões 6.40.8 e 7.0.0 para permitir a criação de um armazenamento de arquivos de usuários não-ativos (Antes da versão 6.40.8 do Zarafa a quantidade máxima de usuários não-ativos era o equivalente a 50%).

Exemplos:

- Subscrição: 10 usuários
- Usuários ativos: 10
- Usuários não-ativos: 20
- Subscrição: 400 usuários
- Usuários ativos: 400
- Usuários não-ativos: 600

Se nem todas as contas de usuários estiverem em uso, é possível usá-las como contas de usuários não-ativos.

Nota

Usuários são definidos "ativos" ou "não-ativos" no momento da criação. Somente é possível converter usuários ativos em usuários não-ativos, ou vice-versa na versão 6.40 do Zarafa ou posterior: Nas versões anteriores a conta de usuário deve excluída e recriada como um tipo diferente.

In LDAP setups, the non-active flag of users can be controlled through the **`ldap_nonactive_attribute`** configuration directive. When using the DB back end, it is possible to specify the **`non-active`** flag with the **`-n`** option when using **`zarafa-admin`** to create users. The Unix user plugin uses the unix shell of the user as specified in **`/etc/passwd`** to determine if the store should be a non-active store.

Instalando

2.1. Requisitos de Sistema

2.1.1. Recomendções de Hardware

To give an estimate on the resource use of ZCP we have created the table below. These are merely guidelines, giving a rough estimation on what hardware is required. In this table we assume the CPU is under low load from other applications and size concerns the storage used in MySQL Server for the mailboxes.

Tabela 2.1. Minimal Hardware Recommendations

Database Size / Users	CPU (Cores)*	Memória	Disco rígido (HD)	Nível de Raid
< 5 GB / 1-25 users	2	2 GB	SATA, SAS, 7.2k	RAID 1
> 5 - < 10 GB / 26-50 users	4	4 GB	SAS, 7.2k	RAID 1
> 10 - < 20 GB / 51-100 users	4	6 GB	SAS, 7.2k	RAID 10
> 20 - < 50 GB / 101-200 users	6	8 GB	SAS, 10k	RAID 10
> 50 GB - < 100GB / 201-300 users	6	10 GB	SAS, 10k	RAID 10
> 100GB - < 250 GB / 301-500 users	6	12 GB	SAS, 10k	RAID 10
> 250 GB / 501-1000 users	8	16 GB	SAS or SATA/SSD Hybrid, 10k	RAID 10
> 1000 users	sizing depends on architecture, most likely Multi-Server			

Importante

Attachments do not require the same speed that is needed for the database storage. These can be saftely put on slower disks/different RAID levels.

Importante

Tuning of the server configuration and the individual software components for the specific onsite usage can drastically improve performance of your ZCP instance. For more than 500 users and/or a total mailbox storage bigger than 250Gb, as well as any high availability structures it is advised to seek professional engineering support.

2.1.2. Connection/bandwidth Recommendation

In order to seamlessly connect Outlook clients to Zarafa the network latency should not be higher than 20ms. Network latencies of 200ms (500ms under exceptional circumstances) should not be exceeded in order to aid the user acceptance.

The needed bandwidth is very much depended on the individual user behaviour. Based on large scale projects we use the following key figures to calculate the minimal needed bandwidth:

For implementations with more than 100 users (with external access) we calculate with an average bandwidth utilization of "x (actual amount of users) * 8kbit/s (ISDN speed)". In real world scenarios not all users will require exactly the same amount of bandwidth at the exact same time, which still leaves room to serve short term higher demands of single users (like requesting an attachment from the server).

Given these key figures (with +20% TCP protocol overhead) the following minimum bandwidth for Outlook users can be calculated: .Minimum Bandwidth Recommendations

Amount of users	Connection speed	Connection speed incl. TCP overhead
25	200 kbit/s	240 kbit/s
50	400 kbit/s	480 kbit/s
100	800 kbit/s	960 kbit/s
150	1200 kbit/s	1440 kbit/s
200	1600 kbit/s	1920 kbit/s
250	2000 kbit/s	2400 kbit/s
500	4000 kbit/s	4800 kbit/s
1000	8000 kbit/s	9600 kbit/s

Of course these are only bare minimums and providing a higher bandwidth will increase download speeds.

2.1.3. Plataformas suportadas

O ZCP consiste numa grande gama de componentes: alguns componentes de back-end que são executados em plataformas Linux, já outros componentes podem ser instalados nos computadores dos utilizadores finais. Esta seção lista as diferentes plataformas que suportamos.

Ao lançar uma nova verão 'major' (como 6.x.x ou 7.x.x) decidimos quais plataformas serão suportadas. Normalmente isso significa que a versão atual e a última antes da atual de uma plataforma serão suportadas. Durante o ciclo de vida da versão novas plataformas poderão ser adicionadas, mas nenhuma será removida.

Recomendamos use os pacotes x86_64 ou 64 bits se o hardware de 64 bits e OS estiverem disponíveis. É recomendado rodar em 64 bits sempre que possível.

Tabela 2.2. Plataformas suportadas para os componentes back-end do Zarafa

Lançamento de OS	Arquiterturas de CPU suportadas
Debian 6.0 (Squeeze)	i386, x86_64
Debian 7.0 (Wheezy)	i386, x86_64
RHEL 6	i686, x86_64

Lançamento de OS	Arquiterturas de CPU suportadas
RHEL 7	x86_64
SLES 11	i586, x86_64
SLES 12	x86_64
Ubuntu 12.04 LTS (Precise)	i386, x86_64
Ubuntu 14.04 LTS (Trusty Tahr)	x86_64
Univention 3.x	i386, x86_64
Univention 4.x	i386 (only for updates), x86_64

Tabela 2.3. Plataformas suportadas para **Zarafa Windows Client**, a **Ferramenta de Migração** e **Plugin ADS**.

Lancamento do Ms Windows	Arquiterturas de CPU suportadas
Windows Server 2003	32bit, 64bit
Windows Server 2008	32bit, 64bit
Windows Server 2012	32bit, 64bit
Windows XP	32bit, 64bit
Windows Vista	32bit, 64bit
Windows 7	32bit, 64bit
Windows 8	32bit, 64bit

Importante

Please be aware that this only specifies the architecture of the operating system and not the architecture of the Office suite.

Estes são as plataformas suportados do Microsoft Windows para os componentes que requirem uma plataforma Windows, como: o Cliente Zarafa, a Ferramenta de Migração e o Plugin ADS.

Nota

Atualmente a **Ferramenta de migração** não está disponível para plataformas de 64bits.

Mais informações sobre navegadores oficialmente suportados, clientes Outlook e níveis de suporte podem ser encontrados no *Documento de Suporte e Lifecycle*[1].

2.1.4. Dependências

Para construir ou instalar os componentes Zarafa back-end é preciso cumprir uma série de requisitos, Essas são as principais dependências do Zarafa:

[1] http://doc.zarafa.com/trunk/Support_Lifecycle_Policy/en-US/html-single

- **MySQL**, sem o serviço do MySQL disponível, o Zarafa Server não pode ser executado. Não é necessário executar o MySQL Server no mesmo sistema que o Zarafa Server já que não tem nenhuma dependência entre os pacotes. MySQL versão 4.0 ou inferior não irão funcionar corretamente. ZCP é testado com MySQL 4.1, 5.0 e 5.1.
- **Apache** ou qualquer outro servidor web que suporte PHP. O ZCP foi testado com Apache 2.0 e 2.2.
- **PHP**, autônomo como CGI ou, preferencialmente, como um módulo de webserver. O ZCP foi testado com PHP 4.3.xe a última versão 5.x.
- **Libicu**, uma biblioteca que fornece um robusto Unicode completo em recursos e em suporte local.
- **SMTP** servidor de escolha. O ZCP foi testado com Postfix, Exim, Sendmail e Qmail.
- Servidor **LDAP** de escolha (opcional para gerenciamento de usuários). O ZCP foi testado com OpenLDAP, NDS e com Microsoft Active Directory.
- **Catdoc** used to index text from Office documents.
- **Poppler-utils** used to index text from pdf files.
- **w3m** used to index HTML text from email.

Most of these dependencies are resolved automatically by the package manager of the Linux distribution that ZCP is being installed on. This allows the 3rd party components used by ZCP to be installed and upgraded automatically through the package manager of the distribution. Some dependencies in the table above are runtime dependencies, these have to be installed manually as they do not necessarily have to run on the same machine.

The default method of deploying ZCP is installing the packages on one of the Linux distributions we support, allowing the 3rd party components used by ZCP to be installed automatically through the package manager of the distribution. In this case the 3rd party components are upgraded in a standard way according to that distribution.

Nota

Se você estiver usando Debian ou Ubuntu, e está começando uma *nova* instalação do seu servidor, você pode usar o **tasksel** para instalar facilmente o LAMP inteiro (Apache, MySQL, PHP) pilha. Isto irá fornecer todos os pacotes necessários para o script de instalação Zarafa seja concluído com êxito.

Importante

We currently do not support the MySQL packages provided directly by Oracle, as they contain an already documented bug where libmysqlclient.so.* includes and exports symbols that actually belong to OpenSSL. For more information please refer to *ZCP-11674*[2] and *MySQL Bug: #65055*[3]

[2] https://jira.zarafa.com/browse/ZCP-11674
[3] http://bugs.mysql.com/bug.php?id=65055

2.2. Instalação

Existem 4 formas de instalar ZCP: (1) através de um gerente de distribuição de pacotes (2), utilizando o nosso script de instalação (3), instalar manualmente pacotes, e (4) a partir da código. Nesta seção cada um destes métodos é explicado junto com seus prós e contras.

Nota

Na versão Community o pacote **`Zarafa-licensed`** não é necessário, em contrapartida, para ter suporte Outlook nessa na versão Community, é preciso executar o daemon **`Zarafa-licensed`**.

Nota

O Calendário multi-usuário contido no pacote **`zarafa-webaccess-muc`** é um recurso não disponível na edição Community. Para esse recurso é nacessária uma assinatura válida.

Nota

As bibliotecas compartilhadas que fornecem os plugins do usuário são instalados em **`/usr/lib64/zarafa`**, e não no diretório **`/usr/lib/zarafa`**. Este caminho tem que ser ajustado no arquivo de configuração **`server.cfg`**. Defina o **`plugin_path`** para **`/usr/lib64/zarafa`**, para que o servidor possa encontrar os arquivos do plugin do usuário.

Nota

The MySQL option **`max_allowed_packet`** should not be set higher than 128M. This can conflict with Zarafa offline mode in Outlook. If the MySQL option must be higher you must also update the Zarafa offline clients. Change the value **`max_allowed_packet`** in **`C:\Program Files (x86)\Zarafa\Zarafa Outlook Client\MySQL\My.ini`** on the client.

2.2.1. Instalar usando o script de instalação

Importante

The possibility to install packages with the install.sh script was removed in Zarafa 7.2.

2.2.2. Instalando os pacotes manualmente

Utilize os pacotes para a distribuição usada. Veja a lista de distribuição na *Seção 2.1.3, "Plataformas suportadas"*. No caso de outras distribuições, é possível usar os pacotes de uma distribuição similar, mas tenha em mente que o Zarafa não pode dar suporte a essas instalações.

The packaging layout is displayed in the following table. Please note that not all download packages contain all the mentioned packages.

Tabela 2.4. Pacote de apresentação

Nome do Pacote	Discrição
pacemaker-zarafa	Contains script to more easily run Zarafa in Pacemaker environments
php(5)-mapi	Contém as extensões php-mapi
python-mapi	Contém as ligações Python MAPI para Zarafa
pthon-zarafa	Contains high-level Python bindings for Zarafa
python-zcp-license	Contém as ligações do licenciamento do python para o Zarafa
zarafa	Pode ser usado para instalar a todo o ZCP em um servidor
zarafa-archiver	Contains the separately sold zarafa-archiver service
zarafa-archiver-extra	Contains helper scripts for zarafa-archiver
zarafa-backup	Contém o backup Zarafa as ferramentas de restauração
zarafa-client	Contém o provedor MAPI para os clientes MAPI
zarafa-common	Contains shared files between ZCP services
zarafa-contacts	Contains the MAPI provider for adding contact folders in the addressbook
zarafa-dagent	Contém o delivery dagent
zarafa-dev	Contains C++ Development files for the Zarafa Collaboration Platform
zarafa-gateway	Contém o gateway POP3/IMAP
zarafa-gsoap	Contains stub generators for gSOAP
zarafa-gsoap-doc	Contains gSOAP documentation
zarafa-ical	Contains the ical library used for Caldav and iCal
zarafa-lang	Contains translations for Zarafa components
zarafa-libarchiver	Contém a biblioteca de-stubbing para o Zarafa Archiver

Nome do Pacote	Discrição
zarafa-libgoogle-perftools	Contains libraries for CPU and heap analysis, plus an efficient thread-caching malloc
zarafa-libgoogle-perftools-dev	Contains development files for the former
zarafa-libgsoap	Contains runtime libraries for gSOAP
zarafa-libgsoap-dev	Contains development files for the former
zarafa-libical	Contains the iCalendar library implementation in C (runtime)
zarafa-libical-dev	Contains development files for the former
zarafa-libmapi	Contains the Zarafa MAPI libraries
zarafa-libvmime	Contains the library for working with mime and rfc822 messages
zarafa-libvmime-dev	Contains development files for the former
zarafa-licensed	Contém os binários de código fechado e arquivos de configuração
zarafa-monitor	Contém a cota do monitor
zarafa-multiserver	Contém as bibliotecas multi-servidor
zarafa-presence	Contains the Zarafa presence daemon
zarafa-search-plus	Contains the Zarafa indexer using the xapian search engine
zarafa-server	Contém o servidor back-end e arquivos de configuração
zarafa-spooler	Contém o spooler
zarafa-utils	Contém as ferramentas de administração, como o Zarafa-admin e o Zarafa-fsck
zarafa-webaccess	Contains the WebAccess (only complementary support)
zarafa-webaccess-muc	Contém o calendário multi-usuário para o WebAccess
zarafa-webapp	Contains the WebApp a new and improved user webinterface for the Zarafa Collaboration Platform

Nome do Pacote	Discrição
zarafa-webapp-browsercompatibility	Displays the current support status of the used browser
zarafa-webapp-clockwidget	Adds a widget containing a clock
zarafa-webapp-contactfax	Adds syntax support for highlighting fax numbers
zarafa-webapp-dropboxattachment	Add attachments from and to Dropbox
zarafa-webapp-extbox	Image preview plugin
zarafa-webapp-facebook	Adds a Facebook widget
zarafa-webapp-feedback	Adds a feedback collector
zarafa-webapp-files	Files integration including adding from and to a files storage
zarafa-webapp-folderwidgets	Adds several folder plugins (unread mails, today's appointments)
zarafa-webapp-gmaps	Google Maps plugin
zarafa-webapp-oauthlib	OAuth v2.0 protocol library plugin
zarafa-webapp-pdfbox	PDF attachment viewer plugin
zarafa-webapp-pimfolder	personal inbox manager plugin
zarafa-webapp-quickitems	Quick Items plugin (new email, new appointment, etc)
zarafa-webapp-salesforce	Salesforce plugin
zarafa-webapp-shellgame	Shell game plugin
zarafa-webapp-spreed	Spreed.com meeting integration (old flash based service)
zarafa-webapp-statslogging	User interface statistics logging plugin
zarafa-webapp-sugarcrm	SugarCRM plugin
zarafa-webapp-titlecounter	Shows the amount of unread mails in the browser title
zarafa-webapp-twidget	Zarafa WebApp Twitter widget
zarafa-webapp-webappmanual	Adds a link to the online WebApp manual
zarafa-webapp-webodf	WebODF plugin
zarafa-webapp-xmpp	XMPP widget
zarafa-webapp-zperformance	Performance monitoring plugin

Nota

Não misture pacotes de diferentes distribuições! Escolha uma distribuição, e utilize os pacotes desta. Se essa regra não for seguida, ocorrerão erros!

2.2.2.1. Distribuições baseadas em RPM

Utilize o comando abaixo para instalar os pacotes do ZCP nas distribuições baseadas em RPM:

```
rpm -Uvh <package files you want to install>
```

Replace **<package file>** with the packages found in the tarball. Start with **libvmime**, **libical** and **zarafa** (in this order) then install the other packages. The package manager might find unresolved dependencies, try to install packages for these dependencies as normal would be done for that distribution (**yum -i** on Red Hat, **zypper -i** on SLES).

Nota

As of Zarafa 7.1.6 the packages libboost_system and libboost_filesystem are required on SLES 11 SP3. Unfortunately these packages are not part of the standard distribution and are only available from the SDK. To successfully install or update the Zarafa packages it is therefore necessary to either download the iso file of the first DVD of the SDK and include it via Zypper or add the „SDK Pool Repository“ and „SDK Update Repository“ to the online update of SLES.

2.2.2.2. Distribuições baseadas em DEB

Em distribuições baseadas em DEB (as mais comuns são Debian e Ubuntu) utilize:

```
dpkg -i <package files you want to install>
```

Para instalar as dependências corretas para o ZCP pode-se usar o **apt-get** ou uma ferramenta equivalente.

Para o MySQL, utilize:

```
apt-get install mysql-server
```

Para o Apache com o suporte PHP necessário, utilize:

```
apt-get install apache2-mpm-prefork libapache2-mod-php5
```

Se os pacotes Zarafa não forem instalados por causa das dependências, utilize o seguinte comando para instalar estas dependências:

```
apt-get -f install
```

Se o Apache com suporte a PHP for instalado depois que os pacotes Zarafa já tiverem sido instalados, por favor use o seguinte comando para atualizar automaticamente a configuração do PHP:

```
dpkg-reconfigure zarafa
```

Nota

The quickest way to install Zarafa is not by one by one selecting packages to install and then resolving their dependencies, but by doing it the other way around. Therefore it is recommended to simply remove packages you explicitly do not want (for example pacemaker-zarafa or zarafa-multiserver) and simply installing the rest by issuing "dpkg -i *.deb" followed by "apt-get install -f" to get the missing dependencies from apt.

2.2.2.3. Instalação a partir do código

O ZCP não conta com suporte oficial pela Zarafa quando instalado a partir do código, mas em algumas situações - ou seja, usando ZCP em ambientes sem apoio, ou quando se prepara correções para o ZCP - é muito útil para instalar a partir do cigo. Como a maioria dos ZCP é distribuído sob uma licença open source (AGPLv3), esta é uma das condições para se instalar a partir do código.

A instalação exata do ZCP a partir da do código está além do escopo deste documento. Além disso, o procedimento é ligeiramente diferente para cada distribuição e está sujeito a alterações. Portanto, visite nosso *wiki*[4] (pesquise por 'from source') para as ter acesso às últimas informações sobre esse tipo de instalação em particular.

2.3. Solucionando problemas de instalação

2.3.1. Processos do servidor

Make sure at least MySQL 5.0 is installed. The server will only run with this version of the database server or a more recent version.

Se os ocorrerem erros no carregamento das bibliotecas ou se a conecção com o MySQL falhar, os erros serão impressos no registro. Sempre verifique se o serviço foi iniciado corretamente.

Quando uma opção de configuração inválido estiver marcada em algum arquivo de configuração, o serviço não será iniciado. As opções de erro sempre sempre serão impressos no console.

2.3.2. WebAccess & WebApp

Para acessar correctamente o WebAccess, são necessárias as seguintes extensões PHP:

- **`gettext`**
- **`session`**
- **`iconv`**
- **`xml`**

Algumas distribuições, por padrão, oferecem suporte para essas extensões através do pacote PHP. Para distribuições SuSE, estes módulos são fornecidos por RPMs separado, por exemplo:

```
php5-gettext-5.2.8-37.4.x86_64.rpm
php5-iconv-5.2.8-37.4.x86_64.rpm
```

[4] http://wiki.zarafa.com/

As versões mais antigas devem diferir das versões mais recentes do SUSE

Para o Red Hat Enterprise Linux e distribuições Debian-based, esse módulos são proporcionados pelo pacote php normal que ainda não terá sido instalado, devido às dependências.

Se estiver enfrentando problemas com o envio de anexos, certifique-se que o servidor web esteja hábil para criar arquivos no diretório **`WebAccess/tmp`**. Se o usuário for automaticamente desconectado quando tentar acessar o WebAccess, certifique-se queo PHP esteja configurado para:

```
register_globals = off
```

Se for usada uma distribuição em combinação com SELinux, poderá surgir uma mensagem de erro ao registrar no uso do WebAccess. A mensagem padrão sugere que a senha digitada esteja errado ou o que o servidor Zarafa não esteja funcionando. Quando o SELinux está ativado, ele bloqueia a conexão do servidor para o servidor Zarafa. O politica do SELinux Zarafa permite que este possa ser encontrado em*http://www.zarafa.com/wiki/index.php/Zarafa_Selinux_policy*.

ou SELinux pode ser desabilitado usando o comando:

```
setenforce permissive
```

Se optado por desabilitar o SELinux o arquivo **`/etc/sysconfig/selinux`** também precisa ser editado, para que o serviço continua desabilitado apos o reboot.

Mais informações sobre o SELinux podem ser encontrados em *http://fedora.redhat.com/docs/selinux-faq* (em inglês).

By default, the WebApp installation requires HTTPS to be configured. A Description how to enable SSL for WebAccess or WebApp can be found on *http://www.zarafa.com/wiki/index.php/Securing_Zarafa_WebAccess_with_SSL*. When SSL is not desired, it is possible to disable the configuration check for these security options inside the config.php file, and disable the option **`CONFIG_CHECK_COOKIES_SSL`**.

2.4. Removing Zarafa

Zarafa can easily be removed by executing the **`uninstall.sh`** script which is provided in the downloadable packages. The script takes care of the following steps:

- stop all Zarafa services
- remove all packages directly related to Zarafa (excluding MTA, MySQL, and Apache)
- (optionally) delete the Zarafa database from MySQL
- (optionally) remove configuration and logfiles of Zarafa

Nota

Before removing Zarafa make sure that all needed data has been migrated to another system. After removing Zarafa there is no meaningful way to access the saved data.

Atualizando a versão

3.1. Preparação

Antes de atualizar o Zarafa para uma nova versão, recomenda-se fazer um backupdo banco de dados e da configuração do s arquivos.

Nota

Ao realizar o upgrade de uma versão licenciada do ZCP para uma nova versão "major", como de 6.40.x para 7.0.x, o serial de subscrição precisa ser convertido. O processo de conversão do serial pode ser realizado no *Portal Zarafa*[1].

Primeiramente interrompa p servidor MTA que estiver rodando em seu servidor. Caso ocorram erros durante a atualização nenhum e-mail será perdido. Em caso de postfix, execute:

```
/etc/init.d/postfix stop
```

Agora, interrompa os serviços em execução, de modo que a banco de dados não mais esteja em uso:

```
/etc/init.d/zarafa-spooler stop
/etc/init.d/zarafa-server stop
/etc/init.d/zarafa-licensed stop
```

E os serviços opcionais também, se eles foram iniciados:

```
/etc/init.d/zarafa-dagent stop
/etc/init.d/zarafa-gateway stop
/etc/init.d/zarafa-ical stop
/etc/init.d/zarafa-indexer stop
/etc/init.d/zarafa-search stop
/etc/init.d/zarafa-monitor stop
```

Importante

Quando os anexos são mantidos no banco de dados, uma atualização para a versão 6.30.x ou posterior vai aumentará o arquivo de armazenamento de banco de dados mais que o tamanho combinado de todos os anexos (como armazenado na "tabela lob"). Durante a atualização de uma tabela temporária para armazenar todos os anexos é criado e removido, uma vez que não é possível reduzir o arquivo de armazenamento de dados que vai crescer o tamanho combinado dos anexos armazenados nele.

Informações sobre a migração dos anexos do banco de dados para o sistema de arquivo podem ser encontradas em *nosso wiki*[2].

[1] https://portal.zarafa.com/
[2] http://www.zarafa.com/wiki/index.php/Store_attachment_outside_of_the_database

3.2. Criando backups

Quanto à criação de backups de banco de dados e arquivos de configuração: Faça uma cópia do diretório **`/ etc / Zarafa`**, que contém os arquivos de configuração.

```
cp -r /etc/zarafa /etc/zarafa.bck
```

As Zarafa stores attachments of items on the filesystem, make a copy of the attachment directory.

```
cp -r /var/lib/zarafa /var/lib/zarafa.bck
```

Para fazer o backup do banco de dados MySQL um **`mysqldump`** pode ser executado:

```
mysqldump --single-transaction -p zarafa > zarafa.sql
```

ou pode-se copiar todo o diretório de dados so MySQL:

```
/etc/init.d/mysqld stop
cp -r /var/lib/mysql /var/lib/mysql.bck
cp -r /etc/my.cnf /etc/my.cnf.bck
```

Nota

The paths could be different when default configuration is changed.

3.3. Dependências da versão 7 do Zarafa

Depois que o backup tiver sido criado com êxito, os pacotes do Zarafa podem ser atualizados. Existem algumas novas dependências que preisam ser resolvidas antes do pacotes serem atualizados.

Tabela 3.1. Dependências da versão 7 do Zarafa

Distribuição	Dependências
Debian 5	libboost-filesystem1.35.0, libboost-system1.35.0, libicu38, w3m, python-mysqldb
Debian 6	libboost-filesystem1.42.0, libboost-system1.42.0, libicu44, w3m, python-mysqldb
Debian 7	libboost-filesystem1.49.0, libboost-system1.49.0, libicu48, w3m, python-mysqldb
RHEL5	libicu, w3m, MySQL-python
RHEL6	boost-filesystem, boost-system, libicu, w3m, MySQL-python
SLES10	libicu, w3m, python-mysql
SLES11	libicu, w3m, python-mysql
Ubuntu 8.04	libicu38, w3m, python-mysqldb
Ubuntu 10.04	libboost-filesystem1.40.0, libboost-system1.40.0, libicu42, w3m, python-mysqldb
Ubuntu 12.04	libboost-filesystem1.46.1, libboost-system1.46.1, libicu48, w3m, python-mysqldb

3.4. Realizando a atualização em RPM baseada em distribuições

Após os backups terem sido criados a atualização pode ser realizada de forma semelhante à instalação manual de um pacote. Para instalações baseadas em RPM use o seguinte comando:

```
rpm -Uvh <package name>.rpm
```

Nota

Na edição da Community o pacote **`zarafa-licensed`** não é necessário. Somente quando na integração com o Outlook é usado o daemon o zarafa-licensed é necessário.

Após os novos pacotes forem instalados, os arquivos de exemplo de configuração encontrados no diretório / usr / share / doc / Zarafa / example-config diretório podem ser verificados para novas opções de configuração. As novas mudanças podem ser encontradas em *Release Notes*[3].

3.5. Realizando a atualização em Debian baseado em distribuições

Descompacte o tarball: tar zxvf zcp-7.0.0rc1-26667-debian-6.0-i386-free.tar.gz

Instale o libvmime 0.9 que vem com Zarafa:

```
dpkg -Bi libvmime0_0.9.2*
```

Instale o libical que vem com o Zarafa:

```
dpkg -Bi libical0_0.44*
```

Instale o pacote python-mapi que vem com oo Zarafa:

```
dpkg -i python-mapi*
```

Para Debian baseado em instalacões execute o comando abaixo para instalar a atualização do Zarafa:

```
dpkg -Bi <package name>
```

Dependendo do conjunto de pacotes da versão 6.x que tiver sido instalada este comando pode acabar apresentando erros no "Zarafa" e no pacote "Zarafa-licensed". Devido à grande divisão e renomeação de pacotes de alguns conflitos não são diretamente resolvido pelo "dpkg". Se você receber quaisquer erros durante a atualização destes pacotes, numa segunda tentativa de instalar esses pacotes usando:

```
dpkg -i <package name>
```

[3] http://doc.zarafa.com/trunk/Release_Notes/en-US/html/_config_file_changes.html

ou execute o comando abaixo:

```
apt-get install -f
```

isso deve resolver tudo.

Quando for questionado a respeito de alterações nos arquivos de configuração, o procedimento a ser seguido irá depender da atual situação e de qual a melhor opção.

Nota

Na edição da Community o pacote **`zarafa-licensed`** não é necessário. Somente quando na integração com o Outlook é usado o daemon o zarafa-licensed é necessário.

Após os novos pacotes forem instalados, os arquivos de exemplo de configuração encontrados no diretório / usr / share / doc / Zarafa / example-config diretório podem ser verificados para novas opções de configuração. As novas mudanças podem ser encontradas em *Release Notes*[4].

3.5.1. Etapas de pré atualização da versão 6.40 do Zarafa

Existem algumas mudanças nas confgurações da versão 6.40 e versões posteriores para suportar novos recursos na LIsta de Endereços Global, como contatod, grupos dinân]micos e grupos de segurança. Especialmente no caso do uso do plugin de usuŕio LDAP, o servidor não iniciará corretamente sem que algumas mudanças tenham sido feitas nos arquivos de configurações LDAP.Entretanto, pode ser útil visualizá-las para configurar novas opções.

Please check the upgrade page on *our wiki*[5] for up-to-date upgrade details.

Para suportar corretamente contatos do Microsoft Active Directory, o **`ldap_user_unique_attribute`** deve ser alterado do **`objectSid`** para o **`objectGuid`**. Uma vez que esse é o identificador exclusivo para os usuários, alterá-lo sem atualizar o banco de dados fará o servidor Zarafa deletar todos os usuários. Como isso não é desejado, é preciso usar o script **`db-upgrade-objectsid-to-objectguid.pl`** encontrado no diretório **`/usr/share/zarafa/doc/`**. Esse script irá detectar as configurações do LDPA existente em no arquivo **`/etc/zarafa/server.cfg`** e alterar o banco de dados para um ID único. Após o script, é necessário atualizar o arquivo de configuração do LDAP para usar o novo ID único. Certifique-se que o processo do servidor Zarafa não está sendo executado ao usar este script.

Nota

Ao usar o OpenLDAP não há necessidade de alterar a **`ldap_user_unique_attribute`**.

A opção de 'send-as' no ldap são opostos entre a versão 6.30 e 7.0. Essa modificação foi feita para poder suportar grupos para a funcionalidade do 'send-as'. Se a opção do 'send-as' está sendo usada para usuários com a versão 6.30, o script **`ldap-switch-sendas.pl`** precisa ser executado. Este script irá realizar o upgrade das informações no LDAP ou ADS para o novo formato.

[4] http://doc.zarafa.com/trunk/Release_Notes/en-US/html/_config_file_changes.html
[5] http://www.zarafa.com/wiki/index.php/Upgrading_to_6.40

```
cd /usr/share/doc/zarafa
chmod 755 ldap-switch-sendas.pl
./ldap-switch-sendas.pl
```

Na versão 6.40, as permissões de 'Enviar como' são selecionadas pelo usuário. Por exemplo, um usuário não ativo **info@company** existe e alguns usuários precisam enviar com aquele endereço no remetente. Os usuários são adicionados sobre o objeto **info@company** na lista de atributos do 'Enviar como'

Na configuração do LDAP, as opções de pesquisa separada para cada objeto do Banco de Dados são combinados em uma opção de filtro de pesquisa chamado **ldap_search_base**. Todas as outras opções **search_base** antigas podem ser removidos. Além disso, todas as opções de escopo devem ser removidos.

Next, object types must be defined. This normally done by means of the **objectClass** attribute. Every user object must be defined by its **objectClass**.

Por fim, os velhos filtros de pesquisa por objeto podem ser esvaziados, uma vez que esses são duplicações. É ainda aconselhável a utilização de **zarafaAccount** no filtro do usuário, assim as opções ainda estarão disponíveis.

Para proteger os servidores de exclusão de usuários uma opção de modo de segurança está disponível em **server.cfg**. Ao ativar essa opção se desativa-se todas as ordens de excluir e cria-se ações de usuários e grupos.

Adicione a opção abaixo no **/etc/zarafa/server.cfg** para ativar o modo de segurança:

```
user_safe_mode = yes
```

Verifique o servidor logfile apos iniciar o Servidor Zarafa para detecção de mudanças de usuários. Se nenhum usuário for criado ou deletado o arquivo de configuração estará correto e **user_safe_mode** pode ser desabilitado.

Importante

É altamente aconselhavel usar o modo de segurança somente após uma atualização. Quando a atualização for concluida com sucesso, o safe_mode deve ser desabilitado. A execução de um sistema de produção com o safe_mode ativado pode resultar em problemas de desempenho.

Nota

When upgrading ZCP 6.30 to 7.0, it is not necessary to first upgrade to the 6.40 packages.

3.5.2. Da versão 6.40 para 7.0.0 e posterior

O tempo necessário para a instalação depende da quantidade de dados que precisa ser convertida ao atualizar para a versão 7.0 do Zarafa, o servidor por padrão, se recusará a atualizar o banco de dados.

Upgrading the Zarafa database will take some hours at least, please keep in mind that the Zarafa system can’t be used during this upgrade. To provide some insight into the upgrade duration we created an upgrade-calculation script to run on your 6.40 installation server. The estimate is rough

as we refine it on a regularly basis using community feedback. Download the script at: *http://www.zarafa.com/upgrade*

Quando você atualizar, a diferente entre o tempo real de upgrade e os valores calculados nos ajudam enormemente. Por favor, nos informe sobre os dados da atualização para que possamos aprimorar o script.

Importante

Por favor, certifique-se que suas definições de innodb do servidor MySQL estejam otimizados. Para mais informações sobre parâmetros importantes de ajustes do MySQL, ver *Capítulo 9, Ajuste de desempenho*.

To upgrade the database, it is recommended to use the **`zarafa7-upgrade`** tool that comes with the zarafa-server package in ZCP 7.0. This upgrade tool will perform the necessary upgrade steps and will keep you informed about the progress. The **`zarafa7-upgrade`** tool can be found in /usr/share/doc/zarafa and requires the **`python-mysqldb`** or **`MySQL-python`** package, as well as the **`python-mapi`** packages. That last one can be found in the ZCP tarball.

Before the **`zarafa7-upgrade`** script can be started, the **`Zarafa-server`** has to be started to convert the database to the latest 6.40 database revision.

```
/etc/init.d/zarafa-server start
```

Confere o arquivo de log **`/var/log/zarafa/server.log`** para monitorar o andamento deste update.

```
[root@zarafa ~]# tail -f /var/log/zarafa/server.log
Mo 27 Feb 2012 09:50:48 CET: Starting zarafa-server version 7,0,5,31880, pid 30725
Mo 27 Feb 2012 09:50:48 CET: Connection to database 'zarafa' succeeded
Mo 27 Feb 2012 09:50:48 CET: WARNING: zarafa-licensed not running, commercial features will
 not be available until it's started.
Mo 27 Feb 2012 09:50:48 CET: Start: Move IMAP subscribed list from store to inbox
Mo 27 Feb 2012 09:50:55 CET: Done: Move IMAP subscribed list from store to inbox
Mo 27 Feb 2012 09:50:55 CET: Start: Update sync table time index
Mo 27 Feb 2012 09:50:58 CET: Done: Update sync table time index
Mo 27 Feb 2012 09:50:58 CET: Start: Update changes table state key
Mo 27 Feb 2012 11:05:12 CET: Done: Update changes table state key
Mo 27 Feb 2012 11:05:12 CET: Start: Converting database to Unicode
Mo 27 Feb 2012 11:05:12 CET: Will not upgrade your database from 6.40.x to 7.0.
Mo 27 Feb 2012 11:05:12 CET: The recommended upgrade procedure is to use the zarafa7-upgrade
 commandline tool.
Mo 27 Feb 2012 11:05:12 CET: Please consult the Zarafa administrator manual on how to
 correctly upgrade your database.
Mo 27 Feb 2012 11:05:12 CET: Alternatively you may try to upgrade using --force-database-
upgrade,
Mo 27 Feb 2012 11:05:12 CET: but no progress and estimates within the updates will be
 available.
Mo 27 Feb 2012 11:05:12 CET: Failed: Rollback database
Mo 27 Feb 2012 11:05:12 CET: Can't update the database: Unable to upgrade zarafa from version
 6.40.30778 to 7.0.5.31880
Mo 27 Feb 2012 11:05:12 CET: Server shutdown complete.
```

Quando o banco de dados foi convertido para o layout correto, o Zarafa-server irá parar a execução automaticamente e alertar que o update deve ser feito manualmente executando o script zarafa7-upgrade. Executa o script **`zarafa7-upgrade`** para terminar a conversão o layout do banco de dados e converter os dados para unicode.

No Debian e no Ubuntu é necessário descompactar o arquivo previamente:

```
gunzip /usr/share/doc/zarafa/zarafa7-upgrade.gz
python /usr/share/doc/zarafa/zarafa7-upgrade
```

Para executar a ferramenta de atualização utilize:

```
[root@zarafa ~]# python /usr/share/doc/zarafa/zarafa7-upgrade
Converting search folders to Unicode: 879 / 879 (100%)
Converting properties for IO performance: 69318024 / 69318024 (100%)
Creating counters for IO performance: 16 / 16 (100%)
Creating common properties for IO performance: 4 / 4 (100%)
Creating message attachment properties for IO performance: 2 / 2 (100%)
Creating tproperties for IO performance: 69318023 / 69318023 (100%)
Converting hierarchy for IO performance: 69318023 / 69318023 (100%)
Creating deferred table for IO performance: 1 / 1 (100%)
Converting changes for IO performance: 56266424 / 56266424 (100%)
Converting names table to Unicode: 10331 / 10331 (100%)
```

O script irá converter todas as tabelas para UTF-8 para ser totalmente compatível com unicode e irá converter as tabelas do banco de dados para o novo layout do ZCP 7.0. O script indicará o progresso do update como mostrado acima.

Alternativamente, o servidor pode ser forçado a atualizar o banco de dados iniciando-o com a opção --force-database-upgrade.

Importante

Não é recomendado usar a opção --force-database-upgrade, uma vez que está não tem indicação de progresso e não pode ser interrompida.

Nota

Ao fazer o upgrade de versões antigas do Zarafa, por exemplo da versão 6.30.x do Zarafa, o Zarafa-server vai primeiro atualizar o banco de dados para o layout da versão 6.40, após esse procedimento o script de atualização poderá ser executado.

3.5.3. From 7.0 to 7.1.0 and higher

The **`zarafa-indexer`** has been replaced by the **`zarafa-search`** package. Make sure you remove **`zarafa-indexer`** when upgrading to 7.1 and install the **`zarafa-search`** package. You can remove the old index directories and files as they won't be used anymore. All directories found in the index_path location (default: /var/lib/zarafa/index/) can be removed. The new **`zarafa-search`** application only creates .kct files and will not interfere with the old index files.

The **`zarafa-search`** options in the server.cfg file have also changed. All the old indexer options are replaced by new search options. The following config options can be removed from the old server config file:

```
index_services_enabled
index_services_path
index_services_search_timeout
```

These options are replaced by the following search options:

```
search_enabled = yes
search_socket = file:///var/run/zarafa-search
search_timeout = 10
```

These options are by default set, so there is no need to change these config values to use the new zarafa-search engine after the upgrade.

When using Debian or Ubuntu, please check if the file **/etc/default/zarafa** contains the following lines at the end.

```
# set to no to disable zarafa-search at startup
SEARCH_ENABLED=yes

# Location of the configuration files
SEARCH_CONFIG=/etc/zarafa/search.cfg

# Additional options that are passed to the Daemon.
SEARCH_OPTS=""
```

If these lines are not available, the **zarafa-search** service will not start automatically. The lines can be manually added or the file can be overwritten by the file provided in the package.

```
mv /etc/default/zarafa.dpkg-dist /etc/default/zarafa
```

ZCP 7.1 introduces stored procedures in MySQL to improve streaming speed used in the **zarafa-search** and for offline users. This changes the privileges zarafa-server needs to correctly use the MySQL database. The mysql user needs the CREATE PROCEDURE privilege, which can be given using the GRANT sql command. Please see *Capítulo 4, Configurar Componentes ZCP* for a full list of all required privileges and grant examples.

Besides this the "enable_sql_procedures" option must be enabled in the server.cfg

The SQL Procedures allow for some optimized queries when streaming with enhanced ICS. This is default disabled because you must set *thread_stack = 256k* in your MySQL server config under the [mysqld] tag and restart your MySQL server.

Nota

Note that any search indexes made with prior releases of 7.1.0 (RC or beta) need to be dropped before use with the final or RC3.

3.6. Finalizando a atualização

Após a verificação das novas configurações, os serviços podem ser novamente iniciados:

```
/etc/init.d/zarafa-server start
/etc/init.d/zarafa-spooler start
/etc/init.d/zarafa-licensed start
```

Os serviços opcionais também podem se iniciados novamente:

```
/etc/init.d/zarafa-dagent start
/etc/init.d/zarafa-gateway start
/etc/init.d/zarafa-ical start
/etc/init.d/zarafa-search start
```

```
/etc/init.d/zarafa-monitor start
```

Como os upgrades geralmente incluem modificações na extensão **`php-mapi`** o servidor web precisa ser reinicializado também:

```
/etc/init.d/apache2 restart
```

ou

```
/etc/init.d/httpd restart
```

O Zarafa 7.0 tem um novo gateway IMAP/POP3 melhorado. O novo gateway oferece melhor compatibilidade e um desempenho superior usando informações adicionais que são armazenadas no banco de dados e no diretório de anexos do Zarafa-server. Como essas informações adicionais ocuparão mais espaço em disco e somente serão usados quando os usuários estão se conectando fora do IMAP, tais recursos foram, por padrão, desabilitados pelo IMAP/POP3.

Quando os usuários tiverem acesso ao IMAP ou POP3 esses recursos terão que ser habilitados manualmente. Leia mais sobre habilitar/desabilitar recursos em *Seção 8.7, “Manuseio do Zarafa Feature”*

To generate for all existing message an optimized IMAP version, the **`optimize-imap.py`** script is available. By executing this script for every existing email the envelope structure and body structure and store these entries in the database. Additionally the whole RFC822 message file is generated and stored gzip compressed in the attachment directory.

The script will only generate this data for the users who have IMAP and POP3 enabled.

To execute the script use the following command:

```
python /usr/share/doc/zarafa-gateway/optimize-imap.py
```

To optimize one or more specific users use the following command: python /usr/share/doc/zarafa-gateway/optimize-imap.py <user1> <user2> <user3>

Nota

For new emails received on ZCP 7.0 the optimized IMAP data is stored automatically when users have IMAP or POP3 enabled.

Configurar Componentes ZCP

A maioria dos componentes ZCP e de terceiros são configurados por um arquivo de configuração. Essa seção explica as opções mais comuns para configurar estes componentes. É importante ressaltar que geralmente os componentes precisam ser reinicializados para usarem os arquivos de configuração modificados, leia mais em *Capítulo 7, Gerenciando os serviços do Zarafa*.

Resumindo, depois de realizar modificações em um arquivo de configuração de um componente, este componente precisa ser reinicializado com:

```
/etc/init.d/zarafa-<nome do componente> restart
```

4.1. Configurando o Servidor Zarafa

O componente Zarafa Server é configurado por um arquivo de configuração válido pelo sistema inteiro, normalmente localizado em:

```
/etc/zarafa/zarafa-<nome do componente>.cfg
```

Ao instalar o ZCP um exemplo deste arquivo pode ser encontrado em:

```
/usr/share/doc/zarafa-<component name>/example-config/zarafa-<component name>.cfg
```

As opções e seus valores padrão são explicados nos comentários do exemplo do arquivo de configuração e na página man:

```
man <nome do componente>.cfg
```

Por exemplo:

```
man zarafa-server.cfg
```

Caso uma linha não seja presente o valor padrão é usado. Para os setups básicos os valores do exemplo poderão ser usados. Neste capitulo serão explicados somente as opções básicas de configuração do Zarafa Server.

O Zarafa Server precisa de um banco de dados MySQL para funcionar, e assim precisa saber como conectar ao servidor MySQL e os dados para a autenticação para o banco a ser usado. O banco com suas tabelas serão inicializados na primeira execução.

Make sure that the MySQL user that the Zarafa Server uses to connect to the database has all privileges, including the right to create a new database. Also make sure to give the user enough permissions to connect from localhost to this database, or --if the Zarafa server connects over the network to the MySQL database-- allow it to connect from the IP-address from which the Zarafa Server will connect.

Por exemplo, a instrução MySQL a seguir concede todos os privilégios para o usuário "zarafa" com a senha "senha" do localhost:

```
GRANT ALL PRIVILEGES ON zarafa.* TO
'zarafa'@'localhost' IDENTIFIED BY 'senha';
```

If you want to restrict the privileges of the zarafa connection, the following grant command lists only the required privileges:

```
GRANT alter, create, create routine, delete, drop, index, insert, lock tables, select, update
 ON zarafa.* TO
'zarafa'@'localhost' IDENTIFIED BY 'password';
```

Para configurar o servidor Zarafa para usar o servidor MySQL as opções começando com **`mysql`** em **`zarafa-server.cfg`** precisam ser definidos. Uma vez que esta é a configuração está completa o o serviço Zarafa deve iniciar normalmente.

4.2. Configurar a linguagem em distribuições baseadas em RPM

Após a criação de novos usuários o Servidor Zarafa irá criar automaticamente a caixa de correio. Esta caixa de correio é criado por padrão no idioma do servidor Linux. Quando um outro idioma é necessário o seguinte arquivo de configuração tem que ser alterado:

```
/etc/sysconfig/zarafa
```

Modifique a opção **`ZARAFA_USERSCRIPT_LOCALE`** para a linguagem correta, como por exemplo **`pt_BR.UTF-8`** ou **`fr_FR.UTF-8`**.

In order to use this language setting make sure the language packs are installed. Red Hat and SuSE based systems contain all language packs by default.

A opção **`ZARAFA_LOCALE`** no arquivo **`/etc/sysconfig/zarafa`** pode ser usado para iniciar o Servidor Zarafa no idioma correto. Esta configuração de idioma é usado para definir as opções padrão, como o nome da pasta pública no idioma correto.

A linguagem do WebAccess pode ser definido na tela de login. Isso pode ser configurado por login de usuário.

Importante

Ao fazer o upgrade de uma versão anterior ZCP, reveja as configurações de idioma a partir do ZCP 7.0.0 o locale tem de ser definido em UTF-8.

4.3. Configurar a linguagem em distribuições baseadas em Debian

When adding new users the Zarafa Server will automatically create the actual mailbox. The mailbox is by default created in english language. To create the mailboxes in English, it is required to have the en_US.UTF-8 locale installed.

When the mailbox should be created in another language the following configuration file has to be changed:

```
/etc/default/zarafa
```

Modifique a opção **`ZARAFA_USERSCRIPT_LOCALE`** para a linguagem correta, como por exemplo **`pt_BR.UTF-8`** ou **`fr_FR.UTF-8`**.

In order to use this language setting make sure the correct language packs are installed and configured.

To install a language pack on an Ubuntu based system, use the following command (this example is for the Dutch -nl pack):

```
apt-get install language-pack-pt
```

On Debian based systems the locale needs to be enabled in /etc/locale.gen. The following command can be used to easily enable and generate the needed locales:

```
dpkg-reconfigure locales
```

A opção **ZARAFA_LOCALE** no arquivo **/etc/default/zarafa** pode ser usado para iniciar o Servidor Zarafa no idioma correto. Esta configuração de idioma é usado para definir as opções padrão, como o nome da pasta pública no idioma correto.

A linguagem do WebAccess pode ser definido na tela de login. Isso pode ser configurado por login de usuário. Para as línguas não-Inglês o language-pack apropriado também precisa ser instalado.

Importante

Ao fazer o upgrade de uma versão anterior ZCP, reveja as configurações de idioma a partir do ZCP 7.0.0 o locale tem de ser definido em UTF-8.

In Debian distributions the follwing entry in /etc/apache2/envvars needs to be set to force the locale for Apache, else locale specific characters might not be displayed correctly in the WebAccess.

```
## Locale usado por alguns módulos, como mod_dav
# export LANG=C
## Incluir a seguinte linha para usar o locale do sistema:
. /etc/default/locale
```

4.4. Autenticação de Usuário

Outra opção de configuração importantes para o servidor Zarafa é o **user_plugi**. Esta configuração determina qual back-end é utilizado para gerenciamento de usuários e grupos. Há quatro opções: **db**, **unix** e **ldap** e **ldapms**.

Por padrão, o plugin **db** é usado, pois não requer qualquer configuração adicional. O plugin **ldap** é usado na maioria de setups maiores, uma vez que demonstra ser mais flexível e se integra muito bem em infra-estruturas existentes.

O plugin **ldapms** é necessário em configurações de ambientes multi-servidor Zarafa. Suporte multi-servidor só está disponível na edição Enterprise.

Mais informações sobre o gerenciamento de usuários podem ser encontradas em *Capítulo 8, Gerenciamento dos Usuários*.

For a comparison between the different plugins, see the table below:

Tabela 4.1. User plugin comparison

Feature	DB	Unix	LDAP	LDAPMS
Create/delete/ modify users	X	X	X	X
Set aliases	On MTA level	On MTA level	X	X

Feature	DB	Unix	LDAP	LDAPMS
Hide users	-	-	X	X
Sendas permissions	X	X	X	X
Sendas permissions of groups	-	-	X	X
Security Groups	X	X	X	X
Distribution groups	-	-	X	X
Hide groups	-	-	X	X
Dynamic groups	-	-	X	X
Contacts support	-	-	X	X
Multi-tenancy support	X	-	X	X
Addresslists support	-	-	X	X
Multi-server support	-	-	-	X

Importante

Although multi-tenancy is already possible when using the DB plugin, we strongly suggest using an LDAP backend when planning to host multiple tenants within one installation.

4.4.1. Autenticação via plugin DB

Este plugin usa o Zarafa banco de dados MySQL para armazenar informações de usuários e grupos. A ferramenta **`zarafa-admin`** pode ser usada para gerenciar usuários.

O plugin DB suporta apenas informações básicas de usuários e grupos. Para configurações mais avançadas, aconselha-se utilizar o plugin LDAP.

Mais informações sobre o gerenciamento de usuários com a ferramenta **`zarafa-admin`** podem ser encontradas em *Capítulo 8, Gerenciamento dos Usuários*.

4.4.2. Autenticação via plugin Unix

O plugin Unix é usada em um servidor que tem todos os suas informações de configuração de usuário no arquivo **`/etc/passwd`**. Informações dos grupos será lida a partir do **`/etc/group`**. As senhas são verificadas em com **`/etc/shadow`**, assim o processo do **`zarafa-server`** precisa ter acesso de leitura a este arquivo (este processo é normalmente executado como root, de modo geral que não é um problema).

Como os arquivos unix não contêm informações suficientes para Zarafa, existem algumas propriedades dos usuários que serão armazenados no banco de dados. Estas propriedades são o endereço de e-mail, configurações de cota e as configurações de administrador. A ferramenta **`zarafa-admin`** precisa ser usado para atualizar essas propriedades dos usuários. Todas as outras propriedades de usuários são feitas usando as ferramentas normais unix.

Um arquivo de configuração, **/etc/zarafa/unix.cfg**, existe para este plugin. O padrão definido por este arquivo geralmente são o suficiente, as linhas de comentários explicam cada opção. Neste arquivo de configuração do a faixa de **uid** que identifiquem usuários no servidor Zarafa precisa ser definido. O mesmo vale para os grupos.

Usuários não-ativos são configurados com um shell específico, padrão **/bin/false**. Esses usuários não podem logar, mas as caixas podem ser abertos por outros usuários. Um administrador deve configurar os direitos de acesso corretos para essas caixas.

Para uma visão geral de todas as opções de configuração do plugin de autenticação Unix, use:

```
man zarafa-unix.cfg
```

4.4.3. Autenticação via plugin LDAP

O plugin LDAP é utilizado para o acoplamento com qualquer servidor LDAP compatível com o Servidor Zarafa. Desta forma, todos os usuários, grupos e informações de associação pode ser recuperada "ao vivo" de um servidor LDAP.

O plugin LDAP suporta além de usuários, grupos e empresas padrão também seguintes tipos de objetos:

- **Contatos** — Contatos externos SMTP que podem ser usados como membros de listas de distribuição
- **Addresslists** — Subcategorias do Catálogo Global de Endereços, baseado em um filtro LDAP específico
- **Dynamic groups** — Grupos criados dinamicamente, baseado em um filtro LDAP específico. Por esta razão, o plugin LDAP é o plugin de usuário recomendado para o ZCP.

O Servidor do Zarafa precisa de duas diretivas de configuração no arquivo de configuração **server.cfg** para utilizar a porta auxiliar LDAP, a saber:

```
user_plugin = ldap
user_plugin_config = /etc/zarafa/ldap.cfg
```

Os padrões para o OpenLDAP e para o Active Directory podem ser encontrados no diretório **/usr/share/doc/zarafa/example-config**. Baseado nestes exemplos, o arquivo **/etc/zarafa/ldap.cfg** deve ser ajustado para configurar o plugin de autenticação do LDAP.

Para mais detalhes sobre a configuração do plugin LDAP com o OpenLDAP, ver *Seção 5.2, “Configurar a integração do ZCP OpenLDAP”* or *Seção 5.3, “Configurar a integração do Active Directory do ZCP”* para o Active Directory.

4.5. Auto-respondedor

O ZCP contém um auto-respondedor que pode ser utilizado quando um usuário está fora do escritório para responder automaticamente a todos os emails que chegam. O auto-respondedor irá automaticamente se ativar sempre que um email for entregue pelo **zarafa-dagent** para uma armazenagem que tem a opção ‘Out of Office’ ligada.

Os usuários podem manusear o auto-respondedor de suas próprias armazenagens, assim como das armazenagens para qual ele tem ao menos direitos de secretariado. Note que isto inclui pastas públicas. Favor verificar o Manual de Uauário para informações sobre como manusear tais definições.

Para evitar que o auto-respondedor repasse a mesma resposta (e.g. quando se envia uma resposta automática para uma resposta automática, a qual por sua vez envia uma resposta automática, etc), o auto-respondedor enviará somente uma mensagem de auto-resposta por dia para qualquer endereço de e-mail exclusivo. O auto-respondedor também não responderá em quem qualquer um dos seguintes casos:

- Enviando uma mensagem de fora-do-escritório para você mesmo.
- A mensagem original era para *mailer-daemon*, *postmaster* ou *root*.
- A mensagem original era de *mailer-daemon*, *postmaster* ou *root*.

Além disso, o atuo-respondedor está configurado por padrão para responder somente a emails nos quais o usuário foi explicitamente mencionado no cabeçalho "Para". isto significa que os emails que foram recebidos porque o usuário estava em cabeçalho "Cc" ou porque o usuário estava em um grupo de distribuição não serão respondidos.

A maior parte do comportamento pode ser configurada editando-se o arquivo **`/etc/zarafa/autorespond`**. Este arquivo contém as seguintes definições, as quais serão utilizadas para todas as mensagens de auto-resposta de todo o servidor:

```
AUTORESPOND_CC=0
```

Defina este valor para '1' para permitir a auto-resposta para mensagens nas quais o recebedor somente foi citado no cabeçado de 'Cc'.

```
AUTORESPOND_NORECIP=0
```

Defina esta valor para '1' para auto-responder a todas as mensagens, mesmo que o recebedor não esteja citado em qualquer cabeçalho (por exemplo, quando o email foi dirigido a uma lista de endereços ou grupo)

```
TIMELIMIT=$[24*60*60]
```

Define o número mínimo de segundos entre auto-respostas para o mesmo endereço de email

As seguintes definições normalmente não precisam ser modificadas:

```
SENDDB=${TMP:-/tmp}/zarafa-vacation-$USER.db
```

(arquivo que armazena a última data de envio por endereço de email)

```
SENDDBTMP=${TMP:-/tmp}/zarafa-vacation-$USER-$$.tmp
```

(arquivo temporário utilizado durante atualização do banco de dados)

```
SENDMAILCMD=/usr/sbin/sendmail
```

(comando utilizado para enviar mensagem atual de folga)

```
SENDMAILPARAMS="-t -f"
```

(parâmetros utilizados para enviar mensagem atual de folga)

Se um auto-respondedor alternativo for requisitado, favor se referir ao manual do **zarafa-dagent**, o qual descreve como utilizar um script alternativo (utilizando a opção **-a**).

4.6. Armazenando anexos fora do banco de dados

Since ZCP version 6.0 it is possible to save the attachments outside the database. ZCP 7.0.5 and higher will use the filesystem as default location for attachment storage.

Para a primeira vez das instalações, o método de armazenagem de anexos deve ser selecionado antes de inicial o servidor pela primeira vez, já que não é fácil trocar o método de armazenamento de anexos posteriormente.

Para mudar o local de armazenagem de anexos, edite a seguinte opção no **/etc/zarafa/server.cfg**.

```
attachment_storage = files
attachment_path = /var/lib/zarafa/attachments
```

For upgrades, a script exists that copies the attachments from the database to the file storage. This script can be found in **/usr/share/doc/zarafa**, and is named **db-convert-attachments-to-files**. This script can be used as follows:

```
db-convert-attachments-to-files <mysqluser> <mysqlpass> <mysqldb> <destination path> [delete]
```

Nota

The script can be executed while the zarafa-server process is running.

Somente é possível a conversão da armazenagem do banco de dados para a armazenagem de arquivos. A alavanca **<delete>** é opcional. Se este parâmetro for dado, os anexos também serão removidos do banco de dados. Tenha em mente que durante a conversão a armazenagem dos anexos no disco rígido irá dobrar. A quantidade de armazenagem no MySQL utilizada pelo ZCP pode ser encontrada com a seguinte instrução do MySQL:

```
mysql> use zarafa;
mysql> show table status;
```

Verifique a coluna **data_length** para a tabela grosseira. Isto contém o número de bytes necessários para a armazenagem do anexo.

Para selecionar este novo método de armazenagem, mude a opção **attachment_storage** no arquivo **server.cfg** e aponte a opção **attachment_path** para a pasta onde os anexos devem ser armazenados. Após mudar esta opção, o **zarafa-server** precisa ser reinicializado uma vez com o parâmetro **--ignore-attachment-storage-conflict**.

As vantagens dos anexos fora do banco de dados são:

- O MySQL não salva grandes bolhas de binários no banco de dados. Isto melhora o acesso geral para leitura e gravação.
- Os anexos não causação esvaziamento de cache do MySQL.

- you can use deduplication techniques (for example filesystem capabilities or through hardlinking) to further reduce hard disk space.

As desvantagens dos anexos fora do banco de dados são:

- A MySQLdump of the database is not enough for a full recovery.
- O armazenamento remoto de anexos necessita de um novo sistema, como uma pasta montada através do NFS ou Samba.

Importante

É muito importante, ao se escolher armazenar os anexos fora do banco de dados, atualizar a estratégia de backup de acordo.

Importante

When using NFS as storage backend for Attachment-Store or as WebAccess/WebApp TMP_PATH we recommend turning of NFS locking by using the *-o nolock* mount option as this potentially can cause severe performance penalties.

4.7. Conexões SSL e certificados

O Servidor do Zarafa é capaz de aceitar diretamente conexões SSL encriptadas.

Esta característica já pode estar disponível quando o servidor APACHE de HTTPS estiver definido para representar estas conexões no Servidor do Zarafa.

Contudo, ter conexões SSL nativas ao servidor tem uma vantagem interessante: os componentes do Zarafa rodando além da hospedagem local podem acessar utilizando seu certificado SSL.

Esta seção descreverá como configurar certificados para para adicionar conexões SSL nativas ao Zarafa.

Primeiro, criaremos o diretório que conterá o certificado e definiremos as permissões, já que ele contém nossa chave secreta.

```
mkdir /etc/zarafa/ssl
chmod 700 /etc/zarafa/ssl
```

Se o Zarafa foi rodado como outro usuário, como descrito na seção Rodando como um usuário que não é da raiz, não se esqueça também de mudar o proprietário da pasta.

Agora estamos prontos para criar um *Certificate Authority* (CA). Este CA será utilizado para criar o certificado do servidor e acessá-lo. Nós cedemos um script **`ssl-certificates.sh`** no diretório **`/usr/share/doc/zarafa`**, o qual utiliza o comando **`openssl`** e o script **`CA.pl`** do OpenSSL. Dependendo da distribuição utilizada, este script pode ser instalado em diferentes diretórios. O script tentará encontrá-lo sozinho. Se ele não for encontrado, pode ser que o OpenSSL não esteja instalado, ou que o script esteja em um local desconhecido, e a localização do script precisará ser informada manualmente. Normalmente, o script **`ssl-certificates.sh`** pode ser rodado sem problemas.

```
cd /etc/zarafa/ssl
```

```
sh /usr/share/doc/zarafa/ssl-certificates.sh server
```

O servidor de parâmetro está adicionado, então o nome do novo certificado será chamado **`server.pem`**. Quando o CA não for encontrado no diretório padrão **`./demoCA`**, ele precisa ser criado. Pressionando-se enter, a criação de um novo CA é iniciada.

Entre com uma senha (palavra-passe) quando for solicitado. Esta é a senha utilizada posteriormente para assinar requisições de certificado. Agora as informações do certificado devem ser informadas. Não deixa o campo **`Common Name`** vazio, caso contrário a criação falhará.

Agora que temos um CA, podemos criar certificados *self-signed*. O script **`ssl-certificates.sh`** continuará automaticamente com este passo. Entre com a senha para a requisição e entre com os detalhes do certificado. Alguns detalhes precisam ser diferentes daqueles digitados quando o CA foi criado. Pelo menos o campo **`Organizational Unit Name`** precisa ser diferente. A senha de desafio ao final pode ser deixada em branco.

Este passo criou uma Requisição de Certificado, o qual precisa ser assinado pelo CA que foi criado no primeiro passo do script. Digite a senha do CA novamente quando for solicitado. Os detalhes do certificado serão mostrados e será pedida a aceitação. Aceite o certificado.

No último passo, a chave pública deste certificado será oferecida. Já que o certificado do servidor acabou de ser criado, a chave pública deste certificado não é necessária.

Agora que o certificado do CA e o certificado do servidor foram criado, o SSL pode ser habilitado no arquivo **`server.cfg`**, o qual normalmente está desabilitado. A porta **`237`** é destinada a conexões SSL. Esta porta pode ser mudada caso seja necessário.

```
server_ssl_enabled = yes
server_ssl_port = 237
```

O certificado do CA deve ser definido no arquivo **`server_ssl_ca_file`**. O certificado do servidor e senha precisam ser definidos nas opções **`server_ssl_cert_file`** e **`server_ssl_cert_pass`**.

```
server_ssl_ca_file = /etc/zarafa/ssl/demoCA/cacert.pem
server_ssl_key_file = /etc/zarafa/ssl/server.pem
server_ssl_key_pass = <password>
```

Reinicie o processo do **`zarafa-server`**, e agora é possível conectar diretamente à porta SSL. Crie um novo perfil do Outlook e marque a opção de conexão SSL. Defina a porta para **`237`**. A conexão ao servidor agora foi encriptada.

4.8. Configurar o Gerenciador de Licensa

Nota

With the ZCP opensource edition the License Manager is not needed.

The License Manager (**`zarafa-licensed`**) expects **`/etc/zarafa/license`** to contain a file named **`base`** which simply holds the license key. To install a subscription key, use the following command:

```
mkdir -p /etc/zarafa/license
```

```
echo <subscription key> > /etc/zarafa/license/base
```

<subscription key> should be replaced with a valid subscription key obtained from Zarafa or one of its partners.

Nota

The subscription key consists only of numbers and capital letters.

If an extra CAL (Client Access License) is also available, the key can be added with:

```
echo 'CAL key' > /etc/zarafa/license/cal1
```

Se mais de um CAL estiver disponível, favor instalar um CAL por arquivo no diretório da licença. O nome do arquivo do CAL não é importante. Subpastas não são permitidas na pasta **/etc/zarafa/license**.

4.9. Configurando o Spooler do Zarafa

O Zarafa-spooler envia emails da fila de saída global para um servidor SMTP, o qual envia os emails ao endereço correto.

Wuando uma mensagem de email é enviada do Outlook ou WebAccess, a mensagem é depositada na pasta de Saída e uma mensagem de envio é enviada para o servidor do Zarafa. O servidor avisa o spooler do Zarafa para enviar o email para o servidor SMTP. O spooler agora começará a converter a mensagem em uma mensagem normal de email. Quando a conversão estiver completa, uma conexão ao servidor SMTP disponibilizado é criada e o email é enviado para o servidor SMTP.

O spooler enviará o email e, após o email ser enviado, moverá o email automaticamente para a pasta de Itens Enviados do usuário.

Se, a qualquer hora, ocorrer um erro, o usuário será notificado com uma mensagem de 'Não-enviável'. A mensagem conterá uma descrição do erro encontrado. Geralmente, o usuário pode tentar reenviar a mensagem.

Nota

Tanto emails externos quanto internos serão enviados via MTA.

4.9.1. Configuração

O spooler está configurado da mesma maneira que o servidor. As opções no arquivo de configuração do spooler são o nome ou o endereço IP do servidor SMTP, onde encontrar o servidor do Zarafa e opções de acesso.

```
smtp_server
```

O nome ou endereço de IP do servidor SMTP, o qual enviará o email para o destino. Este servidor pode também ser dado como um argumento ao se iniciar o spooler.

```
server_socket
```

O socket do UNIX do servidor do Zarafa. O spooler utilizará este socket para criar uma conexão com o o servidor. Este valor deve ser o mesmo que o definido no arquivo de configuração do servidor. O valor padrão é **`/var/run/zarafa`**.

```
[logging]
```

O spooler tem as mesmas opões de configuração que o servidor para configurar as opções de acesso.

Para uma visão geral de todas as opções de configuração do **`zarafa-spooler`**, utilize:

```
man zarafa-spooler.cfg
```

4.10. Configurar o Zarafa Caldav

O Zarafa Caldav é um componente que permite aos usuários visualizar os dados de seu calendário por clientes que suportam o padrão Caldav, como o Sunbird ou Evolution. Este componente se conecta com o Servidor do Zarafa utilizando o MAPI sobre o HTTP.

O Caldav e o iCal ativam e recuperam calendários completos. O Sunbird e outros clientes suportam tanto a recuperação quanto a ativação, enquanto o Evolution suporta somente a recuperação de calendários.

O componente Zarafa Caldav pode ser configurado utilizando-se um arquivo de configuração no mesmo estilo que o Servidor do Zarafa. Ele suporta tanto conexões simples quanto conexões seguras tipo SSL/TLS . Para aumentar a segurança, é recomendado habilitar exclusivamente a conectividade do Caldav seguro.

As opções de configuração são:

```
server_bind
```

O endereço de IP para se vincular. **`0.0.0.0`** para qualquer endereço. Valor padrão: **`0.0.0.0`**

```
ical_enable
```

Habilite o serviço simples com o valor **`yes`**. Valor padrão: **`yes`**

```
ical_port
```

O serviço simples responderá nesta porta para conexões que chegam. Valor padrão: **`8080`**

```
icals_enable
```

Habilite o serviço seguro com o valor **`yes`**. Valor padrão: **`no`**

```
icals_port
```

O serviço seguro responderá nesta porta para conexões que chegam. Valor padrão: **`8443`**

```
server_socket
```

O endereço http do Servidor Zarafa. Valor padrão: **`http://localhost:236/zarafa`**

Importante

Não é aconselhável especificar o socket UNIX aqui. Na configuração padrão, o Zarafa Caldav será então guardado pelo **`zarafa-server`** (como definido nas definição de configuração **`local_admin_users`**). A menos que o Zarafa Caldav seja especificado para rodar como um usuário não confiável, ele sempre autentica os usuários mesmo que eles não forneçam credenciais, ou forneçam credenciais erradas!

```
ssl_private_key_file
```

O arquivo que contém a chave privada utilizada para encriptar as conexões SSL. O caminho absoluto para o arquivo deve ser utilizado. Valor padrão: **`/etc/zarafa/privkey.pem`**

```
ssl_certificate_file
```

O arquivo que contém o certificado para o servidor. O caminho absoluto para o arquivo deve ser utilizado. Valor padrão: **`/etc/zarafa/cert.pem`**

```
ssl_verify_client
```

Permite a verificação do certificado do cliente com o valor **`yes`**. Valor padrão: **no**

```
ssl_verify_file / ssl_verify_path
```

O arquivo ou caminho para os arquivos com o qual é feita a verificação dos certificados dos clientes. O caminho absoluto deve ser utilizado em ambas as opções (não há padrão).

```
[logging]
```

O componente Caldav tem as mesmas opções de configuração que o servidor para configurar opções de acesso.

4.10.1. SSL/TLS

Como já mencionado, o componente Zarafa Caldav suporta SSL/TLS, para isto a livraria OpenSSL é utilizada.

O arquivo da chave privada (para encriptação) e do certificado (para autenticação) pode ser definido no arquivo de configuração com **`ssl_private_key_file`** e **`ssl_certificate_file`**.

O componente Zarafa Caldav pode ainda autenticar os clientes do calendário que tentam se conectar a ele verificando os certificados do cliente, utilizando um ou mais arquivos de verificação. Isto pode ser definido com **`ssl_verify_client`**, **`ssl_verify_file`** e **`ssl_verify_path`**. Os certificados podem ser auto-assinados ou assinados por uma autoridade confiável do certificado.

O seguinte comando gera uma chave RSA de 2048 bytes:

```
openssl genrsa -out /etc/zarafa/privkey.pem 2048
```

Este comando cria um certificado de teste auto-assinado válido por 3 anos:

```
openssl req -new -x509 -key /etc/zarafa/privkey.pem -out /etc/zarafa/cert.pem -days 1095
```

Se um arquivo **.cer** e um arquivo **.key** já estiverem presentes, você pode criar um arquivo **.pem** à partir deles utilizando o seguinte comando:

```
cat my_server.key > my_server_combined.pem
cat my_server.cer >> my_server_combined.pem
```

E então utilizar o arquivo **my_server_combined.pem** para **ssl_private_key_file** ou **ssl_certificate_file**.Favor ter certeza que o arquivo **.key** seja processado, e então o arquivo **.cer**.

4.11. Configurando o Zarafa Gateway (IMAP and POP3)

O Zarafa IMAP & POP3 Gateway permite aos usuários visualizarem as mensagens armazenadas no Servidor do Zarafa com um cliente IMAP or POP3. Por exemplo, o Mozilla Thunderbird ou um dispositivo móvel com o Microsoft Pocket Outlook. Para acessar os dados do usuário, o próprio Zarafa Gateway se conecta ao Servidor do Zarafa com o MAPI.

O POP3 somente pode recuperar a mensagem na Caixa de Entrada à partir do servidor. O IMAP, por outro lado, mostra todas as pastas que podem conter mensagens, como Rascunhos e Itens Deletado. Todas as subpastas são mostrada como no Microsoft Office Outlook ou no Zarafa WebAccess.

O Zarafa IMAP & POP3 Gateway pode ser configurado com um arquivo de configuração. As opções de configuração sâo:

```
server_bind
```

O endereço de IP para se vincular. **0.0.0.0** para qualquer endereço. Valor padrão: **0.0.0.0**

```
imap_enable
```

Habilite o serviço IMAP com **yes**. Valor padrão: **yes**

```
imap_port
```

O serviço de segurança do IMAP atenderá às conexões nesta porta. Valor padrão: **143**

```
imaps_enable
```

Habilite o serviço de segurança do IMAP com o valor **yes**. Valor padrão: **no**

```
imaps_port
```

O serviço de segurança do IMAP atenderá às conexões nesta porta. Valor padrão: **993**

```
pop3_enable
```

Habilite o serviço POP3 com o valor **yes**. Valor padrão: **yes**

```
pop3_port
```

O serviço POP3 atenderá às conexões nesta porta. Valor padrão: **110**

```
pop3s_enable
```

Habilite o serviço POP3 com o valor **yes**. Valor padrão: **no**

```
pop3s_port
```

O serviço POP3 atenderá às conexões nesta porta. Valor padrão: **995**

```
imap_only_mailfolders
```

Habilite somente pastas de mensagens para serem mostradas com o valor **yes**. Valor padrão: **yes**

```
server_socket
```

O endereço http do servidor do Zarafa. Valor padrão: **http://localhost:236/zarafa**

Importante

Não é aconselhável especificar o socket UNIX aqui. Na configuração padrão, a Porta do Zarafa será então creditada pelo **zarafa-server** (como definido em suas definições de configuração **local_admin_users**). A menos que a Porta do Zarafa seja especificada para rodar como um usuário não confiável, ela sempre autenticará os usuários mesmo se eles informarem credenciais erradas ou mesmo não informar credencial alguma.

```
ssl_private_key_file
```

O arquivo que contém a chave privada utilizada para encriptar as conexões SSL. O caminho absoluto para o arquivo deve ser utilizado. Valor padrão: **/etc/zarafa/privkey.pem**

```
ssl_certificate_file
```

O arquivo que contém o certificado para o servidor. O caminho absoluto para o arquivo deve ser utilizado. Valor padrão: **/etc/zarafa/cert.pem**

```
ssl_verify_client
```

Permite a verificação do certificado do cliente com o valor **yes**. Valor padrão: **no**

```
ssl_verify_file / ssl_verify_path
```

O arquivo ou caminho para os arquivos com o qual é feita a verificação dos certificados dos clientes. O caminho absoluto deve ser utilizado em ambas as opções (não há padrão).

```
[logging]
```

A porta tem as mesmas opções de configuração que o servidor para configurar opções de acesso.

4.11.1. SSL/TLS

A Porta do Zarafa suporta o SSL/TLS utilizando a livradia do OpenSSL. Para mais informações, ver *Seção 4.10.1, "SSL/TLS"*, já que as opções são exatamente as mesmas para este dois componentes.

4.11.2. Informações importantes

O IMAP e o POP3 são disponibilizados para compatibilidade inversa e não disponibilizará a mesma experiência como clientes que suportam MAPI (Microsoft Outlook or our WebAccess). Os clientes IMAP/POP3 utilizam estes protocolos somente para mensagens (sendo que o MAPI cuida das mensagens, calendário e contatos).

Definir a mensagem Out of Office não é possível com os clientes IMAP ou POP3.

Regras definidas no Microsoft Outlook não funcionam ao se utilizar o Zarafa IMAP & POP3 Gateway. Alguns clientes podem definir regras, mas estas regras não estão relacionadas às regras definidas por um cliente MAPI habilitado.

Deletar uma mensagem utilizando o IMAP marcará a mensagem para deleção. Isto não é mostrado no Microsoft Outlook e no Zarafa WebAccess. A mensagem será deletada quando o cliente esvaziar a pasta. Alguns cliente permitem esvaziar pastas manualmente e alguns possuem definições para esvaziar uma pasta. Outros clientes esvaziam a pasta automaticamente quando uma mensagem é deletada.

Mover a mensagem para uma pasta diferente com o IMAP é feito copiando a mensagem para a nova pasta e marcando a mensagem original para deleção. Enquanto a mensagem original não foi excluída da pasta, a mensagem será mostrada em ambas as pastas citadas acima.

4.12. Configurando o Gerenciador de Quota do Zarafa

Os usuários podem coletar muitos emails. enquanto o espaço do disco pode ser limitado. O Gerenciador de Quota do Zarafa pode ser utilizado para definir quotas de espaço específico de usuários ou de toda a rede. O Gerenciador de Quota do Zarafa reconhece três níveis: aviso, maleável e rígido. Quando um dos níveis é alcançado, o usuário recebe um email com os tamanhos da quota e qual nível de quota foi alcançado.

As definições de quota podem ser configuradas em toda a rede em **`server.cfg`** ou por usuário via plugin do usuário.

Quando um usuário alcança o nível de aviso da quota, o usuário recebe um email com um aviso e informações da quota. Quando o usuário alcança o limite da quota, ele não é capaz de enviar email até que o tamanho da armazenagem seja reduzido. Quando o limite rígido da quota é alcançado, o email também não pode mais ser entregue para o usuário.

4.12.1. Configurando a quota de todo do servidor

A quota de todo o servidor pode ser configurara no seguinte arquivo no servidor:

```
quota_warn = 100
quota_soft = 150
quota_hard = 200
```

Os valores estão todos em megabytes. Estes valores serão utilizados por todos os usuários presentes no servidor.Quando os valores forem definidos para **`0`**, aquele nível de quota em particular será desabilitado.

4.12.2. Definir a quota por usuário

Utilizando-se a ferramenta **zarafa-admin**, a quota de usuário pode ser definida para um usuário específico. Exemplo:

Defina a quota do usuário John da seguinte maneira: Nível de aviso para 80 Mb, nível maleável para 90 Mb e nível rígido para 100 Mb.

```
zarafa-admin   -u john   --qd 0   --qw 80   --qs 90   --qh 100
```

Nota

Definir a quota do *Capítulo 8, Gerenciamento dos Usuários* **in** não funciona com o LDAP. Com o LDAP, as propriedades são armazenadas no servidor LDAP por usuário. Para mais informações, ver **zarafa-admin**.

4.12.3. Monitorando quando a quota excede

O programa zarafa-monitor checa a toda hora (por padrão) por usuário que excederam seu nível de quota e envia emails para um usuário quando o alarma ou limite maleável de quota é excedido. Definições globais de quota podem ser feitas na configuração do servidor. Níveis específicos de usuários podem ser definidos via **zarafa-admin** quando se utiliza os plugins DB ou Unix, ou editando os valores de LDAP como descrito na seção de Gerenciamento de Usuário.

Para iniciar o zarafa-monitor, utilize:

```
/etc/init.d/zarafa-indexer start
```

ou

```
zarafa-monitor -c /etc/zarafa/monitor.cfg
```

O **zarafa-monitor** fará o daemon do processo, para que o prompt quase imediatamente retorne. Utilize **-F** para iniciar no primeiro plano. Masi informações sobre as opções de configuração podem ser encontradas na página do manual:

```
man zarafa-monitor.cfg
```

4.12.4. Modelos de alerta de quota

Ao se trabalhar com o zarafa-monitor, é possível modificar o conteúdo do email que será enviado quando o usuário ou companhia exceder sua quota. Para cada nível de quota que um modelo separado de quota possa ser especificado, estes podem ser configurados com as opções seguintes:

- **userquota_warning_template**
- **companyquota_warning_template**

Por padrão, os modelos são armazenados no **/etc/zarafa/quotamail**, em cada um desses modepos certas variáveis são dadas, as quais serão substituídas pelo valor real antes que o email seja enviado:

- **ZARAFA_QUOTA_NAME** - O nome do usuário ou companhia que excedeu sua quota
- **ZARAFA_QUOTA_COMPANY** - O nome da companhia à qual o usuário pertence
- **ZARAFA_QUOTA_STORE_SIZE** - Quando um usuário excede sua quota, esta variável contém o tamanho total do armazenamento do usuário. Quando uma companhia excede sua quota esta variável contém o tamanho total de todas as armazenagens, incluindo a armazenagem pública dentro do espaço da companhia.
- **ZARAFA_QUOTA_WARN_SIZE** - O aviso de limite da quota para o usuário ou companhia.

Nota

Variáveis contendo um tamanho sempre incluem a unidade de tamanho (**B**,**KB**,**MB**,**GB**) como parte da variável.

4.13. Configure Zarafa Search

The **zarafa-search** service, introduced in ZCP 7.10, offers full text searching capabilities for the Zarafa Server. The service will continuously index all mails, and optionally their attachments, of a single zarafa-server instance. Each zarafa-server instance in a multi-server setup needs its own zarafa-search service.

When searching for a particular mail, the required time to find the requested emails will be seriously reduced. When attachment indexing is enabled, it is even possible to index the contents of attached files (for common file types that contain text).

4.13.1. Enabling the search service

Para iniciar o serviço de indexação, execute o seguinte comando:

```
/etc/init.d/zarafa-search start
```

Para habilitar a busca no texto completo, edite o arquivo de configuração **/etc/zarafa/server.cfg**:

```
search_enabled = yes
```

During searching the zarafa-server will connect with the **zarafa-search** service. To set the connection path change the following configuration option:

```
search_socket = file://var/run/zarafa-search
```

4.13.2. Search configuration

During indexing, the index file for each store is stored on the harddisk. The location of these files can be configured in **/etc/zarafa/search.cfg**:

```
index_path = /var/lib/zarafa/index/
```

In this folder a file will be created for each store located on the Zarafa server node. A state file will also be present to remember where the indexing process has left upon restart.

Importante

The files within this index path should not be touched while the indexer is running. If a store must be re-indexed, the **zarafa-search** must be stopped first before deleting the file for that particular store.

The **zarafa-search** service uses streaming synchronization offered by the zarafa-server for fast indexing of messages. To enable streaming, ensure that the following configuration option is enabled in the zarafa-server config:

```
enable_enhanced_ics = yes
```

This option is enabled by default, and normally there is no reason to disable it.

4.13.3. Anexos

Optionally the contents of attachments can be indexed as well. When this is enabled, searching for a message will also search through the attachment text as well.

To enable indexing of attachments can be done in **/etc/zarafa/search.cfg**:

```
index_attachments = yes
```

A indexação de anexos é feita através da análise do anexo em texto simples e indexando o texto no índice principal para o email. O tempo necessário para analisar e indexar um anexo em particular depende do tamanho do anexo. Para prevenir que grandes anexos aumentem o tempo de latência da indexação total, a opção de configuração **index_attachment_max_size** pode ser utilizada para prevenir que grandes anexos sejam anexados. O valor dado a este configuração precisa ser definido em kilobytes.

To parse the attachments to plain text a separate configuration script must be provided. By default this script is installed to **/etc/zarafa/searchscripts/attachments_parser** but the exact location can be configured using the configuration option **index_attachment_parser**.

The default script **attachments_parser** will use the file **attachments_parser.db** to decide how the attachment should be parsed to plain text. Within this file is a list containing the command to parse each attachment type to plain text. This file can be edited to control the way attachments are parsed and to add or remove support for particular attachment types.

O esquema de cada linha é como mostrado a seguir:

```
<mime-type>;<extension> `<command>`
```

Cada linha pode ter tantos tipos de mime e extensões quanto necessários, cada tipo de mime e extensão precisa ser separado usando-se semi-colunas. O comando deve ler o **/dev/stdin** para os dados do anexo e precisa retornar o texto simples através do **/dev/stdout**. Algumas ferramentas não podem analisar os dados de anexos de um stream e necessitam que os dados sejam fornecidos como arquivo. Para armazenar o anexo em um arquivo temporário, o script **zmktemp** pode ser utilizado. Esse script gravará todos os dados de anexo em um arquivo temporário e marcará o local do arquivo em **/dev/stdout**.

Para anexos que não podem ser analisados (como imagens, por exemplo), o comando **echo -n** pode ser utilizado.

Após editar o comando, é aconselhável testá-lo para ver se a resposta é a desejada. Testar o comando pode ser feito executando o seguinte comando na linha de comando:

```
cat <attachment> | <command>
```

The resources used by the **attachments_parser** during the parsing of a single attachment can be restricted by limiting the total memory and CPU time usage. To control the maximum amount of memory the script can use is controlled by the configuration option **index_attachment_parser_max_memory**. By default this value is set to **0**, to disable any memory consumption restriction. If a restriction should be applied, the maximum number of bytes should be provided. The best restriction size depends on the maximum attachment size which can be provided to the script (configured using **index_attachment_max_size**) and the 3rd party tools used to parse the attachments.

To prevent the script to take too much time, the configuration option **index_attachment_parser_max_cputime** can be used. By default this value is set to **0**, to disable any CPU time restriction. If a restriction should be applied, the maximum number of seconds should be provided. The best restriction depends on the 3rd party tools used to parse the attachments.

Se qualquer um destes limites for excedido, o script será cancelado e o anexo não será indexado.

4.14. Configure Zarafa WebAccess

The Zarafa WebAccess includes a configuration file, which allows the Administrators for example to enable server side spell correction and set default values for language and themes. This configuration can be found in **/etc/zarafa/webaccess-ajax/config.php** and is also present (as a symlink) in **/usr/share/zarafa-webaccess**.

4.15. Configure Zarafa WebApp

The Zarafa WbApp includes a configuration file, which allows the Administrators for example to define a default language for the WebApp, limit the amount of available languages or disable certain plugins. This configuration can be found in **/etc/zarafa/webapp/config.php** and is also present (as a symlink) in **/usr/share/zarafa-webapp**. In addition this folder also contains configuration files for some of the distributed WebApp plugins like the chat integration or the link to the WebApp manual.

With Version 1.4 of the Zarafa WebApp two new options were introduced to globally define the time frame for free/busy information. *FREEBUSY_LOAD_START_OFFSET* defines the amount of days for which old appointments are kept in the free/busy database and *FREEBUSY_LOAD_END_OFFSET* defines the amount of days for which upcoming appointments are stored in the free/busy database. By default the information for the last seven and the upcoming 90 days are saved.

Configurar componentes de terceiros

5.1. Configurar o servidor web

Normalmente, o pacote Zarafa irá configurar o PHP no sistema automaticamente. Na maioria das situações, este capítulo pode ser ignorado e continuado com *Seção 5.1.2, "Configurar o Apache"*.

5.1.1. Configurar PHP

O PHP é necessário para se usar o WebAccess. A extensão PHP-MAPI está instalado no diretório padrão de distribuição:

- Red Hat Enterprise Linux: **`/usr/lib/php5/modules/`**
- SLES: **`/usr/lib/php/extensions/`**
- Debian: **`/usr/lib/php5/20060613/`**
- Ubuntu: **`/usr/lib/php5/20060613/`**

Se um diretório diferente foi selecionado para extensões PHP, mova os arquivos **`mapi.so*`** para este local, por exemplo:

```
mv /usr/lib/php/mapi.so* \
      /usr/local/lib/php/
```

Para achar o local de extensões PHP, utilize o seguinte comando:

```
php-config --extension-dir
```

Após a extensão PHP estar no diretório correto, adicione-o ao arquivo de configuração **`php.ini`**. Adicione a seguinte linha ao php.ini se ela ainda não existe:

```
extension = mapi.so
```

Os locais comuns para o arquivo **`php.ini`** são:

```
/etc/php.ini
```

```
/etc/php5/apache2/php.ini
```

Com a função **`phpinfo()`** é possível verificar se o módulo será carregado corretamente. Pesquise pela parte do 'MAPI' para verificar o módulo. O phpinfo também pode ser visualizado executando **`php -i`** na linha de comando se o **`php cli`** estiver instalado.

5.1.2. Configurar o Apache

Para carregar corretamente a recém-adicionada extensão **`mapi.so`**, o servidor web precisa ser reiniciado. O exemplo seguinte mostra como reiniciar o Apache2:

```
/etc/init.d/apache2 restart
```

ou

```
/etc/init.d/httpd restart
```

5.1.2.1. For WebAccess

Os arquivos do site da Internet por padrão são instalado no diretório do WebAccess. Verifique se a página de acesso do cliente web pode ser aberta navegando para o endereço correto:

```
http://<ip-address server>/webaccess/
```

Se a página de login não for mostrada, o servidor web precisa ser configurado para deixá-lo acessar o diretório correto. O seguinte exemplo mostra uma configuração para o Apache2:

```
Alias /webaccess /usr/share/zarafa-webaccess/
<Directory /usr/share/zarafa-webaccess/>
        AllowOverride None
        Order allow,deny
        Allow from all
</Directory>
```

Certifique-se de ter digitado o diretório correto que contém os arquivos de PHP WebAccess. O seguinte comando irá dizer ao apache2 para reler seu arquivo de configuração:

```
/etc/init.d/apache2 reload
```

O WebAccess agora deve estar visível. Se ele ainda não aparecer, consulte *Seção 2.3, "Solucionando problemas de instalação"* para mais informações.

5.1.2.2. For WebApp

The website files are by default installed in the WebApp directory. Make sure the webclient's login page can be opened by browsing to the correct url:

```
http://<ip-address server>/webapp/
```

Se a página de login não for mostrada, o servidor web precisa ser configurado para deixá-lo acessar o diretório correto. O seguinte exemplo mostra uma configuração para o Apache2:

```
Alias /webapp /usr/share/zarafa-webapp/
<Directory /usr/share/zarafa-webapp/>
        AllowOverride None
        Order allow,deny
        Allow from all
</Directory>
```

Make sure the correct directory holding the PHP WebApp files is typed. The following command will tell apache2 to reread its config file:

```
/etc/init.d/apache2 reload
```

The WebApp should now be visible. If it still does not show up, please see *Seção 2.3, "Solucionando problemas de instalação"* for more information.

When leaving the configuration at this point, Apache will request the browsers to cache all files as long as they see fit. This may mean that users are still seeing the old interface while the WebApp package on the server has been upgraded. To fix this, the package comes with an example configuration that includes instructions to the browsers on how long WebApp resources may be kept around.

Using this, we are saying that Javascript and CSS files need to be checked against the server versions very often, but Apache can serve these files very quickly from the filesystem. For images, we allow the clients to keep using them for a much longer period (2 months). For this, we use the FileETag setting of Apache to generate a unique identifier for each served static file. To use this, the Apache modules mod_expires and mod_headers need to be loaded.

The following can be included in the Apache configuration within the <Directory> directive as described above:

```
FileETag All

ExpiresActive On

<filesMatch "\.(jpg|gif|png)$">
    ExpiresDefault "access plus 2 months"
    Header append Cache-Control "public"
</filesMatch>

<FilesMatch "\.(js|css)$">
    ExpiresDefault "access plus 2 weeks"
    Header append Cache-Control "no-cache, must-revalidate"
</FilesMatch>

<filesMatch "\.(php)$">
    ExpiresActive Off
    Header set Cache-Control "private, no-cache, no-store, proxy-revalidate, no-transform"
    Header set Pragma "no-cache"
</filesMatch>
```

The example zarafa-webapp.conf that comes with the WebApp package contains a more extensive version of this. Especially if you have a lot of users with Internet Explorer, this will be better suited for you than the terse example above.

5.1.3. Apache como um proxy HTTP

The transmitted data between the client and server is compressed XML, wrapped in HTTP packets. The use of HTTP allows packets to be forwarded by a proxy (or a webserver with built-in proxy functionality, for example Apache version 2).

The following lines are an example of how Apache can be configured to forward incoming connections on port **80** to the Zarafa Server on port **236**. In case the Apache server also accepts HTTPS connections, the proxyed connections can also be encrypted. The **proxy** and **proxy_html** modules of Apache need to be loaded for this to work(for example with *a2enmod proxy proxy_http*).

```
<IfModule mod_proxy.c>
       ProxyPass /zarafa http://127.0.0.1:236/
       ProxyPassReverse /zarafa http://127.0.0.1:236/
</IfModule>
```

Isto significa que URLs que começam com **/zarafa** serão encaminhadas ao **localhost** na porta **236**, onde o servidor Zarafa presta atenção nas conexões que chegam. Essas linhas podem ser colocadas no nível global, ou dentro de uma declaração VirtualHost.

Nota

Keep in mind that using a HTTP proxy will create some performance overhead on your system, so it is not recommended to use this for larger setups.

5.2. Configurar a integração do ZCP OpenLDAP

In several network environments OpenLDAP is used to keep track of various bits of information. Mast notably: users and their credentials. Zarafa integrates with LDAP server and supports the use of OpenLDAP in particular.

As Zarafa doesn't bundle a LDAP server, this has to be setup separately uf there is not yet a server available in the environment. Please read the documentation of the used Linux distribution on how to setup an OpenLDAP server. Zarafa provides an example LDIF file in *Capítulo 13, Appendix C: Example LDIF*.

Connections to the OpenLDAP server run over port **389** or **636** (TLS/SSL). For best speed and reliability it is always recommended to install an OpenLDAP server on the same host as the Zarafa Server itself. This local server can then be setup to replicate the main LDAP server. Besides performance improvements this also allows the Zarafa Server to function even when the main LDAP server is not available.

In the following paragraphs the needed steps are provided to connect Zarafa to an existing OpenLDAP tree. The OpenLDAP configuration is usually located in **`/etc`**, depending on the used distribution the exact location may vary.

For the official supported distributions the locations are:

- RHEL: **`/etc/openldap`**
- SLES: **`/etc/openldap`**
- Debian & Ubuntu: **`/etc/ldap`**

Through out this guide we will use: **`/etc/openldap`**

5.2.1. Configuring OpenLDAP to use the Zarafa schema

To make managing Zarafa user easier it is recommended to import the Zarafa LDAP schema. The schema can be imported by issuing the following command:

```
zcat /usr/share/doc/zarafa/zarafa.ldif.gz | ldapadd -H ldapi:/// -Y EXTERNAL
```

5.2.2. LDAP indices

Indexing entries is a way to improve performance performing a filtered search on the LDAP directory. The following table shows the most important attributes to index and the type of index that should be implemented.

Tabela 5.1. LDAP indices

Attribute name	Type
cn	pres,eq,sub
gidNumber	pres,eq

Attribute name	Type
mail	pres,eq,sub
memberUid	pres,eq
objectClass	pres,eq
ou	pres,eq
sn	pres,eq,sub
uid	pres,eq
uidNumber	pres,eq
zarafaAliases	pres,eq,sub
zarafaAccount	pres,eq
zarafaSendAsPrivilege	preq,eq
zarafaViewPrivilege	pres,eq

Depending on the Zarafa ldap configuration the attributes may be different.

Please check the OpenLDAP or syslog logfiles for further attributes which are not yet indexed and could be included to increase performance. Check below for an example log message:

```
May 13 14:37:17 zarafa slapd[4507]: <= bdb_equality_candidates: (mail) not indexed
```

When using the cn=config backend the following ldif file can be used to add the given attributes to the index of OpenLDAP:

```
dn: olcDatabase={1}hdb,cn=config
changetype: modify
add: olcDbIndex
olcDbIndex: cn eq
olcDbIndex: gidNumber eq
olcDbIndex: mail eq
olcDbIndex: memberUid eq
olcDbIndex: ou eq
olcDbIndex: uid eq
olcDbIndex: uidNumber eq
olcDbIndex: uniqueMember eq
olcDbIndex: zarafaAccount eq
olcDbIndex: zarafaAliases eq
olcDbIndex: zarafaViewPrivilege eq
```

To import this the following comman can be used:

```
cat optimize-index.ldif | ldapmodify -Y EXTERNAL -H ldapi:///
```

5.2.3. Configurando o ZCP para OpenLDAP

Para integrar o ZCP com um servidor OpenLDAP, altere a opção seguinte no arquivo de configuração **`ldap.cfg`**:

Especifique na opção **`ldap_host`** o nome do servidor ou endereço IP do servidor LDAP.

```
ldap_host = localhost
```

Por padrão, o protocolo LDAP simples será utilizado. Para a configuração de LDAP seguro, altere as configurações seguintes. As informações sobre como configurar OpenLDAP com certificados SSL podem ser encontrados no site *http://wiki.zarafa.com*.

```
ldap_port = 389
ldap_protocol = ldap
```

To connect ZCP to multiple LDAP servers, use the following setting:

```
ldap_uri = ldap://ldapserver1:389 ldap://ldapserver2:389
```

The different ldap uri's should be seperated by a whitespace. When using the **`ldap_uri`** option, the options **`ldap_host`**, **`ldap_port`** and **`ldap_protocol`** are ignored.

O servidor Zarafa só vai ler do servidor OpenLDAP. O usuário de ligação especificado deve ter pelo menos acesso de leitura no servidor LDAP.

```
ldap_bind_user = cn=Manager,dc=example,dc=com
ldap_bind_passwd = secret
ldap_authentication_method = bind
```

O método de autenticação pode ser definido para **`password`**, assim o servidor Zarafa irá comparar a senha criptografada do servidor LDAP com a senha criptografada que o usuário inseriu durante o login.

Para este método, o usuário bind especificado tem que ser um usuário administrativo no OpenLDAP e ter acesso de leitura no atributo password.

A base de pesquisa LDAP (base DN) onde a busca de diferentes objetos deve começar. Este deve ser a 'raiz' do diretório LDAP que contem os usuários, grupos e contatos.

```
ldap_search_base = dc=example,dc=com
ldap_object_type_attribute = objectClass
ldap_user_type_attribute_value = posixAccount
ldap_group_type_attribute_value = posixGroup
ldap_contact_type_attribute_value = zarafa-contact
ldap_company_type_attribute_value = zarafa-company
ldap_addresslist_type_attribute_value = zarafa-addresslist
ldap_dynamicgroup_type_attribute_value = zarafa-dynamicgroup
```

Based on the ldap_object_type attribute, the Zarafa Server will create an object in the MySQL database, so it gets listed in the Global Address Book. Make sure that the values are always unique for one type of object, as Zarafa needs to be able to distinguish the different objects.

5.2.4. Configuração do usuário

Normally a user store is created for each object in the LDAP directory that has the user type attribute as mentioned in the previous section (posixAccount in the previous example). An additional search filter can be specified to limit store creation to a subset of the objects that have the user type attribute. For example:

```
ldap_user_search_filter = (zarafaAccount=1)
```

Todos os campos relacionados ao usuário podem ser mapeados pelas seguintes opções:

```
ldap_user_unique_attribute = uidNumber
```

```
ldap_user_unique_attribute_type = text
```

```
ldap_fullname_attribute = cn
ldap_loginname_attribute = uid
ldap_emailaddress_attribute = mail
ldap_emailaliases_attribute = zarafaAliases
ldap_password_attribute = userPassword
ldap_isadmin_attribute = zarafaAdmin
ldap_nonactive_attribute = zarafaSharedStoreOnly
```

O atributo exclusivo de usuário é o mapeamento entre uma caixa de correio no banco de dados e o usuário real no LDAP. Certifique-se que este campo nunca seja alterado por que o Servidor Zarafa perceberá isto como um usuário sendo excluído (e criado), e irá, assim, orfanar a armazenagem do usuário.

The email aliases are shown in the Global Address Book details and can be used for resolving email aliases in Postfix. However it is not possible to deliver email to email aliases with the dagent directly, this needs to be resolved by Postfix.

Informações adicionais do usuário, como endereços, números de telefone e informações sobre a empresa podem ser mapeados por um arquivo de configuração adicional:

```
!propmap /etc/zarafa/ldap.propmap.cfg
```

Os atributos especificados para os usuários também serão utilizados para contatos.

5.2.5. Configuração de grupos

Os grupos podem também ser filtrados por um filtro de pesquisa adicional.

```
ldap_group_search_filter = (objectClass=zarafa-group)
ldap_group_unique_attribute = gidNumber
ldap_group_unique_attribute_type = text
```

Para as relações de associação entre os grupos e usuários, cada objeto do grupo tem um atributo de membro do grupo. Isso pode ser configurado por:

```
ldap_groupmembers_attribute = memberUid
```

Por padrão o servidor Zarafa irá usar o atributo único de usuário como valor do atributo de membro do grupo. Isto pode ser alterado pelo atributo de relação de membro do grupo.

```
ldap_groupmembers_attribute_type = text
ldap_groupmembers_relation_attribute = uid
```

Os grupos podem ser sinalizados como grupos de segurança pelo atributo grupo de segurança. Grupos de segurança estão disponíveis no Catálogo Global de Endereços ao criar um novo e-mail e aplicando permissões. Para alcançar este atribute (aqui **`zarafaSecurityGroup`**) deve ser definido como **`1`**. Quando o atributo zarafaSecurityGroup é definido como 0, o grupo será um grupo de distribuição. Grupos de distribuição estão disponíveis apenas no Catálogo Global de Endereços ao criar um novo e-mail, mas não podem ser usados para configurar permissões de caixa de e-mail.

```
ldap_group_security_attribute = zarafaSecurityGroup
ldap_group_security_attribute_type = boolean
```

5.2.6. Configuração de lista de endereços

Listas de endereços são grupos de usuários que correspondem a uma condição personalizada. Estas listas de endereços são mostradas como subpastas no Catálogo Global de Endereços.

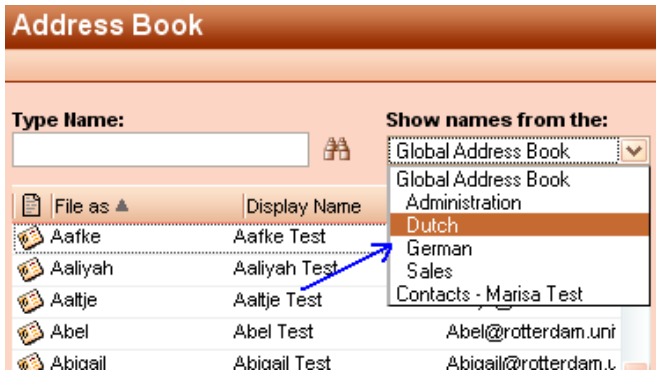


Figura 5.1. Lista de endereços no Catálogo Global de Endereços

Altere ou adicione no **`ldap.cfg`** as seguintes configurações para os objetos de lista de endereços:

```
ldap_addresslist_search_filter =
ldap_addresslist_unique_attribute = gidNumber
ldap_addresslist_unique_attribute_type = text
ldap_addresslist_filter_attribute = zarafaFilter
ldap_addresslist_name_attribute = cn
```

Veja *Seção 8.5, "Gestão do Usuário com o LDAP ou Active Directory"* para mais informações sobre como administrar listas de endereços.

5.2.7. Testando a configuração do LDAP

Após a configuração do LDAP ser feita, as mudanças podem ser ativadas recarregando o Servidor Zarafa.

```
/etc/init.d/zarafa-server reload
```

Para testar se os usuários e os grupos serão listados corretamente usando a configuração do LDAP, utilize:

```
zarafa-admin -l
```

para os usuários e grupos:

```
zarafa-admin -L
```

Se não houver usuários ou grupos mostrados, favor verificar os arguivos de registro do servidor Zarafa por erros. Definindo-se o **`log_level`** para **`6`** no **`/etc/zarafa/server.cfg`**, serão exibidas todas as consultas LDAP enviadas para o servidor e possíveis erros.

Nota

Na primeira vez que o **`zarafa-admin -l`** for executado, todas as caixas de correio serão criadas. Isso pode levar algum tempo, portanto seja paciente.

Mais informações sobre outros atributos do LDAP disponíveis podem ser encontradas na página do manual.

```
man zarafa-ldap.cfg
```

5.3. Configurar a integração do Active Directory do ZCP

5.3.1. Instalando o Plugin Zarafa ADS e arquivos de esquema

ZCP provides an installer for extending the Active Directory schema and installing an Active Directory snap-in for managing the Zarafa specific attributes (zarafaads.exe).

The Zarafa ADS plugin is only available in the commercial editions of ZCP and is part of the distribution packages which can be downloaded from *https://portal.zarafa.com*. The installer can be found inside of the **`windows`** subfolder.

The Zarafa ADS Plugin should be installed as a local administrator user on the Active Directory server which is the schema master.

Nota

Please restart the GUI after install of the Zarafa ADS plugin to show the Zarafa tab in the user details.

5.3.1.1. Servidor Windows 2000

Quando a instalação é executada em um Servidor Windows 2000, a instalação requer acesso de escrita para atualizar o esquema do Active Directory. Para obter o acesso de escrita, a chave de registro "Schema Update Allowed" deve estar ativada.

Para editar a chave de registro, execute os seguintes passos:

1. Clique em Iniciar, clique em Executar e, em seguida, na caixa Abrir, digite: **`regedit`**. Então pressione ENTER.
2. Localize e clique na seguinte chave do registro:

   ```
   HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\NTDS\Parameters
   ```

3. No menu Editar, clique em Novo e, em seguida clique em valor **`DWORD`**.
4. Digite os dados do valor quando o seguinte valor do registro for exibida:

   ```
   nome do Valor: Schema Update Allowed
   Tipo de Dados: REG_DWORD
   Base: Binary
   Dados do Valor: Digite 1 para habilitar esta característica, ou 0 (zero) para desabilitá-
   la.
   ```

5. Feche o editor do registro.

Agora o instalador do Zarafa Active Directory pode ser executado. Para mais informações, dê uma olhada em: *http://support.microsoft.com/kb/285172*

Nota

Não esqueça de mudar a chave de registro de volta ao valor original após a instalação.

5.3.1.2. Servidor Windows 2003/2008

For Windows 2003 and 2008 Server, it is possible to step through the setup by clicking the next button.

Se o Plugin Zarafa ADS estiver instalado, é possível editar os atributos específicos do Zarafa. Para editar um usuário, vá para **`users and computers`**, selecione um usuário e obtenha as propriedades. A aba Zarafa deve estar disponível se a instalação for concluída com sucesso.

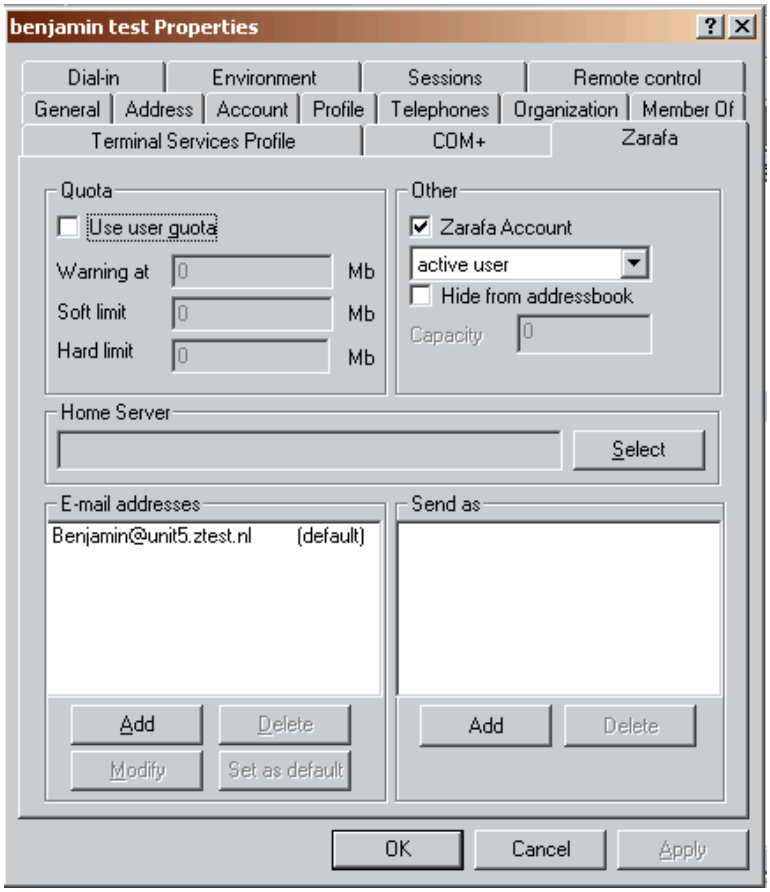


Figura 5.2. Aba de usuário do Zarafa

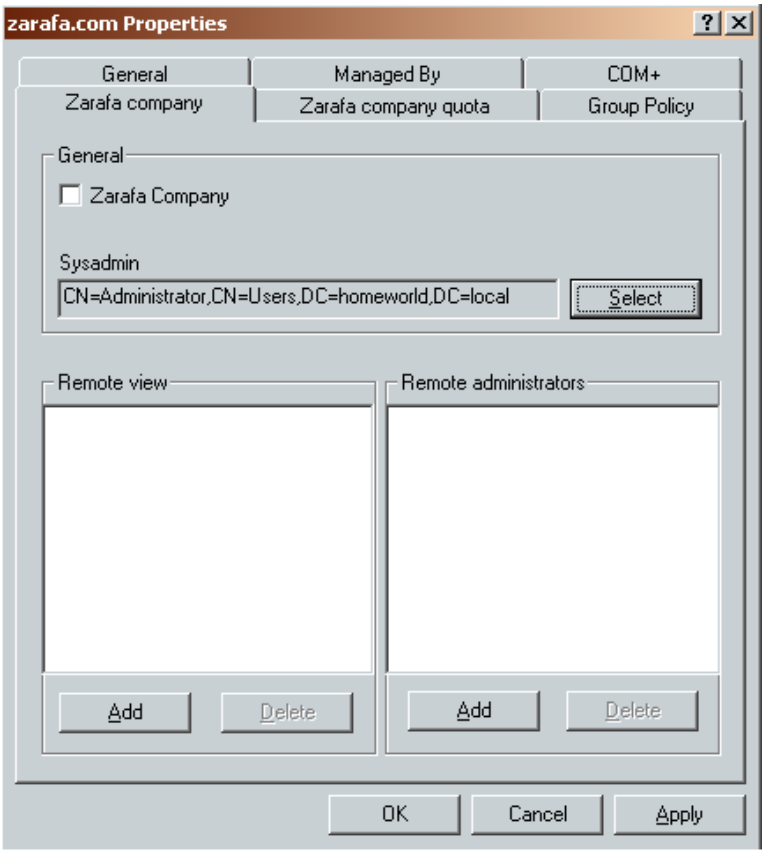


Figura 5.3. Aba de grupo do Zarafa

5.3.2. Configurando o ZCP para ADS

Para integrar o ZCP com um servidor do Active Directory, altere a seguinte opção no arquivo de configuração **`ldap.cfg`**:

Especifique na opção ldap_host o nome ou endereço IP ou nome do servidor do Active Directory.

```
ldap_host = 192.168.0.100
```

Por padrão, o protocolo LDAP simples será utilizado. Para a configuração de LDAP seguro, altere as seguintes definições:

```
ldap_port = 636
ldap_protocol = ldaps
```

Um guia para a configuração do Active Directory com certificados SSL pode ser encontrado em *um artigo em nosso wiki*[1].

To connect ZCP to multiple Active Directory servers, use the following setting:

```
ldap_uri = ldap://dc1:389 ldap://dc2:389
```

The different ldap uri's should be seperated by a whitespace. When using the **`ldap_uri`** option, the options **`ldap_host`**, **`ldap_port`** and **`ldap_protocol`** are ignored.

[1] http://www.zarafa.com/wiki/index.php/Configure_Active_Directory_with_SSL

O servidor Zarafa só lê do (e nunca escreve ao) servidor do LDAP ou Active Directory. Portanto, o usuário vinculado especificado deve ter pelo menos acesso de leitura no servidor LDAP.

```
ldap_bind_user = cn=administrador,cn=usuarios,dc=exemplo,dc=com,dc=br
ldap_bind_passwd = senha
ldap_authentication_method = bind
```

A base de pesquisa LDAP (base DN) especifica um ramo com qual o servidor Zarafa se limitará. Este deve ser a 'raiz' do diretório LDAP que contém os usuários, grupos e contatos.

```
ldap_search_base = dc=examplo,dc=com,dc=br
```

Pelo seguintes atributos de tipo o Servidor Zarafa sabe quais objetos criar no banco de dados e quais listar no Catálogo Global de Endereços. Certifique-se que estes valores são todos únicos.

```
ldap_object_type_attribute = objectClass
ldap_user_type_attribute_value = User
ldap_group_type_attribute_value = Group
ldap_contact_type_attribute_value = Contact
ldap_company_type_attribute_value = ou
ldap_addresslist_type_attribute_value = zarafa-addresslist
ldap_dynamicgroup_type_attribute_value = zarafa-dynamicgroup
```

Como característica de otimização de performance, a definição **ldap_page_size** foi implementada para limitar as séries de resultado nas páginas desse tamanho baixando menos resultados por vez do servidor do LDAP.

```
# Default ADS MaxPageSize is 1000.
ldap_page_size = 1000
```

5.3.3. Configuração do usuário

```
o qual tenha especificado o atributo de tipo do usuário, um filtro de pesquisa adicional pode
 ser especificado. Por exemplo:
```

```
ldap_user_search_filter = (zarafaAccount=1)
```

Todos os campos relacionados ao usuário podem ser mapeados pelas seguintes opções:

```
ldap_user_unique_attribute = objectGUID
ldap_user_unique_attribute_type = binary
```

```
ldap_fullname_attribute = cn
ldap_loginname_attribute = sAMAccountName
ldap_emailaddress_attribute = mail
ldap_emailaliases_attribute = otherMailbox
ldap_password_attribute =
ldap_isadmin_attribute = zarafaAdmin
ldap_nonactive_attribute = zarafaSharedStoreOnly
```

O atributo único do usuário é o mapeamento entre uma caixa de e-mail no banco de dados e o usuário real. Certifique-se que este campo nunca possa ser alterado, caso contrário, uma exclusão do usuário será acionada pelo Servidor Zarafa.

The email aliases are shown in the Global Address Book details and can be used for email aliases in Postfix. However, it is not possible to deliver email to email aliases.

Informações adicionais do usuário, como endereços, números de telefone e informações sobre a empresa podem ser mapeados por um arquivo de configuração adicional:

```
!include /etc/zarafa/ldap.propname.cfg
```

Os atributos especificados para os usuários também serão utilizados para os contatos.

Importante

O atributo **`otherMailbox`** é por padrão não indexado no Active Directory. É exigido indexar este atributo no Active Directory, caso contrário, o servidor do Active Directory terá uma carga alta da CPU durante consultas de pesquisa sobre esse atributo. Para mais informações sobre indexação de atributos no Active Directory, consulte *http://go.microsoft.com/fwlink/?LinkId=46790*.

5.3.4. Configuração de grupos

Os grupos podem também ser filtrados por um filtro de pesquisa adicional .

```
ldap_group_search_filter =
ldap_group_unique_attribute = objectSid
ldap_group_unique_attribute_type = binary
```

Para as relações de associação entre os grupos e usuários, cada objeto do grupo tem um atributo de membro do grupo. Isso pode ser configurado por:

```
ldap_groupmembers_attribute = member
ldap_groupmembers_attribute_type = dn
```

Usando atributo do grupo de segurança, o grupo pode ser especificado como grupos de segurança no Active Directory.

Grupos de segurança serão exibidos apenas quando se está configurando as permissões e por padrão não estão disponíveis no Catálogo Global de Endereços.

```
ldap_group_security_attribute = groupType
ldap_group_security_attribute_type = ads
```

5.3.5. Configuração de lista de endereços

As listas de endereços são grupos de usuários que correspondem a uma condição personalizada. Estas listas de endereços são mostradas como subpastas do Catálogo Global de Endereços.

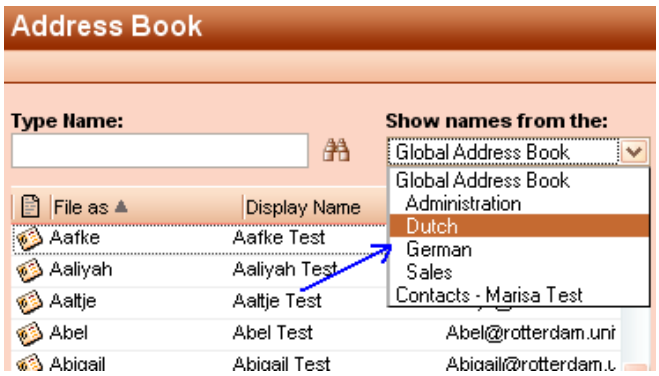


Figura 5.4. Lista de endereços no Catálogo Global de Endereços

Altere ou adicione no **ldap.cfg** as configurações seguintes para os objetos de lista de endereços.

```
ldap_addresslist_search_filter =
ldap_addresslist_unique_attribute = cn
ldap_addresslist_unique_attribute_type = text
ldap_addresslist_filter_attribute = zarafaFilter
ldap_addresslist_name_attribute = cn
```

Veja o *Seção 8.5, "Gestão do Usuário com o LDAP ou Active Directory"* para mais informações sobre como administrar listas de endereços.

5.3.6. Testando a configuração do Active Directory

Após a configuração do LDAP ser feita, as mudanças podem ser ativadas recarregando o Servidor Zarafa.

```
/etc/init.d/zarafa-server reload
```

Para testar os usuários e grupos que serão listados, use:

```
zarafa-admin -l
```

e

```
zarafa-admin -L
```

Se nenhum usuário ou grupo for mostrado, por favor verificar o arquivo de registro do servidor Zarafa a procura de erros. Definindo-se o loglevel para **6** no **/etc/Zarafa/server.cfg** irá exibir todas as consultas LDAP pelo servidor Zarafa e possíveis erros.

Na primeira vez que o **zarafa-admin -l** for executado, todas as caixas de correio serão criadas. Isso pode levar algum tempo, portanto seja paciente.

Mais informações sobre os outros atributos LDAP disponíveis podem ser encontradas na página no manual.

```
man zarafa-ldap.cfg
```

Veja *Capítulo 8, Gerenciamento dos Usuários* para gerenciamento de usuários do Zarafa com o Active Directory.

5.4. Integração do ZCP Postfix

ZCP does not include its own MTA, but can be integrated all established MTAs found in modern Linux distributions. Although ZCP support most Linux MTAs, we advise to use Postfix.

In order to deliver an email into a user's mailbox, the **zarafa-dagent** is executed. Messages are passed to the **zarafa-dagent** from the standard input or by the LMTP protocol. The usage of LMTP is the recommended delivery method as this enable the Single Instance Attachment Storage.

Alguns exemplos da integração do ZCP Postfix são descritos nas seções seguintes. Tenha em mente que o Postfix é muito flexível, por isso muitas configurações diferentes são possíveis, a maioria das quais estão além do escopo deste documento.

Nota

A configuração do antispam e antivírus está além do escopo deste manual. Na internet muitos exemplo de configurações estão disponíveis para os scanners e MTAs mais comuns.

5.4.1. Configurar a integração do ZCP Postfix com OpenLDAP

O MTA Postfix pode se conectar a um servidor OpenLDAP para determinar os endereços de e-mail principal e pseudônimo de usuários e grupos. O pacote Postfix tem suporte LDAP ativado por padrão na maioria das distribuições Linux. Para ler mais sobre suporte de LDAP do Postfix, veja *o LDAP README*[2] no site do Postfix.

Todos os arquivos de configuração do Postfix podem ser encontrados no diretório **`/etc/postfix`**. O arquivo de configuração principal é logicamente chamado **`main.cf`**

Por padrão, o Postfix só aceita e-mails recebidos do servidor local. Para aceitar e-mails da rede completa, configure a seguinte opção:

```
inet_interfaces = all
```

Para tornar Postfix ciente dos domínios locais de emails, adicione a seguinte linha ao **`main.cf`**.

```
virtual_mailbox_domains = exemplo.com.br, exemplo.com, exemplo.org, exemplo.net
```

Postfix will now see the configured domains as its local email domains, however, to accept incoming emails, Postfix will do a recipient check. Add the following lines to the **`main.cf`** to have Postfix use LDAP for looking up (valid) recipients:

```
virtual_mailbox_maps = ldap:/etc/postfix/ldap-users.cf
virtual_alias_maps = ldap:/etc/postfix/ldap-aliases.cf
virtual_transport = lmtp:127.0.0.1:2003
```

Todos os e-mails recebidos são entregues para o serviço LMTP do **`zarafa-dagent`**. A entrega precisa ser feito no endereço de e-mail principal de um usuário. Para resolver o endereço de e-mail primário do usuário, crie o arquivo **`/etc/postfix/ldap-users.cf`** e adicione as linhas seguintes:

```
server_host = localhost
search_base = ou=usuarios,dc=exemplo,dc=com,dc=br
version = 3
scope = sub
query_filter = (&(objectClass=posixAccount)(mail=%s))
result_attribute = mail
```

Para pesquisas de pseudônimos de e-mail, crie o arquivo **`/etc/postfix/ldap-aliases.cf`** e adicione as linhas seguintes:

```
server_host = localhost
search_base = ou=usuarios,dc=exemplo,dc=com,dc=br
version = 3
scope = sub
```

[2] http://www.postfix.org/LDAP_README.html

```
query_filter = (&(objectClass=posixAccount)(zarafaAliases=%s))
result_attribute = mail
```

A base de pesquisa de usuários e pseudônimos precisa corresponder à base de pesquisa do servidor LDAP. Depois que os arquivos de configuração forem alterados, o Postfix precisa ser reiniciado:

```
/etc/init.d/postfix restart
```

Certifique-se que o **zarafa-dagent** seja executado como um daemon e tenha sido iniciado no momento da reinicialização.

Para uso em distribuições baseadas em RPM:

```
chkconfig zarafa-dagent on
/etc/init.d/zarafa-dagent start
```

Para distribuições baseadas em Debian, habilite o zarafa-dagent definindo a opção DAGENT_ENABLED para **yes** no arquivo **/etc/default/zarafa-dagent**. Para habilitar o **zarafa-dagent** na hora da inicialização use:

```
update-rc.d zarafa-dagent defaults
```

Nota

É aconselhável habilitar o registro do **zarafa-dagent** quando estiver rodando no modo LMTP com o objetivo de monitoramento. Habilite as opções de registro no **zarafa-dagent** em **/etc/Zarafa/dagent.cfg**.

5.4.2. Configurar integração do Postfix do ZCP com o Active Directory

O Postfix pode analisar endereços de e-mail principal e pseudônimos de usuários e grupos do servidor do Active Directory. O pacote do Postfix tem suporte ao LDAP ativado por padrão na maioria das distribuições Linux. Para ler mais sobre o suporte do LDAP ao Postfix veja *o LDAP README*[3] no site do Postfix.

Todos os arquivos de configuração do Postfix podem ser encontrados no diretório **/etc/postfix**. O arquivo de configuração principal é logicamente chamado **main.cf**

Por padrão, o Postfix só aceita e-mails recebidos do servidor local. Para aceitar e-mails da rede completa, configure a seguinte opção:

```
inet_interfaces = all
```

Para tornar o Postfix ciente de domínios de emails locais, adicione a seguinte linha ao **main.cf**:

```
virtual_mailbox_domains = exemplo.com.br, exemplo.com, exemplo.org, exemplo.net
```

[3] http://www.postfix.org/LDAP_README.html

Postfix will now see the configured domains as its local email domains, however, to accept incoming emails Postfix will do a recipient check. This recipient check can be done on the Active Directory server. Add the following lines to the **`main.cf`**

```
virtual_mailbox_maps = ldap:/etc/postfix/ldap-users.cf
virtual_alias_maps = ldap:/etc/postfix/ldap-aliases.cf
virtual_transport = lmtp:127.0.0.1:2003
```

Todos os e-mails recebidos são entregues para o serviço LMTP do **`zarafa-dagent`**. A entrega precisa ser feito no endereço de e-mail principal de um usuário. Para resolver o endereço de e-mail primário do usuário, crie o arquivo **`/etc/postfix/ldap-users.cf`** e adicione as linhas seguintes:

```
server_host = 192.168.0.100
search_base = ou=usuarios,dc=exemplo,dc=com,dc=br
version = 3
bind = yes
bind_dn = cn=zarafa,ou=usuarios,dc=exemplo,dc=com,dc=br
bind_pw = senha
scope = sub
query_filter = (&(objectClass=user)(mail=%s))
result_attribute = mail
```

Para pesquisas de pseudônimos de e-mail, crie o arquivo **`/etc/postfix/ldap-aliases.cf`** e adicione as linhas seguintes:

```
server_host = 192.168.0.100
search_base = ou=usuarios,dc=exemplo,dc=com,dc=br
version = 3
bind = yes
bind_dn = cn=zarafa,ou=usuarios,dc=exemplo,dc=com,dc=br
bind_pw = senha
scope = sub
query_filter = (&(objectClass=user)(otherMailbox=%s))
result_attribute = mail
```

O Active Directory tem a possibilidade de criar grupos de distribuição que podem ser usados como lista de distribuição de e-mail no ZCP. Para integrar o Postfix com grupos de distribuição, o Postfix 2.4 ou superior é exigido.

Nota

Algumas distribuições de linux (como o RHEL 4 e 5) não incluem Postfix 2.4 ou superior. Pacotes de versões mais recentes do Postfix geralmente estão disponíveis nos pacotes contribuídos pela comunidade. No caso de RHEL 4 e 5, estes pacotes podem ser encontrados *aqui*[4].

Para suportar os grupos de distribuição, adicione a seguinte linha ao **`virtual_alias_maps`**:

```
virtual_alias_maps = ldap:/etc/postfix/ldap-aliases.cf, ldap:/etc/postfix/ldap-groups.cf
```

Crie um novo arquivo **`/etc/postfix/ldap-group.cf`** e insira nele a configuração do grupo LDAP:

```
server_host = 192.168.0.100
```

[4] http://www.linuxmail.info/postfix-rpm-packages

```
search_base = ou=grupos,dc=exemplo,dc=com,dc=br
version = 3
bind = yes
bind_dn = cn=zarafa,ou=usarios,dc=exemplo,dc=com,dc=br
bind_pw = senha
query_filter = (&(objectclass=group)(mail=%s))
leaf_result_attribute = mail
special_result_attribute = member
```

A base de pesquisa de usuários, pseudônimos e grupos necessita ser semelhante à base de pesquisa do servidor do Active Directory. Depois que os arquivos de configuração forem alterados, o Postfix precisa ser reiniciado:

```
/etc/init.d/postfix restart
```

Certifique-se que o **`zarafa-dagent`** seja executado como um daemon e tenha sido iniciado no momento da reinicialização.

Para uso em distribuições baseadas em RPM:

```
chkconfig zarafa-dagent on
/etc/init.d/zarafa-dagent start
```

Para distribuições baseadas em Debian, habilite o zarafa-dagent definindo a opção DAGENT_ENABLED para **`yes`** no arquivo **`/etc/default/zarafa-dagent`**. Para habilitar o **`zarafa-dagent`** na hora da inicialização use:

```
update-rc.d zarafa-dagent defaults
```

Nota

É aconselhável habilitar o registro do **`zarafa-dagent`** quando estiver rodando no modo LMTP com o objetivo de monitoramento. Habilite as opções de registro no **`zarafa-dagent`** em **`/etc/Zarafa/dagent.cfg`**.

5.4.3. Configurar a integração do Postfix do ZCP com usuários virtuais

Se nenhum OpenLDAP ou servidor do Active Directory estiver disponível, o Postfix pode ser configurado com usuários virtuais em um mapa esboçado. Nesta seção, vamos explicamos como.

Por padrão, o Postfix só aceita e-mails recebidos do servidor local. Para aceitar e-mails da rede completa, configure a seguinte opção:

```
inet_interfaces = all
```

Todos os arquivos de configuração do Postfix podem ser encontrados no diretório **`/etc/postfix`**. O arquivo de configuração principal é logicamente chamado **`main.cf`**

Para tornar o Postfix consciente dos domínios de e-mail locais, adicione a seguinte linha ao **`main.cf`**:

```
virtual_mailbox_domains = exemplo.com.br, exemplo.com, exemplo.org, exemplo.net
```

Postfix will now regard these domains as its local email domains. In order to accept incoming emails, Postfix will also need to validate the recipient. Add the following lines to the **`main.cf`** config file in order to have Postfix look up recipient from a hash map:

```
virtual_mailbox_maps = hash:/etc/postfix/virtual
virtual_alias_maps = hash:/etc/postfix/virtual
virtual_transport = lmtp:127.0.0.1:2003
```

O arquivo **`/etc/postfix/`** deve conter todos os endereços de e-mail e pseudônimos de um usuário, na seguinte estrutura :

```
#Endereço email ou alias       endereço email primário do usuário
joao@exemplo.com.br              joao@exemplo.com.br
usuario1@exemplo.com.br             usuario1@exemplo.com.br
usuario1@exemplo.br              usuario1@exemplo.com.br
alias_usuario1@exemplo.com.br       usuario1@exemplo.com.br
info@exemplo.com.br               usuario2@exemplo.com.br, usuario1@exemplo.com.br
```

A coluna da esquerda contém o endereço de email ou pseudônimo, a coluna da direita contém os endereços de e-mail principais para os quais a mensagem deve ser entregue.

Depois que todos os usuários e pseudônimos fores adicionadas a este arquivo, um mapa esboçado precisa ser criado. O comando seguinte irá criar tal mapa esboçado **`/etc/postfix/virtual.db`**.

```
postmap /etc/postfix/virtual
```

Todos os e-mails recebidos são entregues ao **`zarafa-dagent`** via LMTP usando o endereço de e-mail principal de usuário, conforme especificado no mapa esboçado.

Depois de mudar os arquivos de configuração, reinicie o Postfix através de seu script de inicialização:

```
/etc/init.d/postfix restart
```

Para uso em distribuições baseadas em RPM:

```
chkconfig zarafa-dagent on
/etc/init.d/zarafa-dagent start
```

Para distribuições baseadas em Debian, habilite o zarafa-dagent definindo a opção DAGENT_ENABLED para **`yes`** no arquivo **`/etc/default/zarafa-dagent`**. Para habilitar o **`zarafa-dagent`** na hora da inicialização use:

```
update-rc.d zarafa-dagent defaults
```

Nota

É aconselhável habilitar o registro do **`zarafa-dagent`** quando executado no modo LMTP com o propósito de monitoramento. Para alterar opções de registro para o **`zarafa-dagent`**, ajuste o arquivo de configuração: **`/etc/zarafa/dagent.cfg`**.

5.4.4. Configure ZCP Postfix integration with the DB plugin

Alternatively to managing virtual users in a file, the MySQL Database of Zarafa can be used to check if a message should be delivered. For this to work most of the configuration for *virtual users from a file* can be reused.

Nota

For this to work Postfix needs the ability to do lookups against a MySQL database. In Debian and Ubuntu this can be accomplished by installing the postfix-mysql package. When using Red Hat or Centos Postfix doesn't have the mysql module included. Alternatively the Postfix Package from the *Centos Plus repository*[5] can be used.

Instead of executing **`virtual_mailbox_maps`** and **`virtual_alias_maps`** against */etc/postfix/virtual*, a mysql lookup will be defined inside of **`main.cf`**.

```
alias_maps = hash:/etc/aliases
alias_database = hash:/etc/aliases
virtual_alias_maps = mysql:/etc/postfix/mysql-users.cf
```

This lookup is defined as pictured below:

```
# Replace with the user name and password to log into the MySQL server.
user = root
password = zarafa
hosts = 127.0.0.1
dbname = zarafa
query = select value from objectproperty where objectid=(select objectid from objectproperty
 where value='%s' limit 1) and propname='loginname';
```

This configuration only resolves the primary mail address of an user. Aliases should be kept in the /etc/aliases file or an extra aliases MySQL table.

Nota

Additionally MySQL could query alias definitions also from MySQL. As this would require additional MySQL knowledge from the administrator this has been left out in this manual. Further information on this can be found in the sub-chapter "virtual_alias_maps" of the chapter "Postfix/Database configuration" in the *ISPmail tutorial for Debian Squeeze*[6].

5.5. Configurar Z-Push (ActiveSync remoto para dispositivos móveis)

Este capítulo descreve como configurar o software Z-Push para conectar o ZCP com dispositivos compatíveis com ActiveSync.

Z-Push is an independent project available as an open source from *http://z-push.org/*

Neste manual, apenas a parte do servidor do Z-Push é discutida, por favor consulte o nosso Manual do Usuário para obter instruções sobre como configurar dispositivos móveis.

Telefones celulares, smartphones e PDAs podem ser sincronizados porque o Z-Push emula a funcionalidade do ActiveSync de um servidor MS Exchange no lado do servidor, permitindo aos

[5] http://mirror.centos.org/centos/5/centosplus
[6] https://workaround.org/article/postfixdatabase-configuration

celulares sincronizarem via ActiveSync *over-the-air* (AirSync). Usando o Z-Push, a maioria dos celulares pode sincronizar sem precisar instalar nenhum software adicional no aparelho.

Z-Push needs to be installed on a web server. It is highly recommended to use Apache. It is also highly recommended to use PHP as an Apache module.

Importante

Z-Push >=2.1 requires ZCP 7.0.6 or later.

5.5.1. Compatibilidade

Z-Push allows users with PDAs and smartphones to synchronise their email, contacts, calendar items and tasks directly from a compatible server over UMTS, GPRS, WiFi or other GSM data connections. Among others the following devices are known to by working with Z-Push:

- Apple iPhone and iPad
- Windows Phone 7, 7.5 and 8
- Android phones with Android 4.x and newer
- Blackberry PlayBook and 10 (with ActiveSync)
- outros dispositivos compatíveis com o ActiveSync

For detailed information about the devices and their compatibility status, please consult the Mobile Compatibility List at *http://www.zarafa.com/wiki/index.php/Mobile_Compatibility_List*

5.5.2. Segurança

Para criptografar dados entre os dispositivos móveis e o servidor, é necessário ativar o suporte SSL no servidor da web. Configurar o Apache com certificados SSL está além do escopo deste documento, embora muitos tutoriais possam ser encontrados online.

Keep in mind that some mobile devices require an official SSL certificate and don't work with self signed certificates. For Windows Phone and Windows Mobile you might need to install the certificates on the device (See *Seção 5.6, "Configuring SSL for Windows Mobile and Windows Phone"* for details).

5.5.3. Instalação

Download the latest Z-Push software from *http://z-push.org/download/*

To install Z-Push, simply extract the Z-Push archive to the /usr/share/z-push directory:

```
mkdir -p /usr/share/z-push
tar zxvf z-push-*.tar.gz -C /usr/share/z-push/ --strip-components=1
```

The **`-C`** option is the destination where the files need to be installed.

Z-Push is using a state directory to store a per-user synchronisation status and a log directory for its default logging. Make sure that the 'state' and 'log' directories exists and are writeable for the webserver process, so either change the owner of the 'state' directory to the UID of the apache process or make it world writeable:

```
mkdir /var/lib/z-push /var/log/z-push
```

```
chown www-data:www-data /var/lib/z-push /var/log/z-push
```

O nome do usuário e grupo do Apache variará em cada distribuição do Linux. A tabela abaixo mostra os nomes gerais do usuário e grupo do processo Apache.

Tabela 5.2. Nome do usuário e grupo por distribuição

Distribuição	Nome do usuário Apache	Nome do grupo
Red Hat Enterprise Linux	apache	apache
SLES	wwwrun	www
Debian e Ubuntu	www-data	www-data

On systems with SELinux enabled the security context of these folders might need to be changed, e.g.

```
chcon -R -t httpd_sys_rw_content_t /var/lib/z-push
chcon -R -t httpd_sys_rw_content_t /var/log/z-push
```

Now, Apache must be configured to redirect the URL **`Microsoft-Server-ActiveSync`** to the **`index.php`** file in the z-push directory. This can be done by adding the following line to the **`httpd.conf`** file:

```
Alias /Microsoft-Server-ActiveSync /usr/share/z-push/index.php
```

Certifique-se que a linha seja adicionada à parte correta da configuração do Apache, cuidando de hospedagens virtuais e outras configurações do Apache.

Additional PHP Packages

To use the full featureset of Z-Push 2 and the z-push-top command line utility, additional php packages are required. These provide SOAP support, access to process control and shared memory.

Tabela 5.3. Additional packages per distribution

Distribuição	Package name
Red Hat Enterprise Linux*	php-cli php-soap php-process
SLES**	php53 php53-soap php53-pcntl php53-sysvshm php53-sysvsem php53-posix
Debian e Ubuntu	php5-cli php-soap

- To install the php-process package you need to add an extra channel subscription from the RHEL Server Optional channel.
  - The PHP Posix package is included in the SLES SDK Repository.

Importante

It is not possible to simply rename the **`Z-Push`** directory to **`Microsoft-Server-ActiveSync`**. This will cause Apache to send redirects to the smartphone, which will prevent proper synchronization.

Por fim, certifique-se que o PHP tenha as seguintes configurações:

```
php_flag magic_quotes_gpc = off
php_flag register_globals = off
php_flag magic_quotes_runtime = off
php_flag short_open_tag = on
```

Set this in the **`php.ini`** or in a **`.htaccess`** file in the root directory of Z-Push.

If you have several php applications on the same system, you could specify the z-push directory so these settings are considered only there.

```
<Directory /usr/share/z-push>
    php_flag magic_quotes_gpc off
    php_flag register_globals off
    php_flag magic_quotes_runtime off
    php_flag short_open_tag on
</Directory>
```

If not setup correctly, the smartphone will not be able to login correctly via Z-Push.

Recarregue o Apache para ativar essas mudanças.

To use the Z-Push 2.X command line tools, access the installation directory **`/usr/share/z-push`** and execute:

```
./z-push-top.php
```

and/or

```
./z-push-admin.php
```

To facilitate the access symbolic links can be created, by executing:

```
ln -s /usr/share/z-push/z-push-admin.php /usr/local/sbin/z-push-admin
```

```
ln -s /usr/share/z-push/z-push-top.php /usr/local/sbin/z-push-top
```

With these symlinks in place the cli tools can be accessed from any directory and without the .php file extension.

5.5.4. Gerenciamento de dispositivo móvel

Users can remote wipe own mobile devices from the ZCP Webaccess without interaction of the system administrator. The Mobile Device Management (MDM) plugin can be downloaded at: *https://community.zarafa.com/pg/plugins/project/151/developer/sebastian/mobile-device-management-plugin*

O administrador do sistema pode executar limpeza remota de dispositivos da linha de comando usando a ferramenta **`z-push-admin`**.

5.5.5. Atualização

A atualização para uma versão mais recente do Z-Push segue o mesmo caminho que a instalação inicial.

Ao atualizar para uma nova versão menor, por exemplo, de Z-Push 1.4 a Z-Push 1.4.1, o diretório Z-Push existente pode ser substituído quando da extração do arquivo. Ao instalar uma nova versão

maior é recomendado extrair o arquivo tar para outro diretório e copiar o estado da instalação existente.

Importante

É fundamental sempre manter os dados do diretório de estado para garantir a consistência dos dados nos dispositivos móveis já sincronizados.

Sem a informação de estado, os dispositivos móveis, que já têm um perfil do ActiveSync, irão receber itens duplicados ou a sincronização será interrompida completamente.

Importante

Upgrading to Z-Push 2.X from 1.X it is not necessary to copy the state directory because states are not compatible. However Z-Push 2 implements a fully automatic resynchronizing of devices in the case states are missing or faulty.

Importante

Downgrading from Z-Push 2.X to 1.X is not simple. As the states are not compatible you would have to follow the procedure for a new installation and re-create profiles on every device.

Importante

States of Z-Push 2.0 and Z-Push 2.1 are not compatible. A state migration script is available in the tools folder.

Por favor observe também as notas de lançamento publicadas da nova versão Z-Push. Em algumas versões é necessário, por exemplo, sincronizar o celular novamente.

5.5.6. S/MIME

Z-Push supports signing and en-/decrypting of emails on mobile devices since the version 2.0.7.

Importante

Currently only Android 4.X and higher and iOS 5 and higher devices are known to support encryption/signing of emails.

It might be possible that PHP functions require CA information in order to validate certs. Therefore the CAINFO parameter in the config.php must be configured properly.

The major part of S/MIME deployment is the PKI setup. It includes the public-private key/certificate obtaining, their management in directory service and roll-out to the mobile devices. Individual certificates can either be obtained from a local (company intern) or a public CA. There are various

public CAs offering certificates: commercial ones e.g. Symantec or Comodo or community-driven e.g. CAcert.org.

Both most popular directory services Microsoft Active Directory (MS AD) and free open source solution OpenLDAP allow to save certificates. Private keys/certificates reside in user's directory or on a smartcard. Public certificates are saved in directory. MS AD and OpenLDAP both use userCertificate attribute to save it.

In Active Directory, the public key for contacts from GAB is saved in PR_EMS_AB_TAGGED_X509_CERT (0x8C6A1102) property, and if you save a key in a contact, it is PR_USER_X509_CERTIFICATE (0x3A701102).

In LDAP public key for contacts from GAB is saved in userCertificate property. It should be mapped to 0x3A220102 in ldap.propmap.cfg (0x3A220102 = userCertificate). Make sure it looks like this in LDAP:

```
userCertificate;binary
    MIIFGjCCBAKgAwIBAgIQbRnqpxlPa…
```

Importante

It is strongly recommended to use MS AD or LDAP to manage certificates. Other user plugin options like db or unix might not work correctly and are not supported.

For in-depth information please refer to: *http://www.zarafa.com/blog/post/2013/05/smime-z-push-signing-and-en-decrypting-emails-mobile-devices*

5.6. Configuring SSL for Windows Mobile and Windows Phone

If you don't have a certificate of one of the Certified Authorities, you also need to add the CA Certificate to the Trusted Root Certificates store of the device.

The certificates should be in DER format to install it on a windows device. By default the generated SSL certificates on Linux are in PEM format. The DER certificate is a base64 encoded PEM certificate. You can convert the certificate type by the following commands:

```
openssl x509 -in ca.crt -inform PEM -out ca.cer -outform DER
```

```
openssl x509 -in host.crt -inform PEM -out host.cer -outform DER
```

where **`ca.crt`** is your CA certificate file and **`host.crt`** is your certified file.

After converting both certificates you need to copy them to the PDA. It can be e.g. done by putting the files on a local intranet server and accessing them with the device's browser:

```
http://intranet/certs/ca.cer
```

```
http://intranet/certs/host.cer
```

By selecting the certificates on your PDA they will be stored in the Trusted Root Certificates store of your device.

5.7. Solução de problemas

Configuração geral

A maioria das dificuldades são causadas por configurações incorretas do Apache. A configuração do Apache pode ser testado utilizando-se um navegador como o Firefox, direcionando-o para:

```
http://<servidor>/Microsoft-Server-ActiveSync
```

Se for corretamente configurado, uma janela solicitando nome e senha do usuário deverá ser exibida. Ao autenticar utilizando-se credenciais válidas, será exibida a página de informações do Z-Push, contendo a seguinte mensagem:

Deve ser exibida uma página de informações do Z-Push, contendo a seguinte mensagem:

```
*GET not supported*
This is the z-push location and can only be accessed by Microsoft ActiveSync-capable devices.
```

Verifique o PHP e/ou a configuração do Apache se um erro for exibido.

Problemas de sincronização

Importante

The following text regarding debug.txt and WBXML debug applies to Z-Push 1.X versions only. In Z-Push 2 there is a separate log directory and the loglevel is configured in **`config.php`**.

Se os problemas de sincronização são encontrados, um arquivo **`debug.txt`** tem que ser criado no diretório raiz do Z-Push. Este arquivo deve ser escrevível pelo processo do servidor Apache.

```
touch /var/www/z-push/debug.txt
chmod 777 /var/www/z-push/debug.txt
```

O arquivo **`debug.txt`** irá coletar informações de erros sobre a sincronização.

Para obter uma lista completa da sincronização, o arquivo **`wbxml.php`** deve ser editado e o parâmetro **`WBXML_DEBUG`** definido como verdadeiro:

```
define('WBXML_DEBUG', true);
```

Importante

O arquivo de listagem **`debug.txt` contem dados confidenciais e deve ser protegido** para que não seja possível baixá-lo da internet.

Para proteger o arquivo de listagem **`debug.txt`**, um **`.htaccess`** deve ser criado no diretório raiz z-push, contendo:

```
<Files debug.txt>
 Deny from All
</Files>
```

Log messages

- **Repetidamente "Command denied: Retry after sending a PROVISIONING command":**

Muito provavelmente o dispositivo móvel não suporta provisionamento. O parâmetro LOOSE_PROVISIONING deve ser ativado na configuração. Se as mensagens continuarem, o perfil do ActiveSync deve ser reconfigurado no dispositivo. Se isso não ajudar, o provisionamento pode ser totalmente desabilitado no arquivo de configuração (isso se aplica para todos os dispositivos!). Mais informações podem ser encontradas em: *http://www.zarafa.com/wiki/index.php/Z-Push_Provisioning*

- **Exceções para pedidos de reunião causam itens duplicados se aceitos no celular:**

Por favor, atualize para Z-Push 1.4 ou posterior. Para corrigir as duplicatas existentes, o perfil do ActiveSync no celular deve ser recriado ou, pelo menos, o calendário precisa ser completamente ressincronizado (desativando a sincronização do calendário e ativando-o posteriormente).

Repeated incorrect password messages

If a password contains characters which are encoded differently in ISO-8859-1 and Windows-1252 encodings (e.g. "§") the login might fail with Z-Push but it works fine with the WebApp/Webaccess. The solution is to add setlocale(LC_CTYPE, "en_US.UTF-8"); to the **`config.php`** file.

Importante

The solution above is for ZCP 7 and later versions only. ZCP 6 and earlier versions might not work properly because they lack unicode support.

Configurações avançadas

Este capítulo descreve como configurar itens especiais que vão além das instalações mais comuns do ZCP.

6.1. Rodando componentes do ZCP além da hospedagem local

Ao usar a conexão SSL com certificados não só será possível encriptar a conexão, mas os serviços Linux também serão capazes de logar usando um certificado SSL de um cliente.

Repita a criação do certificado para criar certificados para programas de cliente como o **`zarafa-spooler`**, **`zarafa-monitor`**, **`zarafa-gateway`**, **`zarafa-dagent`** e **`zarafa-admin`**. É possível criar um certificado para todos esses programas, ou um certificado pode ser criado para cada programa separadamente. Esses clientes podem então logar nas conexões SSL com seu certificado como autenticação.

```
Cliente sh /usr/share/doc/zarafa/ssl-certificates.sh
```

Novamente, ao inserir os detalhes do certificado, faça ao menos um nome diferente do Nome da Unidade Organizacional dos outros certificados. Além disso, não esqueça de completar o campo Nome Comum.

Quando necessário criar a chave pública, digite y e pressione enter. Agora um novo certificado chamado **`client.pem`** e uma chave pública chamada **`client-public.pem`** estarão presentes. Como um exemplo, opções de configuração necessárias para editar no arquivo **`dagent.cfg`** estarão como mostradas a seguir:

```
server_socket = https://name-or-ip-address:237/zarafa
sslkey_file = /etc/zarafa/ssl/client.pem
sslkey_pass = ssl-client-password
```

Importante

Para a função **`zarafa-admin`** funcionar corretamento em uma configuração multi-server, uma arquivo **`admin.cfg`** é necessário no diretório de configuração do ZCP, normalmente o **`/etc/zarafa/`**. Ele também deve conter as opções mencionadas acima.

Entre com o nome ou endereço IP correto na opção server_socket. Se outro número de porta para as coneções SSL no servidor for utilizado, entre com o núnero de porta correto da mesma forma. Substitua a senha pela senha utilizada quando da criação do certificado.

Copie o arquivo **`client-public.pem`** para o local do servidor:

```
mkdir /etc/zarafa/sslkeys
mv client-public.pem /etc/zarafa/sslkeys
```

Agora que o cliente e o servidor conhecem a chave privada, o cliente pode entrar com esta chave no servidor de qualquer local na rede ou internet.

Nota

Atenção ao arquivo **`client.pem`**. Qualquer pessoa que tenha a chave privada pode entrar no servidor Zarafa e será um usuário interno do SISTEMA, que pode fazer qualquer coisa sem restrições.

6.2. Configurações de Multi-locação

Esta seção explica sobre a funcionalidade de multi-locação, a qual foi introduzida no Zarafa 6.10. Esta característica está disponível em todas as edições, mas somente suportada oficialmente nas edições Enterprise e Hosted.

O modo multi-locação permite que organizações rodem múltiplas organizações em um único servidor ZCP onde membros das diferentes organizações não verão um ao outro.

6.2.1. Plugins de usuário suportados

O suporte ao modo multi-locação somente pode ser disponibilizado quando se utiliza o plugins DB ou LDAP. Atualmente não é possível utilizar o plugin Unix. Ao utilizar o plugin DB, a ferramenta **`zarafa-admin`** pode ser usada para administrar os locatários (companhias), enquanto ao se utilizar o plugin LDAP toda informação virá diretamente do LDAP ou do Diretório Ativo.

Importante

O plugin de usuário recomendado para configurações de multi-locação é o plugin LDAP.

6.2.2. Configurando o servidor

A seguintes opções de configuração no **`server.cfg`** serão utilizadas ao se habilitar o suporte à multi-locação.

```
enable_hosted_zarafa
```

When set to **`true`**, it is possible to create tenants within the Zarafa instance and assign all users and groups to particular tenants. When set to **`false`**, the normal single-tenancy environment is created.

```
createcompany_script
```

Local do script **`createcompany`** que será executado quando um novo locatário tiver sido criado.

```
deletecompany_script
```

Local do script **`deletecompany`** que será executado quando um locatário tiver sido deletado.

```
loginname_format
```

Ver *Seção 6.2.2.2, “Configurando o nome de acesso”* para mais detalhes sobre esta opção de configuração.

```
storename_format
```

Ver *Seção 6.2.2.3, "Configurando o nome de armazenagem"* para mais detalhes sobre esta opção de configuração.

6.2.2.1. Habilitando a Multi-locação

Para habilitar o suporte à multi-locação, mude a seguinte opção de configuração em **`server.cfg`**:

```
enable_hosted_zarafa = true
```

6.2.2.2. Configurando o nome de acesso

O nome de acesso de um usuário deve ser único para permitir corretamente a tentativa de acesso. Ao se habilitar o suporte à multi-locação no Zarafa, ter um nome de acesso exclusivo pode se tornar difícil ao passo que o número de companhias (locatários) aumenta. É mais fácil quando o *nome de acesso* contém também o *nome da companhia*, para assegurar que todos dos *nomes de acesso* sejam exclusivos.

A maneira com que o *companyname* é 'vinculado' ao nome do usuário para criar o nome de acesso pode ser configurada com a opção de configuração **`loginname_format`** em **`server.cfg`**. Esta configuração pode conter as seguintes variáveis:

- **%u** - O*nome do usuário*
- **%c** - O*nome da companhia* à qual o usuário pertence

Como caracter de separação entre o *nome do usuário* e o *nome da companhia*, um caracter deve ser escolhido que não faça parte do *nome do usuário* ou do *nome da companhia*. Caracteres válidos são **@** and ****, por exemplo.

Alguns exemplos de **`loginname_format`** para um usuário chamado "João da Silva" que é membro da "Exemploorg":

- **%u >** joao
- **%u@%c >** *joao@exemploorg*[1]
- **\\%c\%u >** \\exemploorg\joao

Embora ter um *nome de acesso* que contenha um **%c** seja obrigatório para o plugin DB, ele é opcional para o plugin LDAP. Coordenar *nomes de usuários exclusivos é mais fácil no LDAP porque é possível utilizar o endereço de email como o nome de acesso*.

Nota

Ao passar um nome de usuário para a ferramenta **`zarafa-admin`** ele deve ser formatadao como foi configurado. Por exempro, se o valor de configuração **`loginname_format`** incluir a variável de nome de companhia (**%c**), o nome da companhia deve ser ser passado para a ferramenta **`zarafa-admin`** toda vez que um nome de usuário for necessário.

6.2.2.3. Configurando o nome de armazenagem

Quando as relações entre múltiplos locatários (companhias) são permitidas, é possível que usuários compartilhem seus arquivos armazenados com usuários de outro locatário. Para diferenciar facilmente os arquivos armazenados de diferentes locatários, o nome da armazenagem pode ser

formatado para conter o nome do locatário (*nome da companhia*) à qual o usuário/armazenagem pertence.

Em **`server.cfg`**, a opção de configuração **`storename_format`** foi feita exatamente para este propósito. No formato diferentes variáveis estão disponíveis, as quais podem ser utilizadas em diferentes tipos de informação.

- **`%u`** — The *username*
- **`%f`** — O*nome completo* do usuário
- **`%c`** — O *nome da companhia*, nome do locatário ao qual o usuário pertence

Alguns exemplos para um usuário chamado 'João da Silva', o qual é membro do locatário 'Exemploorg':

- **`%u` >** joao
- **`%f` >** Joao da Silva
- **`%f (%c)` >** Joao da Silva (Exemploorg)

6.2.2.4. Configurando o plugin LDAP

Ao utilizar o plugin DB, nenhuma configuração adicional é necessária. Para o plugin LDAP há várias opções de configuração que podem requerer mudanças.

For a multi-tenancy LDAP setup, it is necessary to have the different company in the LDAP tree and below every company container the users, groups and contacts within that specific company. It's not possible to assign a user to a specific company by an LDAP attribute.

Veja a imagem abaixo de um exemplo de estrutura LDAP.

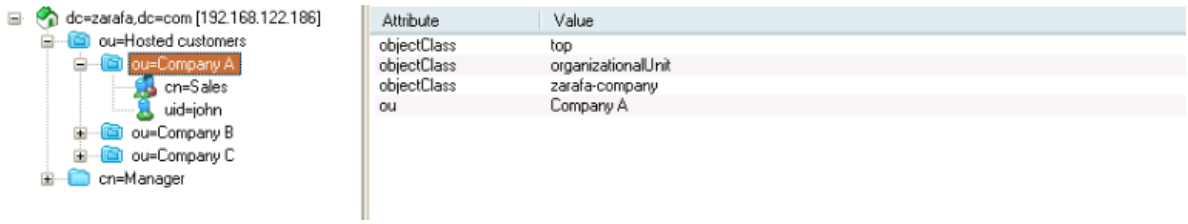


Figura 6.1. Ambiente multi-locatário da raiz do LDAP

Mude as seguintes linhas no arquivo de configuração do LDAP para configurar o supote à multi-locação.

```
ldap_company_unique_attribute = ou
ldap_companyname_attribute = ou
ldap_company_scope = sub
```

Teste as configurações usando **`zarafa-admin --list-companies`** e **`zarafa-admin -l`**.

Caso não for mostrada nenhuma companhia ou usuário, verifique possíveis erros no arquivo de acesso ao servidor Zarafa. Definino do nível de acesso em **`6`** no **`/etc/zarafa/server.cfg`** mostrará todas as dúvidas sobre o LDAP pelo servidor Zarafa e possíveis erros.

Com o suporte à multi-locação habilitado não só é possível ter diferentes organizações em um único servidor, mas também outras definições avançadas podem ser configuradas, como delegação de caixa de mensagens de organizações cruzadas, diferentes níveis de administrador e níveis de quotas de organizações.

Veja a página principal **zarafa-ldap.cfg** para informações mais detalhadas sobre essas caracteríscitas do LDAP de multi-locação.

```
man zarafa-ldap.cfg
```

6.2.2.5. Pastas de armazenagem públicas

Uma vez que o servidor tenha inicializado corretamente, pastas de armazenagem podem ser criadas. Há dois tipos pastas de armazenagem: particulares e públicas. Pode haver somente uma pasta de armazenagem pública por companhia. Ao criar uma companhia, a pasta de armazenagem pública será criada simultaneamente. Se por alguma razão a pasta de armazenagem pública não for criada, ela pode ser criada manualmente executando-se o seguinte comando:

```
/usr/bin/zarafa-admin -s -I <tenant>
```

Substitua **<tenant>** pelo nome do locatário (companhia) para a qual a pasta pública deve ser criada. Quando a opção **-I** não é utilizada, a pasta pública será criada para um ambiente de locatário único (e não será acessível quando a multi-locação do Zarafa estiver habilitada). A pasta pública é, por padrão, disponível para todos os usuários que fazem parte do locatário (companhia).

6.2.3. Admininstrando espaços do locatário (companhia)

Nota

A administração de espaços de locatário (companhia) através do **zarafa-admin** só é possível ao se usar o plugin DB. Quando o plugin LDAP é utilizado, toda a administração precisa ser feita através do LDAP ou do servidor do Diretório Ativo.

Para criar um espaço da companhia use o seguinte comando:

```
/usr/bin/zarafa-admin --create-company <companyname>
```

Para deletar um espaço da companhia use o seguinte comando:

```
/usr/bin/zarafa-admin --delete-company <companyname>
```

Para alterar um espaço da companhia use o seguinte comando:

```
/usr/bin/zarafa-admin --update-company <companyname>
```

Este comando pode ser combinado com a opção **--qw** para definir o nível de alerta de quota para o espaço da companhia especificado.

Para controlar os privilégios de visualização para espaços de companhia os seguintes comandos podem ser utilizados:

```
/usr/bin/zarafa-admin --add-to-viewlist <viewer> -I <companyname>
/usr/bin/zarafa-admin --add-to-viewlist <viewer> -I <companyname>
/usr/bin/zarafa-admin --list-view -I <companyname>
```

O **<viewer>** é o nome da companhia que recebe ou perde permissão para visualizar **<companyname>**. Com os privilégios e visualização o Catálogo Global de Endereços pode ser

compartilhado entre múltiplas organizações ou usar a delegação de caixa de mensagens de organizações cruzadas.

```
/usr/bin/zarafa-admin --add-to-adminlist <admin> -I <companyname>
/usr/bin/zarafa-admin --del-from-adminlist <admin> -I <companyname>
/usr/bin/zarafa-admin --list-view -I <companyname>
```

The **<admin>** is the loginname of the user who receives or looses admin privileges over the company **<companyname>**. Please note that a user that is administrator over a tenant still needs to be given view privileges to this tenant to see its stores.

6.2.4. Administrando usuários e grupos

Ao utilizar o plugin DB, usuários e grupos devem ser criados utilizando a ferramenta **zarafa-admin**. Para detalhes sobre como utilizar a ferramenta **zarafa-admin** veja **man zarafa-admin**. O nome do usuário ou grupo que deve ser informado para a ferramenta **man zarafa-admin**. depende da opção de configuração **loginname_format**.

Por exemplo, quando **loginname_format** for definido para **%u@%c**, a criação de um usuário para locatário **exemploorg** deverá ser:

```
/usr/bin/zarafa-admin -c john@exampleorg ...other options...
```

E a criação de um novo grupo para locatário **exampleorg** deverá ser:

```
/usr/bin/zarafa-admin -g grupo@exemploorg ...outras opções...
```

6.2.5. Níveis de quota

Ao utilizar uma instalação de multi-locação há 2 tipos de quota, quais sejam a quota para o locatário (companhia) e a quota para o usuário individual. A quota para o locatário é verificada sobre o tamanho total da pasta de armazenagem de todos os usuários dentro do locatário mais a armazenagem pública.

Neste ponto somente a quota de alerta pode ser configurada para um locatário, o que significa que não é possível definir a quota maleável ou rígida para limitar as capacidades de email do locatário.

Assim como a quota do usuário, há vários níveis de quota de locatário e há até mesmo um novo nível para a quota de usuário. Um sumário com os possíveis níveis de quota que podem ser definidos em um ambiente de multi-locação:

1. Quota do locatário (companhia):
    a. **Global company quota**: Configurado em **/etc/zarafa/server.cfg** e afeta todos os locatários dentro do sistema.
    b. **Specific company quota**: O nível de quota para um locatário configurado através de plugin(LDAP ou ferramenta **zarafa-admin**).
2. Quota do usuário:
    a. **Global user quota**: esta quota é configurada em **/etc/zarafa/server.cfg** e afeta todos us usuários de um locatário.
    b. **Company user quota**: Este é o nível da quota padrão para todos os usuários dentro de um locatário e é configurado através do plugin no nível de locatário.

c. **`Specific user quota`**: Este é o nível da quota para um usuário específico e é configurado através do plugin de usuário.

Como mencionado acima, o **`Global company quota`** e **`Global user quota`** podem ser configurados no arquivo **`/etc/zarafa/server.cfg`**, onde há as opções **`quota_warn`**, **`quota_soft`** e **`quota_hard`** para a quota de usuário e as opções **`companyquota_warn`** para a quota de locatário.

Para configurar a **`Specific company quota`** a ferramenta **`zarafa-admin`** pode ser utilizada ao se usar o plugin DB. O seguinte comando definirá os vários níveis de quota referentes ao locatário:

```
zarafa-admin --update-company <tenant> --qo y --qw <warningquota>
```

Para configurar o **`Specific user quota`**, a ferramenta **`zarafa-admin`** pode ser utilizada ao se usar o plugin DB. O seguinte comando definirá os vários níveis de quota referentes ao usuário:

```
zarafa-admin -u <user> --qo y --qh <hardquota> --qs <softquota> --qw <warningquota>
```

Para configurar o **`Company user quota`**, a ferramenta **`zarafa-admin`** pode ser utilizada com o plugin DB usando o argumento **`--update-company`**. O seguinte comando definirá os vários níveis de quota de usuário sobre o locatário:

```
zarafa-admin --update-company <tenant> --udqo y --udqh <hardquota> --udqs <softquota> --udqw
 <warningquota>
```

Ao se utilizar o plugin LDAP, os atributos que controlam os níveis de quota podem ser configurados em **`/etc/zarafa/ldap.cfg`**.

6.2.6. Usuários Administradores

Em uma instalação de multi-licação, há dois tipos de usuários administradores:

- Administrador Amplo do Sistema
- Aministrador da empresa

O administrador do sistema pode acessar todas as caixas de correio dentro do ambiente hospedado. Um administrador da empresa somente pode acessar as caixas de correio dentro da organização local.

Um administrador do sistema pode ser configurado definindo o atributo **`zarafaAdmin`** para 2 ao se utilizar o LDAP ou utilizar -a 2 ao utilizar o plugin DB. Um administrador da empresa pode ser configurado definindo o atributo do **`zarafaAdmin`** para 1.

O tipo de usuário administrador pode ser requisitado utilizando-se as ferramentos do zarafa-admin:

```
zarafa-admin --details <admin username>
Username: admin@example.com
Fullname: Administrator
Emailaddress: admin@example.com
Active: yes
Administrator: yes (system)
```

6.3. Configuração de Multi-servidor

Este capítulo dá informações sobre a funcionalidade multi-servidor, a qual foi introduzida no Zarafa 6.30.

Nota

Para que se possa utilizar esta funcionalidade, uma chave de licensa válida da Zarafa Enterprise é necessária e uma licença-zarafa em processamento é requerida.

6.3.1. Introdução

A funcionalidade multi-servidor do ZCP possibita a distribuição do ZCP para múltiplos servidores. Nesta situação, as pastas de usuário do Zarafa são divididas para vários servidores, mas ainda agem como um sistema central. Os usuários, grupos e locatários (companhias)precisam ser gerenciados em um servidor de Diretório Ativo ou em um LDAP.

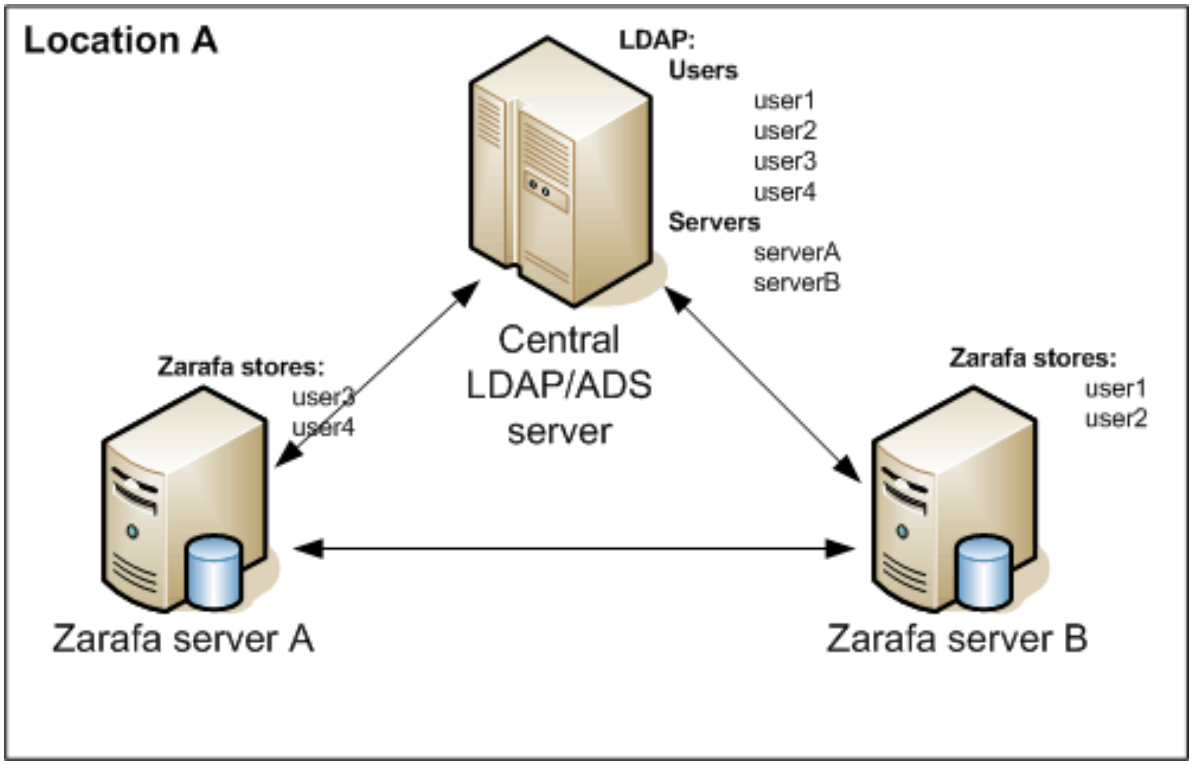


Figura 6.2. Multiserver environment in one location

O suporte multi-servidor pode também ser usado como apoio para um grande número de usuários ou para espalhar caixas de mensagens em locais geográficos diferentes, como visto em *Figura 6.3, "Ambiente multi-servidor em dois locais"*.

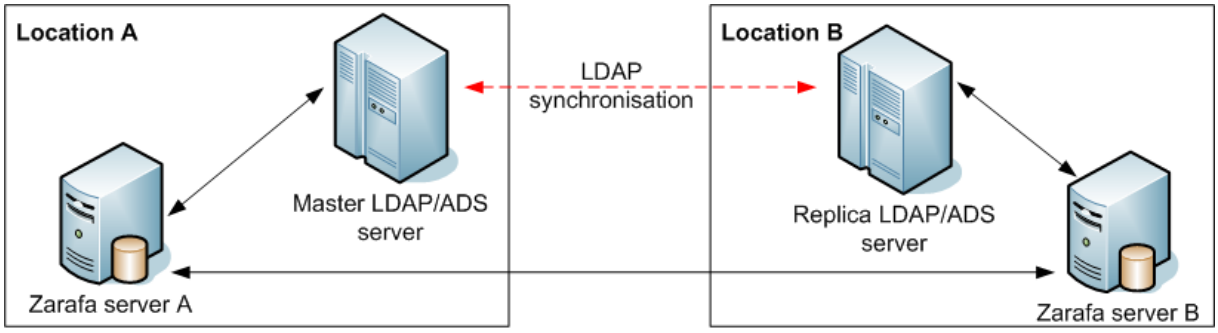


Figura 6.3. Ambiente multi-servidor em dois locais

A caixa de correiro de um usuário sempre é armazenada em um único servidor. Não é possível sincronizar caixas de correio de múltiplos servidores.

Ao acessar múltiplas caixas de correio, as quais estão localizadas em diferentes servidores, o cliente fará uma conexão a diferentes nódulos multi-servidor. Veja como isto acontece em *Figura 6.4, "Ambiente multi-servidor"*.

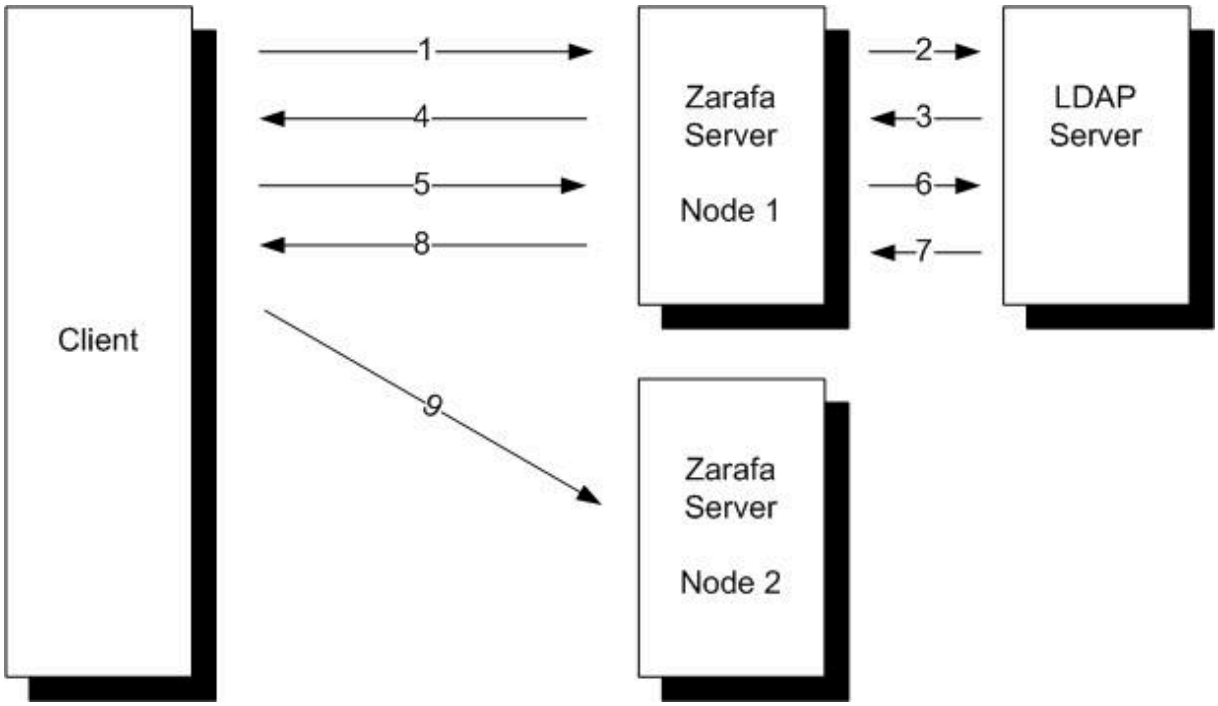


Figura 6.4. Ambiente multi-servidor

O usuário *Joao* está localizado no *Nódulo1* e a usuária *Maria* está localizada no *Nódulo 2*. João tem acesso para leitura na caixa de correio de Maria.

1. *Joao* inicia seu cliente MS Outlook, o qual se conecta ao *Nódulo1*.
2. O Servidor Zarafa *Nódulo1* verifica o atributo Home Server no servidor LDAP central.
3. O Home Server do usuário *Joao* é enviado de volta ao Servidor Zarafa.
4. *John's* mailbox is located on *Node 1*, so the mailbox is loaded.
5. *Joao* envia uma requisição para o Servidor Zarafa para abrir a caixa de correio de *Mary*.
6. O Servidor Zarafa *Nódulo1* verifica o atributo Home Server de *Maria* no servidor LDAP central.
7. O Home Server da usuária *Maria* é enviado de volta para o Servidor Zarafa.

8. Um pedido de redirecionamento é enviado de volta para o cliente

9. The client makes a connection to *Node 2* to open the mailbox of *Mary*.

No exemplo acima o cliente tem uma conexão aberta para ambos os nódulos para acessar as caixas de mensagens.

6.3.2. Preparando / definindo o servidor LDAP para configuração multi-servidor

A versão multi-servidor do Zarafa somente pode ser utilizada com o plugin de usuário LDAP.

In a multi-server setup the Zarafa Server will not only request user and group information from the LDAP server, but also information about the different multi-server nodes.

1. Defina o sevidor LDAP usando *Seção 5.2, "Configurar a integração do ZCP OpenLDAP"* ou *Seção 5.3, "Configurar a integração do Active Directory do ZCP"* neste manual.

2. In the LDAP structure add a folder or organizational unit for each Zarafa Server node in the multi-server setup.

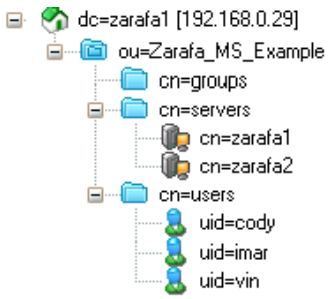


Figura 6.5. Defina o diretório com todos os nódulos dos multi-servidores

3. Add all the multi-server nodes to this directory or organizational unit. In Active Directory the **`Computer`** template can be used for this. When using OpenLDAP a custom LDAP object can be created, with the **`device`**, **`ipHost`** and **`zarafa-server`** *objectClass*.

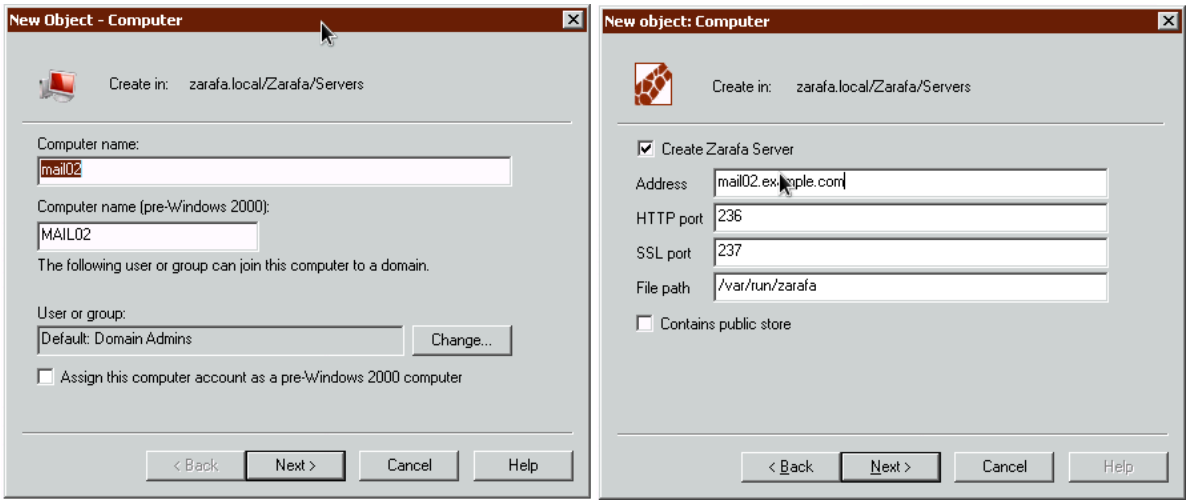


Figura 6.6. Computer creation wizard in ADS

4. Todo nódulo multi-servidor deve ter um **`nome comum`**, **`FQDN`** oo **`ip-address`** e o **`Zarafa server details`**.

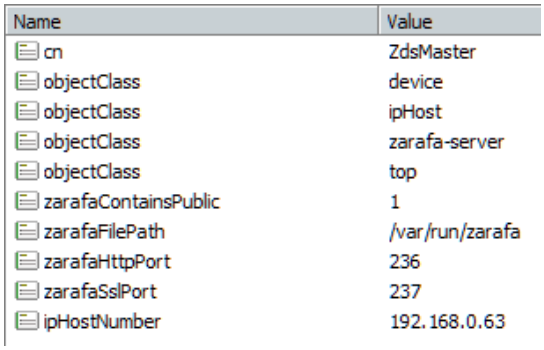


Figura 6.7. Atributos do servidor LDAP

5. The attribute **`ZarafaContainsPublic`** can only be set for one multi-server node at a time. At the moment there is no support for having a single Public Folder onto multiple nodes.

6. The Zarafa LDAP configuration needs to be extended with some extra multi-server configuration options. An example configuration file for the multi-server setup can be found in the **`/usr/share/doc/zarafa-multiserver/example-config`** directory. The files **`ldapms.*.cfg`** are the specific multi-server configuration files. The following LDAP configuration entries need to be configured for a multi-server setup:

```
ldap_server_type_attribute_value = zarafa-server
 ldap_user_server_attribute = zarafaUserServer
 ldap_server_address_attribute = ipHostNumber
 ldap_server_http_port_attribute = zarafaHttpPort
 ldap_server_ssl_port_attribute = zarafaSslPort
 ldap_server_file_path_attribute = zarafaFilePath
 ldap_server_search_filter =
 ldap_server_unique_attribute = cn
```

7. Todo usuário do Zarafa criado no servidor LDAP precisa ser designado para um nódulo de servidor do Zarafa. Isto pode ser feito utilizando o atributo ZarafaUserServer. O atributo deve conter um nome de servidor exclusivo.

In a multi-tenancy situation, all created tenants (companies) in LDAP have to be updated with the **`zarafaCompanyServer`** attribute. Use the server name as well for this.

6.3.3. Configurando os servidores

As seguintes opções de configuração em **`server.cfg`** estão disponibilizadas para suporte a Multi-servidor.

```
enable_distributed_zarafa
```

Habilitar ambiente multi-servidor. Quando definido como **`verdadeiro`** é possível distribuir usuários e companhias para múltiplos servidores. Quando definido para **`falso`**, o ambiente de servidor único é criado.

```
server_name
```

The unique server name used to identify each node in the setup. This server name should be configured correctly in the DNS. This server name should be the same as the value of the **`zarafaUserServer`** attribute.

Para habilitar o suporte ao multi-servidor no Zarafa, modifique as seguintes opções de configuração em **server.cfg**:

```
user_plugin = ldapms
 enable_distributed_zarafa = yes
 server_name = <servername>
 server_ssl_enabled = yes
```

Nota

Uma atualização do servidor único para suporte ao multi-servidor não é uma taréfa fácil. Favor verificar com o Suporte Zarafa se a migração é possível para a configuração utilizada.

6.3.4. Criando certificados SSL

In a multi-server setup, it is required to configure SSL support, because clients like the **zarafa-dagent**, **zarafa-admin**, **zarafa-monitor** need an SSL certifcate to login to the different multi-server nodes.

Primeiramente é necessário criar certificados do lado do servidor, para que o Servidor Zarafa esteja apto a aceitar conecções SSL. Para a autenticação SSL dos clientes Linux, como o **zarafa-dagent**, uma chave privada e púlibca precisam ser criadas.

Siga os passos abaixo para criar tanto os certificados de servidor como os de cliente.

1. Primeiro, crie o diretório que conterá os certificados.

   ```
   mkdir /etc/zarafa/ssl
    chmod 700 /etc/zarafa/ssl
   ```

2. Crie o certificado do servidor utilizando o script **ssl-certificates.sh** no diretório **/usr/share/doc/zarafa**, o qual usa o comando openssl e o script **CA.pl**. Antes que um certificado de servidor possa ser criado, uma raiz CA é necessária. Caso nenhuma raiz CA seja encontrado, o script criará primeiro um CA próprio.

   ```
   cd /etc/zarafa/ssl/
    sh /usr/share/doc/zarafa/ssl-certificates.sh server
   ```

3. Entre com uma senha (passfhase) se você quiser usar uma senha para a chave do servidor. Se uma senha for definida, então esta senha será necessária para assinar requisições de certificados à partir de então. Dê atenção extra ao Nome Comum. Este deve ser o fqdn do servidor. A senha de desafio no final pode ser deixada vazia. Ao final da criação do certificado, o mesmo precisa ser assinado contra o CA. Aceite duas vezes seguidas a pergunta para assinatura e entre com a senha para o CA novamente quando foi necessário.

4. No último passo, o script perguntará se deve mostrar a chave pública desse certificado. isto não é necessário, uma vez que os certificados já foram criados.

5. Após completar o script **ssl-certificates.sh**, o certificado do servidor é criado no diretório atual. O certificado do CA da raiz pode ser encontrado no mesmo diret´roio ou no diretório SSL padrão da distribuição Linux. No Ubuntu o CA da raiz será criado como **./demoCA/cacert.pem**, no RedHat o CA da raiz será criado como **/etc/CA/cacert.pem**. Edite as seguintes linhas em **/etc/zarafa/server.cfg**.

```
server_ssl_enabled = yes
 server_ssl_port = 237
 server_ssl_ca_file = /etc/zarafa/ssl/demoCA/cacert.pem
 server_ssl_key_file = /etc/zarafa/ssl/server.pem
 server_ssl_key_pass = <ssl-password>
 sslkeys_path = /etc/zarafa/sslkeys
```

6. Após um reinício do servidor Zarafa, o servidor deverá aceitar conecções HTTPS. Favor verificar o arquivo de log do servidor à procura de quaisquer erros.

7. Para mais opções sobre certificados SSL favor ver também o arquivo zarafa-server.cfg.

8. Se os certificados do servidor tiverem sido criados com sucesso, os certificados do cliente poderão ser criado da seguinte forma:

```
cd /etc/zarafa/ssl
 sh /usr/share/doc/zarafa/ssl-certificates.sh client
```

9. Fill in all the information, like the server certificate. On some Linux distributions, the Common Name may not be the same as in the server certificate. At the end of the creation, it is required to sign again the certificate against the CA and create a public key for the certificate.

10. Dois certificados de cliente são criados: **`client.pem`** e **`client-public.pem`**. O **`client.pem`** é a chave privada e será utilizada por um cliente (como dagent ou spooler). O **`client-public.pem`** é a chave pública a qual é utilizada pelo servidor.

11. Create **`/etc/zarafa/sslkeys`** and move the public key into it.

```
mkdir -p /etc/zarafa/sslkeys
mv /etc/zarafa/ssl/client-public.pem /etc/zarafa/sslkeys
```

12. Reinicie o **`zarafa-server`** em todos os nódulos para ativar os novos certificados.

```
etc/init.d/zarafa-server restart
```

13. Para testar os certificados SSL do cliente, modifique as seguintes linhas no **`/etc/zarafa/dagent.cfg`**.

```
server_socket = https://127.0.0.1:237/zarafa
 sslkey_file = /etc/zarafa/ssl/client.pem
 sslkey_pass = <ssl-client-password>
```

Após os certificados tiverem sido configurados será possível então enviar emails usando a conexão ssl com a chave privada do dagent, neste teste no servidor local.

```
zarafa-dagent -v -c /etc/zarafa/dagent.cfg <username_on_this_node>
Subject: test email
Test
<ctrl-d>
```

Quando se conecta através de ssl, o dagent verificará o CA privado com o CA da raiz. Nos sistemas baseados no Red Hat ons nomes de arquivos criados aleatoreamente devem vir dos certificados da raiz:

```
yum install openssl-perl
cp /etc/CA/cacert.pem /etc/pki/tls/certs/zarafa-ca.pem
c_rehash /etc/pki/tls/certs
```

Desta forma o dagent é capaz de verificar a chave privada contra o pacote de CA. Nos sistemas Debian este passo pode ser ignorado.

14. Se o teste tiver sucesso, é possível mudar o seguinte valor no dagent.cfg de volta para:

```
server_socket = file:///var/run/zarafa
```

15. Organize todos os certificados nos diferentes nódulos multi-servidor:

```
scp -r /etc/zarafa/ssl /etc/zarafa/sslkeys root@node2:/etc/zarafa/
```

Lembre-se de copiar o CA da raiz para os diferentes nódulos se este arquivo for colocado fora dos diretórios que acabaram de ser copiados.

16. Repita os passos acima para configurar **`server.cfg`** e **`dagent.cfg`** em todos os nódulos diferentes. Nos nódulos baseados no Red Hat também adicione o CA da raiz no pacote de CA. Quando feito, teste a taxa de envio remoto com:

```
zarafa-dagent -v -c /etc/zarafa/dagent.cfg <username_on_other_node>
Subject: test email
Test
<ctrl-d>
```

Este envio não deve resultar em nenhum erro de envio, caso contrário verifique os certificados criados. Agora é possível enviar email de um MTA central para diferentes nódulos multiservidor.

Os certificados SSL do cliente podem ser utilizados para as seguintes ferramentas para se conectar a um servidor remoto do Zarafa:

```
zarafa-dagent
zarafa-spooler
zarafa-backup, zarafa-restore
zarafa-admin
```

Para ambientes multi-servidor avançados e a melhor configuração do Zarafa para uma definição específica, os Zarafa Professional Services estão abertos para aconselhamento e suporte.

6.4. Atualizador do Zarafa Windows Client

O ZCP contém um mecanismo que permite aos Clientes do Zarafa Windows atualizarem-se para a última versão disponível.

Nota

O Zarafa Windows Client Updater somente está disponível nas edições ZCP Professional ou Enterprise.

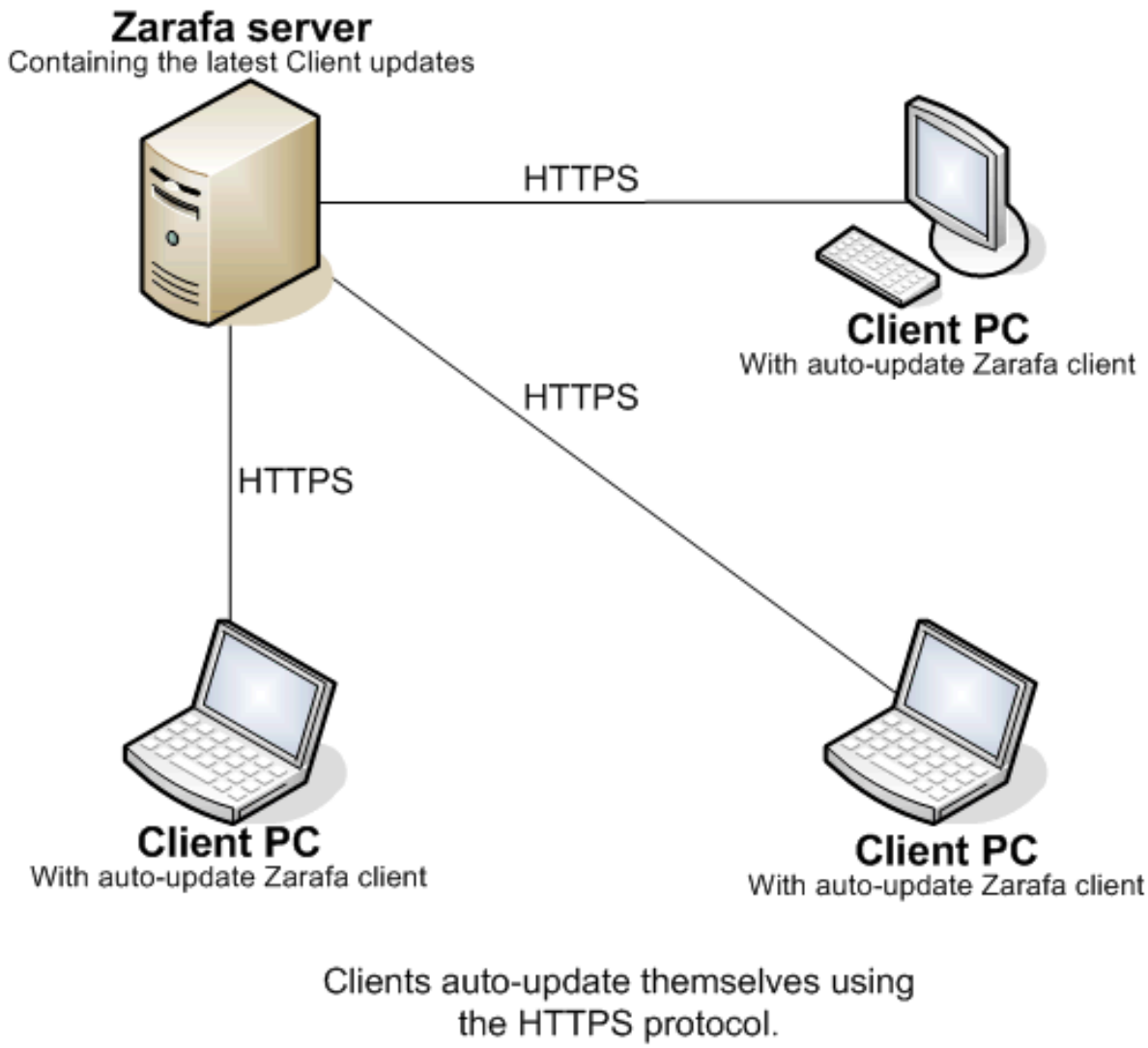


Figura 6.8. Estrutura do Auto-update

Restrições:

- O mecanismo de auto-update não suporta da habilidade de instalar uma versão anterior do cliente, ele sempre atualizará o Zarafa Windows Client para a versão mais alta disponível.
- The Zarafa Windows Client Updater is not available for Windows 2000 or earlier releases. = The Zarafa Windows Client Updater can not automaticly switch between 32bits and 64 bits installations.

6.4.1. Configuração do servidor de apoio

O Zarafa Windows Client Updater pode ser habilitado definindo-se o valor seguinte para **`yes`** no **`server.cfg`** do zarafa-server:

```
client_update_enabled = yes
```

Quando um **`zarafa-server`** é atualizado, ele copiará o mais recente instalador do cliente para o local que for especificado no arquivo de configuração do servidor **`server.cfg`**, como mostrado abaixo.

```
client_update_path = /var/lib/zarafa/client
```

O cliente de auto-update pode enviar a informação de acesso para o servidor. Se o atualizador não funcionar, então os arquivos de acesso serão enviados para o servidor por padrão. Este comportamento pode ser modificado com a seguinte definição: client_update_log_level = 1

As seguintes opções podem ser definidas: 0 desabilitado 1 envia somente os arquivos de cesso para o servidor quando um erro ocorre 2 sempre envia os arquivos de acesso para o servidor

Os arquivos de acesso recebidos do cliente de auto update são colocados no seguinte local no servidor: client_update_log_path = /var/log/zarafa/autoupdate

As atualizações na pasta de atualização do cliente seguem uma convenção para nomeação. O Zarafa Server funcionará somente com aquelas atualizações que se aderem a esta convenção:

```
zarafaclient-<major version>.<minor version>.<update number>-<build number>.msi
```

For example **`zarafaclient-7.1.5-42673.msi`** is a valid name of an update.

Baseado na convenção de nomeação, o Zarafa Windows Client Updater descobre se uma atualização do software do cliente está disponível. Se um cliente envia uma requisição para receber uma nova versão, o **`zarafa-server`** enviará o novo pacote de atualização para o cliente, para que ele posse se atualizar para a última versão.

Nota

Se o perfil padrão for definido para usar encriptação via porta **237**, o certificado do CA da raiz precisará ser instalado no aparelho utilizado.

6.4.2. Configuração do servidor de apoio

O mecanismo de auto-update do Zarafa Windows Client consiste de um aplicativo para iniciar o processo de auto-update pelo nome de **`ZarafaLaunchUpdater.exe`** e um serviço Windows conhecido como**`ZarafaUpdaterService.exe`**.

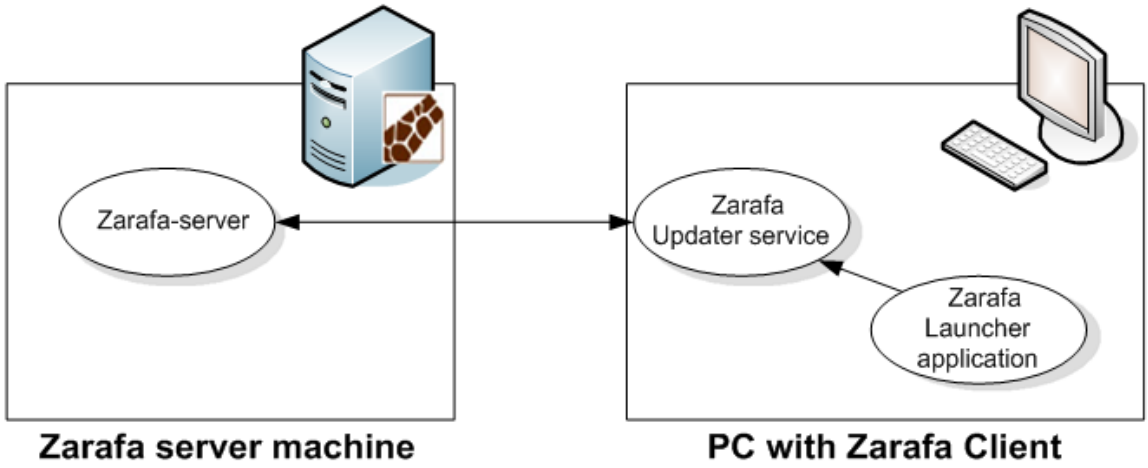


Figura 6.9. Estrutura do Auto-update

O aplicativo Launch Updater será iniciado com a inicialização do Windows. O comando para rodar o aplicativo está localizado no registro aqui:

```
HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Run
```

Este aplicativo descobrirá a versão atualmente instalada através da seguinte chave de registro:

```
HKEY_LOCAL_MACHINE\Software\Zarafa\Client\Version
```

Esta chave de registro contém a versão atual do Zarafa Windows Client instalada na máquina.

O aplicativo Launch Updater irá ler o perfil padrão do Outlook no registro para juntar as credenciais necessárias para se conectar ao Servidor Zarafa. Ele informa ao Servidor Zarafa qual versão do Zarafa Windows Client está atualmente instalada, o Servidor Zarafa responde com um Zarafa Windows Client mais novo no caso deste existir.

6.4.2.1. Zarafa Updater Service

O Zarafa Updater Service como uma conta de sistema local. Assim sendo, ele tem todos os privilégios necessários para instalar o Zarafa Windows Client na máquina.

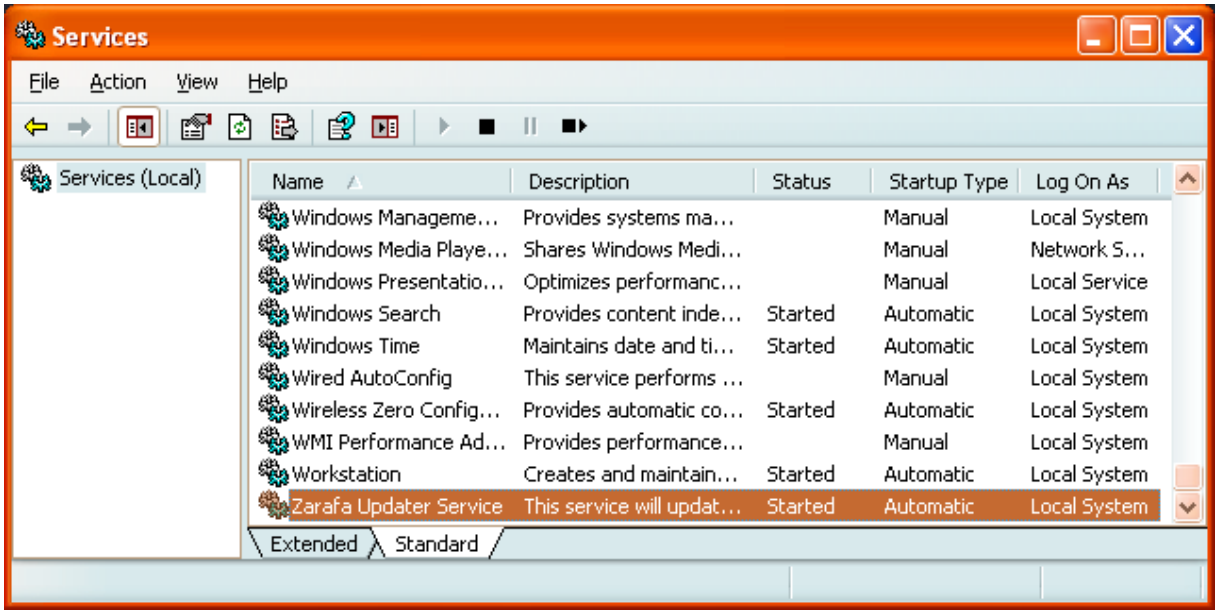


Figura 6.10. Serviços

O Zarafa Updater Service esperará em um canto até que o Zarafa lance o aplicativo de atualização para enviar a ele a versão atual do cliente e os detalhes do Servidor Zarafa com o qual se conectará. Se houver uma atualização apropriada, o serviço o baixa para **`c:\windows\temp\zarafaclient.msi`**. O Zarafa Updater Service lança esta atualização para instalação em um modo silencioso.

Embora todo o processo de atualização seja silencioso, registros serão gerados para solução de problemas. O registro do serviço de atualização estará escrito no diretório **`All users\Application data\`** e o registro de inicialização do atualizador estará escrito no diretório **`<user>\Application data\`**.

Quando o Updater Service iniciar a atualização do cliente, ele criará os arquivos zarafa-<trackid>.log e zarafa-<trackid>msi.log no diretório **`<user>\Local Settings\Temp`**. Estes arquivos são enviados para o servidor dependendo das definições do servidor.

Nota

O cliente somente encontrará atualizações com sucesso se o perfil padrão do Outlook estiver configurado para trabalhar com um Servidor Zarafa e se atualizações estiverem disponíveis naquele servidor. Mesmo com a definição de prompt `*prompt for the profile to be used'* o Zarafa Windows Client Updater terá sucesso desde que o menu drop-down (acinzentado) especifique o perfil configurado para o Zarafa. Favor ver o manual do usuário sobre como configurar perfis do Outlook.

6.4.2.2. Status do Zarafa Updater

O **`zarafa-server`** informa o status do atualizador do cliente Zarafa no **`server.log`**. O **`zarafa-admin`** informa o último status da atualização do cliente. Utilizando o comando seguinte, pode-se ver a informação da atualização por usuáriuo: zarafa-admin --details <user>

+

```
 Client update Information:
Trackid:                1889610488
Last update:            <date>
From version:           <version>
To version:             <version>
Computername:           <name>
Update:                 Succeed
```

Quando uma atualização de cliente falha, os arquivos de registro são salvos no diretório configurado no **`server.cfg`** campo **`client_update_log_path`** (por padrão, isto está definido para /var/log/zarafa/autoupdate). O valo de trackid pode ser usado para encontrar os arquivos de registro, por exemplo: /var/log/zarafa/autoupdate/0x70A12AF8/

+

zarafa-autoupdate.log zarafa-msi.log

6.4.3. Opções MSI

Se preferir retirar a instalação do cliente Zarafa do seu servidor de domínio do Windows, você provavelmente deseja que a instalação não instale o Zarafa Update Service. As seguintes opções podem ser utilizadas para isso:

ADDDEFAULT="Client"
: Isto fará o instalador somente instalar a parte do Cliente Outlook e não o Updater Service. Para instalar esta característocas, adicione Updater nesta opção.

APPDIR=D:\Zarafa\Client
: Para mudar o caminho padrão da instalação, use a variável APPDIR. Deixe esta opção para instalar normalmente no diretório "Program Files".

/q
: Faça a instalação silenciosa. Nenhuma interface gráfica será mostrada. Para mostrar o progresso da instalação, use o modificador b (para gui básico) ou r (para gui reduzido). Caso você mostre o gui completo (modificador f), ele será interativo.

Execute **msiexec** para ver uma lista de outras opções que podem ser usadas. Para uma instalação automatizada típica, use o seguinte comando:

```
msiexec /i zarafaclient.msi ADDDEFAULT=Client /q
```

6.5. Armazenagem de anexos em caso singular

Desde o ZPC 6.30 o Servidor Zarafa oferece o Single Instance Attachment Storage para evitar armazenamento repedido de anexos. Esta característica, como o nome implica, somente mantém uma cópia de cada anexo quando uma mensagem é enviada a múltiplos destinatários dentro do mesmo servido. Este mecanismo, então, minimisa os requisitos de espaço em dico e aumenta notavelmente a eficiência de entrega quando as mensagens com anexos enviadas para grandes listas de distribuição.

Imaginemos a seguinte situação: o usuário A pertence a um servidor Zarafa; ele envia uma mensagem com 10 MB de anexos para 30 usuários que residem no mesmo servidor. Em uma situação normal, 30 cópias dos arquivos seriam salvos na base de dados, levando a um uso ineficiente do espaço de armazenamento (310 MB de dados). Com o armazenamento de anexos de caso único, somente uma cópia de cada anexo é salva na base de dados (somente 10 MB de datos neste exemplo) e todos os 30 usuários podem acessar o anexo através de um indicador de referência.

Nota

Anexos de caso singular são acessíveis entre locatários (companhias) bem como (mesmo quando os locatários não podem ver um ao outro) o locatário1 e 50 usuários do locatário2, sendo que os locatários precisam residir no mesmo servidor, somente uma cópia do anexo é salva.

Nota

Anexos de caso singular serão geridos por servidor, quando enviar um email com anexo para múltiplos usuários Zarafa espalhados em vários servidores, cada servidor receberá seu próprio anexo de caso singular.

6.5.1. Armazenagem de anexos em caso singular e LMTP

To use the Single Instance Storage it's required to use the LMTP delivery method executed from the **virtual_transport** in Postfix.

Com a já mencionada configuração, emails recebidos de fora com um anexo enviado para múltiplos usuários internos serão processados com eficiência salvando o anexo somente uma vez.

The usage of **virtual_transport** in Postfix will deliver only one email with a list of the internal users to the dagent instead of one email per internal user. Without virtual transport option, Single Instance can not know that the attachment is similar in the email item(s).

6.6. Rodando o ZPC Services com privilégios de usuário regulares

Normalmente os seviços do Zarafa são rodados como raiz. Desde a versão 5.0 há a opção para mudar o usuário para o qual o serviço roda e ainda assim iniciar os serviços como raiz. Contudo, há muitas coisas a fazer antes que os serviços possam rodar perfeitamente como um usuário que não seja da raiz.

Se o **log_method** for definido para **file**, verifique ao certo se o usuário ou grupo para o qual o serviço esteja rodando possa escrever nele. Quando um logrotate acontece, por enviar ao serviço o sinal HUP, um novo arquivo é criado, o qual será de propriedade do usuário para o qual o serviço está sendo rodado.

O serviço ainda deverá ser inicializado como raiz já que ele criará um arquivo pid sob o local do sistema **/var/run**, e abrirá as conexões de rede as quais provavelmente tem um súmero abaixo de **1024**, o qual somente pode ser aberto como **root**.

O seguinte exemplo mostra como configurar o **zarafa-server** para rodar como usuário **zarafa** e grupo **zarafa**:

```
addgroup --system zarafa
adduser --system ---home /dev/null ---no-create-home \
       --ingroup zarafa \
       --disabled-password --gecos 'Zarafa services' \
       --shell /bin/false zarafa
mkdir /var/log/zarafa
chown zarafa:zarafa /var/log/zarafa
chown zarafa:zarafa /etc/zarafa/report
chown -R zarafa:zarafa /var/lib/zarafa
```

Nota

As ferramentas addgroup e adduser podem ter sintaxe diferente em distribuições diferentes.

Edite as opções **run_as_user** e **run_as_group** no arquivo **server.cfg**, e defina ambos para o Zarafa. Tenha certeza que a opção **local_admin_users** ainda contenha raiz como um usuário administrativo, para que a ferramenta **zarafa-admin** possa ainda ser utilizada. Caso contrário **su** ou **sudo** precisa ser usado cada vez que a ferramenta **zarafa-admin** for iniciada.

6.7. Single Sign On com o ZCP

Este capítulo descreve como configurar um ambiente de Single Sign On com o ZCP, para que os usuários possam autenticar sem precisar de senha. O ZCP suporta ambos os protocolos de autenticação NTLM e Kerberos. O suporte ao Kerberos está disponível à partir do ZCP 6.40.2 e superiores.

Os dois métodos serão descritos nas seções seguintes.

6.7.1. NTLM SSO com ADS

6.7.1.1. Instalando o software Linux

O seguinte software precisa ser instalado:

- **winbind**
- **kinit**

Dependendo da distribuição Linux usada, isso vem através de diferentes nomes dos pacotes. No Debian use:

```
apt-get install krb5-user winbind
```

krb5-user também instalará os arquivos de configuração da biblioteca do Kerberos em **/etc**. O pacote **winbind** depende do **samba-common**, o qual será também instalado. No Linux Red hat Enterprise ambos os pacotes **krb5-workstation** e o **samba-common** são necessários para isto.

Para habilitar o NTLM SSO com o ZCP defina a seguinte opão em **/etc/zarafa/server.cfg**:

```
enable_sso  = yes
```

6.7.1.2. ADS: configuração de rede específica

Os seguintes pré-requisitos são necessários antes de prosseguir:

- Todo servidor precisa ter um nome DNS, para que seus endereços de IP possam ser encontrados pelo DNS.
- A hora de todos os servidores precisa estar sincronizada. A hora não poder ter intervalo de alguns minutos.

Este documento tem os seguintes nomes como exemplo:

- **FQDN** doservidor Windows ADS: **ADSERVER.ADSDOMAIN.LOCAL**. Desta forma, o servidor windows é nomeado: **ADSERVER**, o reino é **ADSDOMAIN.LOCAL**, e o nome de domínio é **ADSDOMAIN**. Estações de trabalho podem, então, tanto participar do domínio usando o **ADSDOMAIN** ou o nome **ADSDOMAIN.LOCAL**.
- O **FQDN** do servidor Linux é **LINUXSERVER.LOCAL**. Este nome não importa muito, enquanto ele é for manuseado pelo servidor DNS.

6.7.1.3. Configurando a biblioteca do Kerberos

Primeiro vamos configurar a biblioteca do Kerberus. O arquivo de configuração é **/etc/krb5.conf**. Sob a seção **libdefaults**, defina:

```
default_realm = ADSDOMAIN.LOCAL
```

Sob a seção **realms**, adicione o reino do windows:

```
[realms]
ADSDOMAIN.LOCAL = {
        kdc = 192.168.0.100
        admin_server = 192.168.0.100
        password_server = 192.168.0.100
        default_domain = ADSDOMAIN.LOCAL
}
```

Aqui, **192.168.0.100** é o endereço de IP para o servidor de domínio do Windows ADS.

Agora que a biblioteca do Kerberos está configurada, é possível ter acesso utilizando o **kinit** no servidor linux:

```
kinit Administrator
```

Será perguntada uma senha:

```
Senha do Administrator@ADSDOMAIN.LOCAL:
```

Digite a senha de administrador aqui e um ticket do Kerberos será dado pelo servidor ADS.

6.7.1.4. Prticipando do domínio ADS

Primeiro configuraremos o samba. Edite o arquivo **`/etc/samba/smb.conf`** e adicione/defina as seguintes opções:

Para Samba < 3.4

```
[global]
realm = ADSDOMAIN.LOCAL
use kerberos keytab = true
security = ads
```

Para Samba >= 3.4

```
[global]
realm = ADSDOMAIN.LOCAL
kerberos method = dedicated keytab
dedicated keytab file = /etc/krb5.keytab
security = ads
```

The value of **`kerberos method`** may also be set to **`system keytab`**, and **`dedicated keytab file`** may be left out. Please consult the **`smb.conf(5)`** manual page for more information about these settings.

Com este ticket podemos participar do domínio Windows, sem ser necessário digitar a senha novamente:

```
net ads join
```

ou, se isto não funcionar:

```
net ads join -S ADSDOMAIN -U Administrator
```

Este comando pode também ser diferente para diferentes versões do Samba. Se este comando pedir uma senha, algo está errado e deve ser fechado com Ctrl-C. Quando tudo dá certo, a seguinte mensagem é mostrada na tela:

```
Joined 'LINUXSERVER' to realm 'ADSDOMAIN.LOCAL'
```

ou alguma outra mensagem de sucesso.

Agora é necessário reinicial o winbind daemon, pois ele retém muitos itens armazenados:

```
/etc/init.d/winbind restart
```

E está feito. Para testar se a autenticação de fato funciona, tente o seguinte comando:

```
ntlm_auth --username=john
```

Onde **`john`** é um usuário em um servidor ADS.

O programa solicitará uma senha. Após digitá-la, deve aparecer a mensagem:

```
NT_STATUS_OK: Success (0x0)
```

Se este passo não funcionar, tente reiniciar o **winbind**, verifique os nomes DNS, verifique com **strace** o que o **ntlm_auth** tenta fazer, veja com o **tcpdump** se há tráfego de fato na rede do **ntlm_auth** para o servidor de domínio e outras ferramentas de debug de nível inferior.

6.7.2. NTLM SSO with Samba 3

6.7.2.1. Instalando o software Linux

O seguinte software precisa ser instalado no servidor ZCP:

```
winbind
```

Dependendo da distribuição Linux usada, isso vem através de diferentes nomes dos pacotes. No Debian use:

```
apt-get install winbind
```

No Linux da Red Hat Enterprise o pacote **samba-common** é necessário para isto.

Para habilitar o NTLM SSO com ZCP defina o seguinte no arquivo **/etc/zarafa/server.cfg**:

```
enable_sso  = yes
```

6.7.2.2. Participando do domínio

Agora o servidor precisa fazer parte do domínio Samba executando o seguinte comando:

```
net rpc join
```

Termine digitando a senha de administrador. Se tiver sucesso, aparecerá a mensagem:

```
Joined domain <DOMAIN>
```

A configuração SSO agora está feita. Para testar se a atenticação de fato funciona, tente o seguinte comando:

```
ntlm_auth --username=john
```

Onde **john** é um usuário do Samba válido.

O programa solicitará uma senha. Após digitá-la, deve aparecer a mensagem:

```
NT_STATUS_OK: Success (0x0)
```

Se este passo não funcionar, tente reiniciar o **winbind**, verifique os nomes DNS, verifique com **strace** o que o **ntlm_auth** tenta fazer, veja com o **tcpdump** se há tráfego de fato na rede do **ntlm_auth** para o servidor de domínio e outras ferramentas de debug de nível inferior.

6.7.3. SSO com Kerberos

6.7.3.1. Requisitos e Convenções

- O servidor que roda o ZCP precisa ter o programa MIT Kerberos instalado.

- O ZCP versão 6.40.2 ou superioe precisa ser instalado para o SSO com Outlook.
- Todo servidor precisa ter um nome DNS, para que seu endereço de IP possa ser encontrado pelo DNS. Também é necessário que todos os servidores tenham um registro de PTR.
- A hora de todos os servidores precisa estar em sincronia.

Este documento tem os seguintes nomes como exemplo:

- FQDN do servidor de diretório ativo do Windows: **`ADSERVER.ADSDOMAIN.LOCAL`**. Assim, o servidor windows é nomeado: **`ADSERVER`**, o reino é **`ADSDOMAIN.LOCAL`**, e o nome do grupo de trabalho é **`ADSDOMAIN`**.
- O FQDN do Servidor Zarafa é **`ZARAFA.LINUXDOMAIN.LOCAL`**.

Neste exemplo, o Servidor Zarafa está situado em um domínio diferente. Isto não é um requerimento, mas deixa o documento um pouco mais claro sobre como criar o Kerberos principal.

6.7.3.2. Configuração do Active Directory

Crie dois Kerberos principais no Active Directory, um para o SSO com WebAccess e outro para o SSO com Outlook.

1. Adicione um novo usuário **`httpd-linux`** ao Active Directory (este usuário será utilizado para criar o sistema principal para SSO com WebAccess, o nome do usuário pode ser diferente).
2. Adicione um novo usuário **`zarafa-linux`**ao Active Directory (este usuário será utilizado para criar o sistema principal para SSO com Outlook, o nome do usuário pode ser diferente).
3. Tenha certeza que a opção *Password never expires* esteja habilitada.
4. Nas propriedades da conta para estes usuários, habilite: *Use DES encryption types for this account*.
5. Após definir esta propriedade da conta, é fortemente aconselhável resetar a senha para estes usuários.

Nota

The following commands will use the ktpass.exe utility, which should be installed by default when the ActiveDirectory is running on the same machine. In any other case you can find it with the "Windows Support tools" on the install cd or download them from the Microsoft website.

Nota

Ao criar um keytab no Windows Server 2008, tenha certeza de especificar **`RC4-HMAC-NT`** como o cripto, *-mapop set +desonly* deve ser deixado de fora.

Execute o seguinte comando para criar o arquivo de keytab para o servidor Apache:

```
ktpass.exe -princ HTTP/fqdn@REALM -mapuser account@DOMAIN
 -crypto DES-CBC-MD5 -ptype KRB5_NT_PRINCIPAL
 -mapop set +desonly -pass <password> -out c:\keytab.apache
```

or for Windows Server 2008:

```
ktpass -princ HTTP/fqdn@REALM -mapuser account@DOMAIN
 -crypto RC4-HMAC-NT -ptype KRB5_NT_PRINCIPAL
 -pass <password> -out c:\keytab.apache
```

Execute o seguinte comando para criar o arquivo de keytab para o Servidor Zarafa:

```
ktpass.exe -princ fqdn@REALM -mapuser account@DOMAIN
 -crypto DES-CBC-MD5 -ptype KRB5_NT_PRINCIPAL
 -mapop set +desonly -pass <password> -out c:\keytab.zarafa
```

or for Windos Server 2008:

```
ktpass -princ fqdn@REALM -mapuser account@DOMAIN
 -crypto RC4-HMAC-NT -ptype KRB5_NT_PRINCIPAL
 -pass <password> -out c:\keytab.zarafa
```

- Copy the file **`keytab.apache`** to **`/etc/apache2`** (Deban and Ubuntu) or **`/etc/httpd/`** (RHEL & SLES) on the Linux server.
- Copie o arquivo **`keytab.zarafa`** para **`/etc/zarafa/`** no servidor Linux.

6.7.3.3. Configuração do Kerberos

Abra o arquivo **`/etc/krb5.conf`** e insira as linhas seguintes:

```
[libdefaults]
                              default_realm = ADSDOMAIN.LOCAL
                              default_tgs_enctypes = des-cbc-md5 arcfour-hmac-md5
                              default_tkt_enctypes = des-cbc-md5 arcfour-hmac-md5
                              permitted_enctypes = des-cbc-md5 arcfour-hmac-md5

[realms]
                            ADSDOMAIN.LOCAL = {
                                                        kdc = adserver.adsdomain.local
                                                        admin_server =
 adserver.adsdomain.local
                            }

[domain_realm]
     .adsdomain.local = ADSDOMAIN.LOCAL
     adsdomain.local = ADSDOMAIN.LOCAL
```

Configurando o ZCP para Kerberos SSO com Outlook

Adicione a linha seguinte à seção **`[libdefaults]`** do **`/etc/krb5.conf`**:

```
default_keytab_name = /etc/zarafa/keytab.zarafa
```

6.7.3.4. Configuração do Servidor Zarafa

Para habilitar o Outlook SSO com o ZCP defina o seguinte no arquivo **`server.cfg`**:

```
enable_sso = yes
```

Se o nome de hospedagem do servidor Linux (ver o comando **`hostname`**) não for igual ao FQDN do servidor Linux, a variável **`server_hostname`** precisará ser mudada no arquivo **`server.cfg`**:

```
server_hostname = zarafa.linuxdomain.local
```

Reinicie o zarafa-server para ativar as alterações.

```
reinicialização de service zarafa-server
```

6.7.3.5. Apache configuration (for SSO with WebAccess/WebApp)

Install the **`mod_auth_kerb/libapache2-mod-auth-kerb`** Apache module, e.g. for Red Hat:

```
yum install mod_auth_kerb
```

For Debian/Ubuntu: apt-get install libapache2-mod-auth-kerb

Open the vhost configuration of WebAccess/WebApp and add the following lines to the <Directory> directive:

```
<Directory /usr/share/zarafa-webaccess>
  AuthType Kerberos
  AuthName "Kerberos Login"
  KrbMethodNegotiate On
  KrbMethodK5Passwd Off
  KrbServiceName HTTP
  KrbAuthRealms ADSDOMAIN.LOCAL
  Krb5KeyTab /etc/httpd/keytab.apache
  require valid-user
</Directory>
```

Defina as permissões do arquivo de sistema do arquivo keytab para 400 e mude o proprietário para o usuário do Apache:

```
chmod 400 /etc/httpd/keytab.apache
chown apache:apache /etc/httpd/keytab.apache
```

Reinicie o Apache para ativar as alterações, por exemplo, para Redhat:

```
service httpd restart
```

6.7.3.6. WebAccess/WebApp configuration

To setup a Single Sign On environment for Zarafa Collaboration Platform, it's necessary to make a trust between the Apache webserver and the Zarafa Storage Server. The trust is necessary to manage the user authentication through the webserver and not anymore through Zarafa.

There are two ways to establish this trust. The first option is to have the system user running the Apache process acting as an administrator within Zarafa, which can only be recommended when Zarafa is running on the same systen and no other applications (for instance Z-Push) are running on the same server. The second option is to use ssl client certificates (see *Seção 6.3.4, "Criando certificados SSL"*) to establish this trust only for a specific web application.

Using client certificates for authentication

To use ssl client certificates for authentication (see *Seção 6.3.4, "Criando certificados SSL"*) the client certificate has to be readable by the user of the webserver. Afterwards the DEFAULT_SERVER, SSLCERT_FILE and SSLCERT_PASS has to be changed in the **`config.php`** of WebAccess/WebApp.

```
// Default Zarafa server to connect to.
define("DEFAULT_SERVER","https://localhost:237/zarafa");
```

```
// When using a single-signon system on your webserver, but Zarafa is on another server
// you can use https to access the zarafa server, and authenticate using an SSL certificate.
define("SSLCERT_FILE", "/usr/share/zarafa-webapp/zarafa-client.pem");
define("SSLCERT_PASS", mypassword);
```

Running the webserver as an administrator

To have the webserver act as an administrator, the user running the webserver process has to be added on the following line of the **`server.cfg`**:

```
local_admin_users = root apache
```

Typical users are apache for RHEL, www-data for Debian/Ubuntu and wwwrun for SLES.

Nota

This method will only work, when the WebAccess/WebApp is running on the same server as Zarafa.

Restart the zarafa-server processes to activate this change, e.g. for Red Hat:

```
reinicialização de service zarafa-server
```

Atenção

Setting the webserver als local_admin_user will allow other applications running on the same server to log in with admin privileges as well. As passwords will no be checked for admin users this means, that user will be able to log in with any password!

Common steps

As the passed user in Single Sign On environments also contains the domain/realm (e.g. *user@domain*[2]), the WebAccess/WebApp has to remove this before logging in. This can be configured in the **`config.php`** file:

```
define("LOGINNAME_STRIP_DOMAIN", true);
```

6.7.3.7. Configuração do navegador

Antes que o Single Sign On possa ser usado em um navegador, configure as seguinte definições:

Firefox

1. Digite na barra de endereço **`about:config`**

[2] mailto:user@domain

2. Filtro no **`auth`**

3. Mude as opções: **`network.negotiate-auth.trusted-uris`** e **`network.negotiate-auth.delegation-uris`** para **`.testdomain.com`**

Internet Explorer

1. Vá em *Tools* **>** *Internet options* **>** *Advanced*

2. Tenha certeza que a opção *Enable integrated Windows authentication* esteja habilitada

3. Adicione a url do Servidor Zarafa (*http://zarafa.linuxdomain.local*) aos sites de *Local Intranet*.

Reinicie o navegador e abra o WebAccess via o FQDN (*http://zarafa.linuxdomain.local/webaccess*). Se a configuração for feita corretamente, o usuário terá acesso ao WebAccess sem digitar o nome de usuário e senha.

6.7.4. Funcionando

Agora que o SSO parece funcionar com o servidor Linux, ele será automaticamente usado pelo **`zarafa-server`**. Agora acesse uma estação de trabalho do Windows no domínio e crie um novo perfil do Outlook para o usuário que acabou de entrar, mas deixe o campo de senha vazio. O Outlook deve criar o perfil sem a senha.

6.8. Rastreando mensagens com o Zarava Archive

Esta seção fornece informações sobre como controlar todas as mensagens enviadas e recebidas usando a tecnologia de arquivamento do Zarafa. Isto pode ser útil em ambientes de e-mails mais estritos onde é importante ser capaz de ver o que foi enviado e recebido independentemente do que o *proprietário* das mensagens fez com elas.

6.8.1. Arquivo na entrega

Arquivo no momento da entrega é o processo de certificar-se cada mensagem que é recebida também será colocada em cada arquivo anexado. Se a mensagem não pode ser arquivado ela **não** vai ser entregue. Em vez disso, irá resultar em uma falha temporária, fazendo o MTA tentar novamente entregar a mensagem em um momento posterior.

Arquivo na entrega é implementado pelo processo **`zarafa-dagent`** e pode ser controlado com a opção de configuração **`archive_on_delivery`** no arquivo de configuração do dagent.

For Archive on delivery to work, an archive configuration file needs to be present in the Zarafa configuration directory. In this configuration file settings for sslkey_file and sslkey_pass must me set to values such that Zarafa server can contact the archvier server sucessfully.

Quando uma mensagem é arquivada com o método de arquivar na entrega, ela se tornará uma entrada do arquivo regular, significando que as regras normais se aplicam. Isto significa que se um usuário move a mensagem na pasta principal, a mensagem também será movida no arquivo. Isto inclui movimento para a caixa de lixo.

Importante

Quando uma mensagem é excluída da pasta primária, a mensagem **não** é excluída do arquivo. Em vez disso, ela é movida para uma pasta especial Itens Apagados no arquivador.

Atenção

Due to the current implementation of the Dagent messages that are moved by a rule will sadly be skipped during any subsequent archiving.

6.8.2. Arquivar ao enviar

Arquivar ao enviar é o processo de certificar se cada mensagem que está sendo enviada pelo spooler também será colocado em cada arquivo anexado. Se a mensagem não for arquivada, ela **não** será enviada. Em vez disso ela irá retornar uma mensagem de falha para o usuário.

Arquivo em enviar é implementado pelo processo **`zarafa-spooler`** e pode ser controlado com a opção de configuração **`archive_on_send`** no arquivo de configuração do spooler.

Importante

E-mail que é enviado diretamente para um servidor SMTP (normalmente quando utilizando uma conta IMAP) não será arquivado diretamente porque o **`zarafa-spooler`** não está envolvido no processo de envio nesta situação.

Quando uma mensagem é arquivada com o método de arquivar ao enviar, ela torna-se um arquivo separado. Isso significa que ela não tem nenhuma referência à mensagem original na pasta principal. Também não há nenhuma mensagem na pasta principal que irá incluir uma referência à mensagem arquivada.

Nota

A menos que seja desabilitado, as mensagens na pasta de itens enviados são arquivados como qualquer outra mensagem. Armazenamento adicional é necessário porque as mensagens também foram arquivadas pelo spooler.

6.9. Zarafa Python plugin framework

The Zarafa Spooler and the Zarafa Dagent support the Zarafa python plugin framework. This framework makes it easier for advanced system administrators and developers to integrate systems with the spooler and dagent. The advanced system administrator and developer can easily add new functionality or change some behaviours of the existing system. The plugin framework is based on the programming language Python which means that you need to create your own hook in python.

6.9.1. How it works

If the plugin framework in the spooler or dagent is enabled it will search for python files in the directory **`plugin_path`** and look for a specific type of plugin. If the plugins are found it will be verified and loaded. Everytime the spooler or dagent is called it will execute the hooks based on priority. Plugins can validate and change a message on different stages of the spooler and dagent process.

6.9.2. General Options

The options for the python plugin framework are for every client the same except the file locations, see *Tabela 6.1, “Table Python plugin framework options”*

Tabela 6.1. Table Python plugin framework options

Option	Default	Description
plugin_enabled	yes	Enable the plugin framework in the specfic component
plugin_manager_path	/usr/share/zarafa-**<componentname>**/python	Path to the plugin manager.
plugin_path	/var/lib/zarafa/**<componentname>**/plugins	Path to the activated plugins.

The value **<componentname>** can be *dagent* or *spooler*

6.9.3. How to use

After the installation of the component zarafa-dagent or zarafa-spooler, it is possible to activate a plugin. The default plugins are installed in the directory '/usr/share/zarafa-**<componentname>**/python/plugins/'. To activate a plugin, create a symbolic link in the **plugin_path** directory to the plugin which you want to activate. For example, to activate the disclaimer plugin in the spooler, run the following command:

```
ln -s /usr/share/zarafa-spooler/python/plugins/disclaimer.py /var/lib/zarafa/spooler/plugins/
disclaimer.py
```

6.9.4. Zarafa-DAgent plugins

6.9.4.1. Move to public

The move to public plugin moves incoming messages to a folder in the public store.

Enable the move to public plugin, run the following command:

```
ln -s /usr/share/zarafa-dagent/python/plugins/movetopublic.py /var/lib/zarafa/dagent/plugins/
movetopublic.py
```

For this plugin is a config file required. Make a copy of the configuration file with the following command:

```
cp /usr/share/zarafa-dagent/python/plugins/movetopublic.cfg /etc/zarafa/movetopublic.cfg
```

6.9.4.2. BMP2PNG converter

The BMP2PNG plugin converts a BMP to PNG in the incoming email. This plugin can be used to reduce the image size of the delivered email.

Enable the BMP2PNG plugin, run the following command:

```
ln -s /usr/share/zarafa-dagent/python/plugins/BMP2PNG.py /var/lib/zarafa/dagent/plugins/
BMP2PNG.py
```

Nota

The package **`python-imaging`** is required to use this plugin.

6.9.5. Zarafa-Spooler plugins

6.9.5.1. Disclaimer

The disclaimer plugin add a disclaimer to every email sent with the Zarafa spooler.

The disclaimer plugin supports plain text and HTML emails. RTF emails are not supported. To use the disclaimer plugin, it is necessary to create the directory */etc/zarafa/disclaimers* which must include the disclaimers. The plugin is using the following files for the disclaimer:

Tabela 6.2. Table Disclaimer files

Filename	Description
default.txt	The plain text version of the disclaimer
default.html	The HTML version of the disclaimer
<companyname>.txt	The plain text version of the disclaimer of a company
<companyname>.html	The HTML version of the disclaimer of a company

Importante

All files must encoded in utf8

Enable the disclaimer plugin, run the following command:

```
ln -s /usr/share/zarafa-spooler/python/plugins/disclaimer.py /var/lib/zarafa/spooler/plugins/
disclaimer.py
```

6.9.6. Troubleshooting

How to troubleshoot issues you might have while installing or using the Python plugin framework in the Zarafa dagent and spooler.

6.9.6.1. Log explanation

The Python plugin framework can log a lot of information, so if there are issues, it is recommended to set the **`log_level`** to 6. This will show all the information about the plugin framework.

Python error: No module named mapiplugin

The path to the plugin manager is invalid, this means the plugin framework can not be loaded and will result in the following error:

```
<DATE> [id] PYTHONPATH = /usr/share/zarafa-dagent/python/Unknown_path
<DATE> [id]   Python type: (null)
```

```
<DATE> [id]   Python error: No module named mapiplugin
<DATE> [id] Unable to initialize the dagent plugin manager
```

Check the path in **plugin_manager_path** should refer to the directory with the following files,

- mapiplugin.py
- pluginmanager.py
- plugintemplates.py
- wraplogger.py

Plugins not loaded

The path to the plugins directory is invalid or the permissions on the directory are invalid if this is the case you will receive the following error:

```
<DATE> [id] * Loading plugins started
<DATE> [id] ! Plugins directory '/usr/share/zarafa-dagent/python/plugins/invalid' doesn't
exists. Plugins not loaded.
```

Check the path in **plugin_path** by default it refer to the directory '/var/lib/zarafa/dagent/plugins', the permissions on the directory must atleast have read and execute permissions.

Python error: *PySwigObject* object has no attribute *Log*

There is an invalid version of MAPICore loaded. The old beta python-MAPI package installed the files in another directory but after removing the package the generated files are not removed after you start the dagent or spooler the old generated file is loaded an cause the following error:

```
<DATE> [id] PYTHONPATH = /usr/share/zarafa-dagent/python/
<DATE> [id]   Python type: (null)
<DATE> [id]   Python error: 'PySwigObject' object has no attribute 'Log'
<DATE> [id]   Python trace: /usr/share/zarafa-dagent/python/mapiplugin.py(13) __init__
<DATE> [id]   Python trace: /usr/share/zarafa-dagent/python/pluginmanager.py(16) loadPlugins
<DATE> [id]   Python trace: /usr/share/zarafa-dagent/python/wraplogger.py(16) logInfo
<DATE> [id] Unable to initialize the dagent plugin manager
```

To fix this issue remove the MAPICore.pyc files from your system. One of the locations can be **/usr/lib/python2.6/dist-packages/MAPICore.pyc**

6.9.6.2. Problem - Solution

No plugins are loaded in the zarafa-dagent

Does the plugin exist in the directory **plugin_path** by default in '/var/lib/zarafa/dagent/plugins'? If not, create a symlink to the plugin to activated or just copy the plugin into the directory.

No plugins are loaded in the zarafa-spooler

Does the plugin exist in the directory **plugin_path** by default in '/var/lib/zarafa/spooler/plugins'? If not, create a symlink to the plugin to activated or just copy the plugin into the directory.

6.10. Running ZCP multi-server behind Reverse Proxy

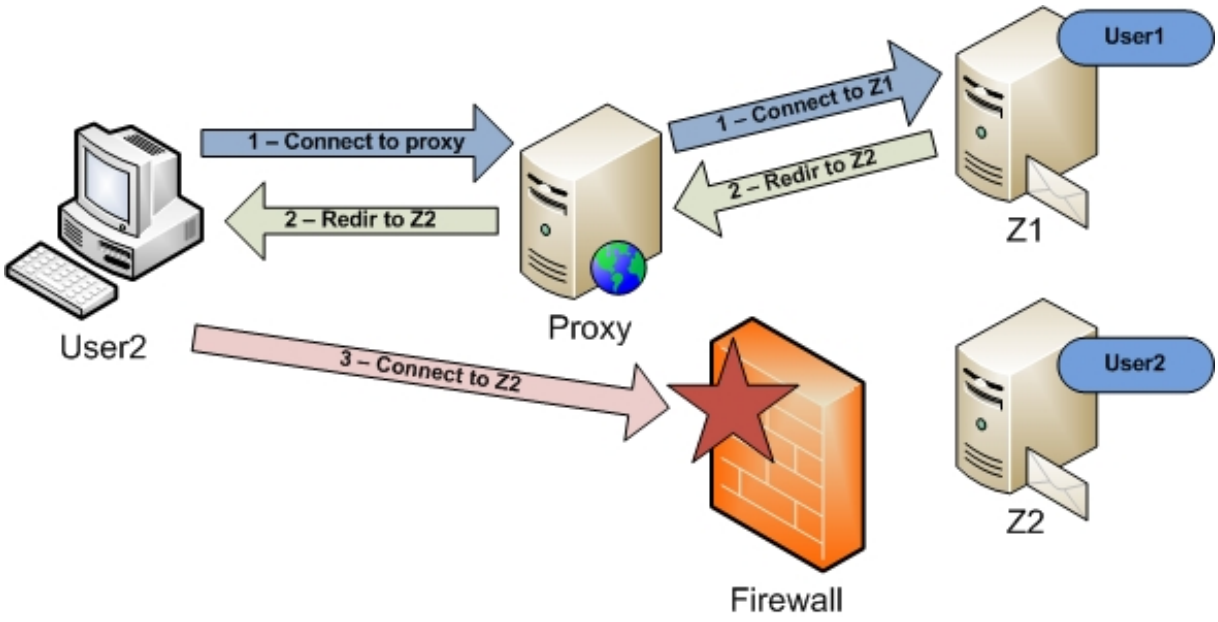
Certain setups require that zarafa-server is not exposed directly to the internet. When offering Outlook access, it is sometimes needed to configure a reverse proxy so that Outlook users can connect to the reverse proxy and not directly to zarafa-server.

Setting up a reverse proxy with a single zarafa-server is quite easy and can be found in chapter 5.1.3 of this administrator manual, however when using a multi-server setup this is a completely different story. Due to the redirection protocol within Zarafa it is quite difficult to setup a reverse proxy for a MutliServer environment, however not impossible.

6.10.1. Description of redirection problem

With redirection the following problem may arise when using a reverse proxy:

1. Outlook connects to a reverse proxy, and the reverse proxy connects to node Z1.
2. Node Z1 will send a redirect for User2 to node Z2.
3. Outlook tries to connect directly to node Z2, but this connection will break on the Firewall.

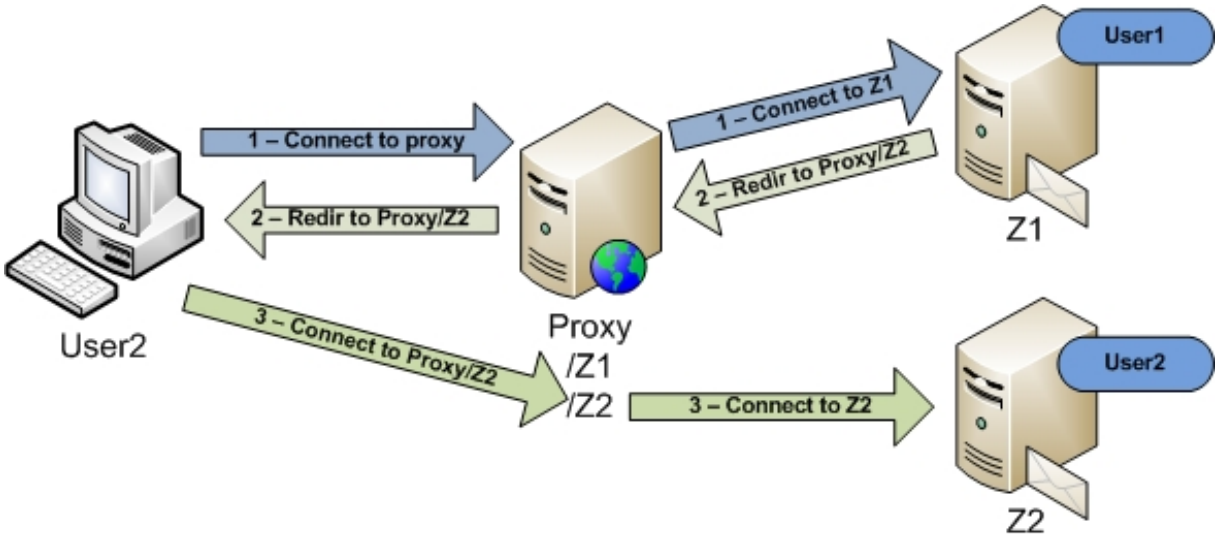


Therefore zarafa-server has some new options since version 7.1, which will make it easier to setup a reverse proxy for a multi-server environment.

In our new setup the reverse proxy will add extra header information, so the zarafa-server will detect that a connection is being made through a reverse proxy. When a connection is made through a reverse proxy (when the extra header is detected) Zarafa will not reply with the normal redirection string, but it will fetch the connection string from a new **`ldap`** attribute: ZARAFAPROXYURL.

Outlook will then still connect to the reverse proxy, even when a redirect command is given:

1. Outlook connects to the reverse proxy, and the reverse proxy adds the extra header and connects to node Z1.
2. Node Z1 detects the extra header and will send a redirect for User2 to node Proxy/Z2.
3. Outlook will now connect again to the proxy, but with a different path /Z2. The proxy will now connect to Z2 so the store of User2 can be opened.



6.10.2. Setup Prerequisites

When setting up a reverse proxy for a multi-server setup using the new ZCP options, the following prerequisites need to be met:

1. ZCP 7.1 or newer.
2. OpenLDAP or ADS with the schema extensions from ZCP 7.1 or newer.
3. A reverse proxy which fully supports HTTP/1.1 (make sure that also the transport encoding "Chuncked Encoding" is supported).

6.10.3. Example Setup with Apache

Apache 2.2 and newer fully supports HTTP/1.1 in the mod_proxy module.

In our example setup we will use an Apache setup which listens on port 237. In your Apache config you will need to add the following:

```
<IfModule mod_ssl.c>
  NameVirtualHost *:237
  Listen 237
</IfModule>
```

We assume that you have created the correct certificates for Apache, so that Outlook is able to connect using SSL.

6.10.3.1. Configuring Apache

In our example setup we will create a virtual host which is used for reverse proxying:

1. /zarafa will be reverse proxied to node Z1 (Default connection is made to /zarafa)
2. /z1 will be reverse proxied to node Z1 (When a redirection is made to node Z1)
3. /z2 will be reverse proxied to node Z2 (When a redirection is made to node Z2)

In our Apache config we will setup this virtual host:

```
<VirtualHost *:237>
```

```
  ServerName zproxy.example.com
  SSLProxyEngine On

  ProxyPass /zarafa https://z1:237/zarafa retry=0
  ProxyPassReverse /zarafa https://z1:237/zarafa retry=0

  ProxyPass /z1 https://z1:237/z1 retry=0
  ProxyPassReverse /z1 https://z1:237/z1 retry=0

  ProxyPass /z2 https://z2:237/z2 retry=0
  ProxyPassReverse /z2 https://z2:237/z2 retry=0

  Header add zarafa_proxy "yes"
  RequestHeader set zarafa_proxy "yes"

  SSLEngine On
  SSLVerifyDepth 2

  SSLCertificateFile /path/to/WEB.CRT
  SSLCertificateKeyFile /path/to/WEB.KEY
  SSLCACertificateFile /path/to/CA.CRT

  CustomLog /var/log/apache2/zproxy.example.com-access.log combined
  ErrorLog /var/log/apache2/zproxy.example.com-error.log
</VirtualHost>
```

Nota

When using Apache as a reverse proxy, it is advised to use Apache in a threaded model and not in a prefork model, as the threaded model is able to handle far more concurrent connections then the prefork model.

6.10.3.2. Adding attribute to Servers

We assume you have installed the ZCP 7.1 or newer schema extensions.

In **ldap** add the attribute ZARAFAPROXYURL to all servers in the multi-server environment.

For node Z1 this will be:

```
ZARAFAPROXYURL: https://zproxy.example.com:237/z1
```

So the complete **ldap** record for node Z1 may look something like this:

```
objectClass: top
objectClass: zarafa-server
objectClass: device
objectClass: ipHost
ZARAFAHTTPPORT: 236
ZARAFASSLPORT: 237
ZARAFAFILEPATH: /var/run/zarafa
ipHostNumber: 192.168.1.1
cn: z1
ZARAFAPROXYURL: https://zproxy.example.com:237/z1
```

For node Z2 this will be:

```
ZARAFAPROXYURL: https://zproxy.example.com:237/z2
```

So the complete **ldap** record for node Z2 may look something like this:

```
objectClass: top
objectClass: zarafa-server
objectClass: device
objectClass: ipHost
ZARAFAHTTPPORT: 236
ZARAFASSLPORT: 237
ZARAFAFILEPATH: /var/run/zarafa
ipHostNumber: 192.168.1.2
cn: z2
ZARAFAPROXYURL: https://zproxy.example.com:237/z2
```

6.10.3.3. Configuring Zarafa Server

Now zarafa-server needs to be configured, so that it will send the correct redirect command when the proxy header is detected.

In this example we configured Apache to add the header "zarafa_proxy", if a connection is being made through our reverse proxy.

On all the zarafa servers in the multi-server environment we will need to add an extra config option to the server.cfg:

```
proxy_header = zarafa_proxy
```

Zarafa-server will now send the ZARAFAPROXYURL as redirect string to the client when the header "zarafa_proxy" is detected.

However, internal (´behind´ the proxy) redirections must **not** be redirected to the proxy since this is not necessary. So any internal service (e.g. BES server) will not connect to the reverse proxy, so the extra header is not added and zarafa-server will send the normal redirect string which is generated from the **ldap** database.

The proxy_header option can have different values:

1. Empty: proxy_header option will not be used.
2. [header]: zarafa-server will check for [header], when found zarafa-server send the ZARAFAPROXYURL as redirect string.
3. *: will force zarafa-server to send the ZARAFAPROXYURL as a redirect string everytime a redirect command is given. With this value set, you do not need to add the extra header in your reverse proxy. However also internal (´behind´ the proxy) services will be redirected to the reverse proxy.

Gerenciando os serviços do Zarafa

7.1. Iniciando os serviços

Existem 7 serviços que podem ser executados:

- **`zarafa-server`**, o servidor de processos
- **`zarafa-spooler`**, enviar um e-mail da caixa de saída para um servidor SMTP
- **`zarafa-monitor`**, verifica os limites de quota
- **`zarafa-gateway`**, proporciona acesso IMAP e POP3
- **`zarafa-ical`**, proporciona acesso ao iCal e CALDAV aos clientes que utilizam esse tipo de calendário
- **`zarafa-licensed`**, item necessário quando se está usando algum módulo fechado do Zarafa no Zarafa-server
- **`zarafa-search`**, provides a full text indexing service for quick searching through email and attachments
- **`zarafa-dagent`**, executa como um serviço quando se está usando protocolo local de transferência de e-mail (LMTP, veja *Seção 5.4, "Integração do ZCP Postfix"*)

O processos **`zarafa-server`** e **`zarafa-spooler`** são obrigatórios para executar o Zarafa. Já o **`zarafa-monitor`**, **`zarafa-gateway`**, e **`zarafa-ical`** São serviços opcionais. Para iniciar um serviço, digite:

```
/etc/init.d/zarafa-<servicename> start
```

Coloque **`<servicename>`** com o serviço que precisa ser iniciado. Para iniciar o **`zarafa-server`**, digite:

```
/etc/init.d/zarafa-server start
```

Este script vai iniciar o servidor. O script **`init.d`** pode iniciar, interromper e reiniciar os serviços. Se o script **`init.d`** não puder ser usado, o servidor precisará ser iniociado manualmente. É possível dizer explicitamene ao Zarafa onde o arquivo de configuração está usado o interruptor **`-c`**.

```
/usr/bin/zarafa-server -c /etc/zarafa/server.cfg
```

O **`zarafa-server`** vai daemonise, então retornará quase que imediatamente. Utilize **`-F`** para iniciar em primenro plano.O interruptor **`-F`** também pode ser usado por programas, como daemontools, que monitoram serviços.

7.1.1. Interrompendo os serviços

Para interromper um serviço, digite:

```
/etc/init.d/zarafa-<servicename> stop
```

A maior parte dos serviços será interropida imediatamente. O **zarafa-spooler** deve levar até 10 segundos para parar. O **zarafa-server** deve levar até 60 segundos para parar.

7.1.2. Recarregando configurações de serviços

Algumas opções podem ser modificadas e recarregados pelo serviço em um ambiente real. As opções que podem ser recarregadas são descritas no pagima de arquivo de configuração do serviço no manual. Exemplo: para o **zarafa-server**, digite o comando abaixo para obter a página do manual de configuração:

```
man zarafa-server.cfg
```

No capítulo de **recarregamento** estão todas as opções que podem ser recarregadas para determinado serviço. Para fazer o serviço recarregar o arquivo de configuração, digite:

```
/etc/init.d/zarafa-<servicename> reload
```

7.2. Opções de acesso

Cada componente permite escolher o método de registro a ser usado em seu arquivo de configuração. Duas formas de métodos de registro são suportados: arquivo e syslog.

Normalmente, todos os componentes de registro do Zarafa e seus respectivos arquivos se encontram em **/var/log/zarafa**. Esse diretório é criado quando os pacotes são instalados. Quando este diretório não estiver presente, ou não é gravável para o usuário que está executando, os serviços não serão capazes de abrir os seus arquivos de registro e irá imprimir as mensagens de registo para a saída padrão.

é possível configurar mensagens de registo do servidor. As seguintes opções precisam ser alterados no arquivo de configuração:

```
log_method
```

Como registrar as mensagens.O **file** envia as mensagens para um arquivo. No sistema Linux, **syslog** envia as mensagens para o e-mail de registro padrão através do syslog.

```
log_file
```

Quando o **log_method** está configurado para **file**, está será a variável que definirá o nome do arquivo. O servidor precisa gravar o acesso ao diretório a ao arquivo.

```
log_level
```

Aumente o nível de mensagens que serão registradas. Nível **6**é o maior nível.

```
log_timestamp
```

1 ou **0**; Este irá habilitar ou desbilitar uma marcação de hora, quando se estiver usando um arquivo como método de registro.

Registro de outros serviços que não o **Zarafa-server** são configurados em uma mesma maneira como o servidor.

7.3. Registro de segurança

Nas versões 7.0 e 6.40.7 do Zarafa, foi acrescentado um recurso de segurança adicional. Com base nesta auditoria o registro pode ser feito no Zarafa-server. Este registro conterá as mensagens de inicialização, autenticações de utilizador e ações de acesso em caixas de delegado.

7.3.1. Itens de registro

7.3.1.1. Inicialização

Quando o servidor é (re)iniciado, a mensagem abaixo aparecerá no registro seguro:

```
zarafa-server startup by user uid=0
```

É possível que apareça a seguinte marca na linha de inicialização:

uid
: O ID do usuário usado para iniciar o servidoe (não necessariamente o usuário no qual o servidor será executado)

7.3.1.2. Sinais

Quando o servidor recebe um sinal, a mensagem abaixo irá surgir no registro de segurança:

```
zarafa-server signalled sig=15
```

É possivel que apareça a seguinte marca na linha de sinal:

sig
: O sinal recebido pelo servidor. Veja **`man 7 signal`** para ter uma lista dos sinais mais comuns.

7.3.1.3. Autentificações

Quando um usuário (não o usuário do SISTEMA interno) se conecta, a mensagem abaixo surgirá no registro de segurança:

Autentificação correta:

```
authenticate ok user='john' from='127.0.0.1' method='User supplied password'
 program='apache2'
```

Autentificação incorreta:

```
authenticate failed user='john' from='127.0.0.1' program='apache2'
```

Somente com acesso sso:

```
authenticate spoofed user='john' requested='test' from='192.168.50.178'
 method='kerberos sso' program='OUTLOOK.EXE'
```

As seguintes etiquetas podem aparecer na linha de autentificação:

usuário
: O nome de usuário enviado para o servidor Zarafa.

solicitado
: O nome no perfil MAPI para abrir a caixa de, a etiqueta usuário será a do usuário real autenticado. (somente SSO)

de
: Socket Unix ou endereço IP da conexão para a qual o servodor foi feito.

método
: Método através do qual o usuário foi validado: socket, certificado, senha, ntlm sso ou kerbero sso.

programa
: O programa que está sendo usado para acessar.

7.3.1.4. Authentications with impersonation

When a user logs in and authenticates as another user, the following message will be printed in the security log:

Correct impersonation:

```
authenticate ok user='john' from='127.0.0.1' method='User supplied password'
 program='apache2'
impersonate ok user='jane', from='127.0.0.1' program='apache2'
 impersonator='john'
```

Incorrect impersonation:

```
authenticate ok user='john' from='127.0.0.1' method='User supplied password'
 program='apache2'
impersonate failed user='jane', from='127.0.0.1' program='apache2'
 impersonator='john'
```

The following tags are possible in the impersonation line:

usuário
: The username of the user being impersonated.

de
: Socket Unix ou endereço IP da conexão para a qual o servodor foi feito.

programa
: O programa que está sendo usado para acessar.

impersonator
: The user that is impersonating another user. This is the user whose credentials are being checked.

7.3.1.5. Ações de compartilhamento

When a user opens objects that are not within his own store, a message will be logged. This also accounts fo the **`Public store`**.

A mensagem abaixo surgirá no regístro de segurança:

Ações de compartilhamento permitidas:

```
access allowed objectid=387538 type=3 ownername='test' username='constant' rights='view'
```

Ação de compartilhamento negada

```
access denied objectid=387538 type=3 ownername='test' username='constant' rights='view'
```

É possível que as etiquetas abaixo apareçam na linha de compartilhamento

objectid
: O objeto que está sendo executado.

tipo
: The MAPI type of the object. Possible values are 3 (store), 5 (folder) and 7 (message).

Nome do Proprietário
: O proprietário da conta onde o objeto está armazenado (não necessáriamente o usuário que criou esse objeto).

Nome de usuário
: O usuário que executa a ação no objeto.

direitos
: A ação que está sendo executada.

Nota

Para a **`Pasta Pública`** nome de proprietário será SYSTEM no modo single-tenancy, e o nome da empresa no modo muti-tenancy.

Ações possíveis em matéria de direitos:

ler
: Leitura do objeto.

Criar
: Criar um novo objeto

Editar
: Editar um objeto já esxistente (por exemplo, alterar propriedades; adicionar/remover destinatários e anexos)

Deletar
: Deletar (sodtdelets) ou remover o objeto

Criar pasta
: Criar uma nova pasta

Exibição
: Leitura da hierarquia de pastas / conteúdos tabelas

Permissões de pastas
: Alternando permissões, modificando e excluindo pastas

Proprietário
: Enviar/concluir/suspender envio de uma mensagem, nunca são permitidos na conta de outra pessoa, salvo se você for o proprietário.

Adiministrador
: Não utilizado, nunca será realmente impresso

7.3.1.6. Análise de regístro

Quando um usuário está acessando uma conta ou pasta delegada, um registro é feito para o audit.log. É possível ter uma visão panorãmica das pastas delegadas que foram acessadas analisando o audit.log.

O comando abaixo vai analisar o arquivo de registro e tornar a saída mais fácil de ser utilizada:

```
perl /usr/share/doc/zarafa/audit-parse.pl < /var/log/zarafa/audit.log
```

O script mostrará p nome exato da pasta que foi acessada na conta delegada:

```
access allowed rights='view' type='folder' objectid='store\27\IPM_SUBTREE\Calendar'
 username='john' ownername='mary'
```

Neste exemplo o usuário John havia aberto o calendário do usuário Mary.

7.3.1.7. Não registrado

Somente os direitos dos objetos de nivel superior são verificados, então você não verá as ações realizadas em anexos, destinatários ou objetos msg-in-msg.

7.3.2. Configuração

No **`/etc/zarafa/server.cfg`**são adicionadas as opções abaixo:

```
audit_log_enabled = no
audit_log_method = syslog
audit_log_file = -
audit_log_level = 1
audit_log_timestamp = 0
```

Por padrão, o auditor de registro está desabilitado. Quando ativado, o padrão é registrar através de syslog, pois ele pode ser configurado para enviar as mensagens para um servidor syslog externo. A facilidade do syslog **`authpriv`** será usada para enviar mensagens.

7.4. Monitoramento estatístico do Zarafa

As estatísticas e o status do servidor podem ser verificados com a ferramenta Zarafa-stat que oferece as seguintes opções:

- **`--system`** Fornece informações sobre tópicos, SQL e caches
- **`--session`** Fornece informações sobre as seções e tempo gasto no servidor com ligações SOAP
- **`--users`** Fornece informações sobre usuários, armazenamento e quotas
- **`--company`** Fornece informações sobre filiais, tamanho das filiais e quota
- **`--top`** Apresenta uma lista das pricipais informações informações sobre sessões e recursos de servidor usados

Para usar a ferramenta Zarafa-stat utilize o comando a seguir:

```
zarafa-stats --top
Last update: Tue Mar 29 13:40:18 2011
Sess: 1     Sess grp: 1       Users: 1    Hosts: 1    CPU: 0%   QLen:    QAge:
SQL/s SEL:    0 UPD:   0 INS:   0 DEL:   0        Threads(idle): ()
        SOAP calls: 6

VERSION      USERID     IP/PID          APP            TIME    CPUTIME CPU    NREQ   TASK
7,0,0,24874  SYSTEM     4527            zarafa-spooler 0:00    0:00    0      6
tableQueryRows
```

O resumo **`--top`** fornece a cada segundo, informações sobre o uso do CPU, número de clientes conectados, tópicos abertos, tamanho de fila e consultas SQL. Quando o servidor tem um comprimento da fila grande, a quantidade de tópicos deve aumentar normalmente.

7.5. Sistema de exclusão

Se um usuário deleta e-mails, itens de calendário ou pastas inteiras, por padrão, o conteúdo tais objetos são movidos para a pasta de **`Itens excluídos`**.

Quando os itens são removidos da pasta **`Itens Excluídos`**, tais itens ainda não foram realmente deletados do banco de dados. Eles são marcados como excluídos, de modo que o usuário não mais os vê. Mesmo que o usuário exclua os itens com o comando <SHIFT> <delete>, esses itens não são removidos do banco de dados, mas sim marcados como deletados.

Isso permite que a restauração de itens seja rápida e fácil no Outlook: escolha*Extra* na barra de menu do Outlook, e clique em *Restaurar itens excluídos*. Os itens são agrupados por pasta das quais foram excluídos. A maior parte do itens aparecerão na pasta *Itens excluídos*, já que é de lá que eles foram excluídos.

Esse tipo de exclusão sempre permanece no banco de dados, até que sejam removidos. Quando um item será removido é definido por configurações de valores do **`softdelete_lifetime`**. O valor padrão é de **`30`** (dias).

Neste exemplo o valor está definido como **`30`**. Isso significa que os itens que foram deletados permanecerão no banco de dados por **`30`** depois da data em que eles foram excuídos. Quando essa opção estiver configurada para **`0`** (zero), os itens nunca serão removidos do banco de dados.

A limpeza também pode ser acionada com o seguinte comando:

```
zarafa-admin --purge-softdelete <days>
```

`<days>` denota o número de dias que os itens removidos recentemente são mantidos. Quando for **`0`** (zero) todos os itens removidos são excluídos.

Por motivos de desempenho uma limpeza manual do sistema softdelete é aconselhável para ambientes maiores no Zarafa. Isto pode ser configurado simplesmente por um cron job.

Gerenciamento dos Usuários

8.1. Pasta Pública

Depois que o servidor iniciou corretamente, as contas podem ser criadas. Existem dois tipos de contas: privadas e públicas. Só pode haver uma conta pública. Ela pode ser criada com o seguinte comando:

```
/usr/bin/zarafa-admin -s
```

A armazenagem pública é a pasta que todos os usuários sempre podem abrir. Após a instalar e configurar o servidor, uma armazenagem pública precisa ser criada antes que armazenagens particulares possam ser feitas. Se o ZCP for configurado para multi-locação, uma armazenagem pública será criada automaticamente para cada companhia.

Ao utilizar o suporte de multi-servidor, a armazenagem pública somente pode ser criada no nódulo multi-servidor que contém o atributo ZarafaContainsPublic ativado. Atualmente a Armazenagem Pública pode ser criada somente em um servidor. Para mais informações, ver *Seção 6.3.2, "Preparando / definindo o servidor LDAP para configuração multi-servidor"*.

Nota

O armazenamento público é, por padrão, acessível e gravável para todos os usuários. Favor rever as permissões antes de iniciar o uso do sistema Zarafa.

8.2. Uso geral da ferramenta Zarafa-admin

O ZCP oferece a ferramenta administrativa **`zarafa-admin`** para o gerenciamento de usuários e grupos. Ao utilizar o plugin **`DB`**, a ferramenta pode ser utilizada para criar ou deletar usuários e grupos. Ao utilizar o plugin **`unix`** ou **`ldap`** a ferramente não pode ser utilizada para criação de usuários e grupos, mas pode ainda ser utilizada para conseguir mais informações obre usuários e grupos.

Todos os usuários ou grupos disponíveis podem ser mostrados utilizando-se os seguintes comandos:

```
zarafa-admin -l
zarafa-admin -L
```

Mas mostrar mais informações sobre um usuário específico, utilize:

```
zarafa-admin --details john
Username: john
Fullname: John Doe
Emailaddress: j.doe@example.com
Active: yes
Administrator: no
Address book: Visible
Last logon: 03/25/11 19:50:29
Last logoff: 03/25/11 19:50:29
Quota overrides: no
Warning level: 1024 MB
Soft level: 2048 MB
Hard level: 3072 MB
Current store size: 462 MB
```

```
Groups (1):
Everyone
Sales team
```

Mas mostrar mais informações sobre um grupo específico, utilize:

```
zarafa-admin --details sales --type group
Groupname: sales
Fullname: sales
Emailaddress:
Address book: Visible
Users (1):
Username Fullname Homeserver
---------------------------------------------
john John Doe
mary Mary Jones
```

Quando um usuário for deletado, a caixa de mensagens do usuário ainda será mantida no banco de dados. Utilize o seguinte comando para recuperar uma lista de armazenagens sem um usuário, e usuários sem uma armazenagem:

```
/usr/bin/zarafa-admin --list-orphans
Stores without users:
       Store guid                               Guessed username        Last modified
Store size

-------------------------------------------------------------------------------------------------
       CAC27E6D70BB45B0B712B760AE6BA0A8         steve                   2010/03/22 14:22
2334KB

Users without stores:
       Username
       ----------------------------
       jane
```

Pode ser decidido remover a armazenagem do banco de dados ou ligar a armazenagem para outro usuário para que seja possível acessá-la novamente. Para remover a armazenagem do banco de dados, a qual é uma ação irreversível, utilize o seguinte comando:

```
/usr/bin/zarafa-admin --remove-store <store-guid>
```

Para ligar a armazenagem para outro usuário, utilize o seguinte comando:

```
/usr/bin/zarafa-admin --hook-store <store-guid> -u <user>
```

O usuário ao qual for dada a opção **`-u`** terá agora uma nova armazenagem ligada a ele. Faça um novo login com o webaccess ou crie um novo perfil no Outlook para acessar a armazenagem.

Importante

Quando uma armazenagem é conectada a um usuário que já tem uma armazenagem própria, a armazenagem original será desfeita. Esta armazenagem original pode ser encontrada utilizando-se as opções de **`list-orphans`** do comando **`zarafa-admin`**.

Nota

No Zarafa 6.30.6 e versões anteriores, a armazenagem do usuário era movida para a pasta "Armazenagens deletadas" na armazenagem pública após um usuário ser deletado. Esta pasta somente é disponível para usuários administrativos. Os administradores podem visualizar as pastas ou deletar as armazenagens completamente removendo a pasta correspondente da pasta "Armazenagens deletadas". Isto é relevante para todos os plugins de usuário.

Mais informações sobre todas as opções do **zarafa-admin** podem ser encontradas no manual.

```
man zarafa-admin
```

8.3. Gerenciamento de usuários com o plugin DB

Por padrão, o plugin DB será utilizado como o plugin de gerenciamento de usuário. Abaixo será descrito como gerenciar usuários com o comando **zarafa-admin**. Para o gerenciamento de usuário com o plugin de usuário LDAP, favor ver *Seção 8.5, "Gestão do Usuário com o LDAP ou Active Directory"*.

No momento o ZCP não tem uma interface de gerenciamento de usuário baseada na web ou em gráficos, contudo há diferentes produtos de terceiros que fornecem gerenciamento do sistema Zarafa baseado na web.

8.3.1. Criando usuários com o plugin DB

Para criar um novo usuário, utilize o seguinte comando:

```
/usr/bin/zarafa-admin -c <user name> -p <password> \
-e <email> -f <full name> -a <administrator>
```

Os campos entre <> devem ser preenchidos como mostrado abaixo:

- **Nome do usuário:** O nome do usuário. Com este nome o usuário irá acessar a armazenagem.
- **Senha:** A senha em texto simples. A senha será armazenada encriptada no banco de dados.
- **Email:** O email do usuário. Geralmente é do tipo **<user name>@<email domain>**.
- **Nome completo:** O nome completo do usuário. Como o nome completo terá espaço como caracter, e talvez outro caracter que não seja alfa-numérico, o nome deve ser escrito entre aspas (**''**).
- **Administrator:** This value should be **0** or **1**. When a user is administrator, the user will be allowed to open all Zarafa stores of any user. It is also possible to pass **2** as administrator level, this will make the user a system administrator who can access mailboxes within other companies.

Todos os campos exceto o endereço de email são sensíveis ao uso de caixa alta.

A senha também pode ser definida utilizado a chave **-P**. A senha então não é dada na entrada do comando, mas é pedida pela ferramenta **zarafa-admin**. A senha não é ecoada na tela e precisa ser digitada duas vezes para verificação.

8.3.2. Usuários não-ativos

Um usuário não-ativo não pode se logar ao ZCP, mas emails podem ser entregues a este usuário e a armazenagem pode ser aberta por usuários com as permissões corretas. Os usuários não-ativos podem ser especialmente utilizados para caixas de mensagens funcionais, recursos e salas.

Para criar um usuário não-ativo, utilize o seguinte comando:

```
zarafa-admin -c <user name> -P -e <email> -f <full name> -n 1
```

Nota

No ZCP versão 6.30 e anteriores, não é possível trocar um usuário ativo com um não-ativo e vice versa. Trocar o valor de não-ativo acarretará na deleção da caixa de mensagem.

8.3.3. Atualizando informação do usuário com o plugin DB

A mesma ferramenta **`zarafa-admin`** pode ser utilizada para atualizar as armazenagens e informações do usuário. Utilize o seguinte comando para atualizar:

```
/usr/bin/zarafa-admin -u <user name> [-U <new user name>] \
[-p <new password>] [-e <email>] \
[-f <full name>] [-a <0|1>]
```

Todas as mudanças não opcionais. Por exemplo, somente uma senha para um usuário existente pode ser atualizada, deixando as outras informações do usuário como eram.

8.3.4. Deletando usuários com o plugin DB

Para deletar um usuário do servidor, utilize o seguinte comando:

```
/usr/bin/zarafa-admin -d <user name>
```

O usuário será deletado do banco de dados. Contudo, a armazenagem será mantida no banco de dados, mas não é acessível. Ver *Seção 8.2, “Uso geral da ferramenta Zarafa-admin”* para mais informações sobre o manejo desse tipo de armazenagem.

8.3.5. Configurando permissões de "Send as"

O ZCP suporta dois tipos de delegação de envio:

Permissões de Send on Behalf

Se um usuário cede a permissão apropriada para outro usuário, este último pode enviar itens em nome do outro usuário. Neste caso, um email ou requisição de reunião será enviado com o seguinte campo "De": **`<delegate>`** on behalf of **`<user>`**. Este item somente pode ser definido à partir do WebAccess ou cliente Outlook.

Permissões de Send As

Se o administrador do sistema der os direitos ao usuário B para "enviar como" o usuário A, o recebedor de um email não verá que o usuário B o enviou. O recebedor somente verá o endereço de email do usuário A no campo "de".

Configurar a delegação de sendas com **zarafa-admin** somente é aplicável com o plugin DB ou INIX. Para configurar o LDAP ou o Active Directory, verifique *Seção 8.5, “Gestão do Usuário com o LDAP ou Active Directory”*.

Adicione um usuário à lista do delegado que está sendo atualizado como um usuário de "enviar como". O delegado pode agora enviar mensagens como o nome do usuário atualizado, a menso que o usuário atualizado defina o delegado como um delegado baseado no usuário. Esta opção somente é válida com a ação de atualização **-u**.

```
zarafa-admin -u <delegate> --add-sendas <user>
```

Por exemplo:

```
zarafa-admin -u helpdesk --add-sendas john
```

Remova um usuário da lista do delegado que está sendo atualizado como um usuário de "enviar como". Esta opção somente é válida como a ação de atualização **-u**.

```
zarafa-admin -u <delegate> --del-sendas <user>
```

Liste todos os usuários que estão na lista do delegado.

```
zarafa-admin --list-sendas helpdesk
Send-as list (1) for user helpdesk:
Username Fullname
----------------------------
john John Doe
```

Nota

Com o plugin DB, as permissões sendas não poder ser configuradas em grupos.

Nota

Quando as permissões "em nome de" e "sendas" forem configuradas ao mesmo tempo no mesmo usuários, o email será sempre enviado com "em nome de".

8.3.6. Grupos

O servidor suporta grupos. Usuários podem pertencer a qualquer número de grupos. Todo usuários sempre pertence ao grupo especial Todos. A definição de configurações de segurança em pastas e itens são as mesmas tanto para usuários quanto para grupos.

Por exemplo, o grupo Todos tem acesso de leitura para a Caixa de Entrada de Peter. Neste ponto, todo usuário pode ler o email na caixa de entrada de Peter, pois todos os usuários são membros do grupo Todos.

Quando um novo usuário Zarafa é criado, somente a informação free/busy está aberta para acesso a leitura para o grupo Todos, por padrão.

8.3.6.1. Criando grupos com o plugin DB

Utilizando-se a ferramenta **zarafa-admin**, grupos podem ser criados e usuários podem ser adicionados ou removidos dos grupos. No exemplo seguinte, um usuário john é criado, uma administração de grupo é criada e o usuário john é adicionado à administração do grupo.

```
zarafa-admin -c john -p secret -f “John Doe” -e “j.doe@domain.com”
zarafa-admin -g administration
zarafa-admin -b john -i administration
```

Utilizando-se as opções **-l** ou **-L**, uma lista de usuários ou grupos pode ser listada à partir do servidor.

Todos os usuários criados serão membros do grupo "Todos, e isto não pode ser alterado. Grupos criados com o plugin DB podem ser utilizados tanto para configurar permissões quanto para enviar emails para um grupo específico.

8.4. Gerenciamento de usuários com o plugin UNIX

Ao se integrar o ZCP com os usuários e grupos padrão do servidor Linux, parte da administração do usuário precisa ser feita via ferramentas de gerenciamento de usuários padrão do Linux, como a ferramenta **useradd** e a administração de usuário específico do Zarafa precisa ser feita com a ferramenta **zarafa-admin**.

8.4.1. Criando usuários com o plugin Unix

Para criar um novo usuário, utilize o comando padrão **adduser**.

```
useradd <username> -c "Full name"
passwd <username>
```

Como o endereço de email do usuário não poder ser especificado no comando **adduser**, o endereço padrão de email será <username>@default_domain. O domínio padrão está especificado no arquivo **/etc/zarafa/unix.cfg**.

Este endereço de email pode ser mudado utilizando-se a ferramenta **zarafa-admin**.

```
zarafa-admin -u <username> -e <email address>
```

8.4.2. Usuários não-ativos

Um usuário não-ativo não pode se logar ao ZCP, mas emails podem ser entregues a este usuário e a armazenagem pode ser aberta por usuários com as permissões corretas. Os usuários não-ativos podem ser especialmente utilizados para caixas de mensagens funcionais, recursos e salas.

Para criar um usuário não-ativo com o plugin Unix, tenha certeza que a estrutura de login do usuário esteja definida para **/bin/false**. A estrutura de login para usuário não-ativos pode ser também configurada no **/etc/zarafa/unix.cfg**.

Nota

No ZCP versão 6.30 e anteriores, não é possível trocar um usuário ativo com um não-ativo e vice versa. Trocar o valor de não-ativo acarretará na deleção da caixa de mensagem.

8.4.3. Atualizando informação do usuário com o plugin Unix

Alterar a informação do usuário ao se utilizar o plugin Unix pode ser feito para se obter algumas informações com as ferramentas padrão de gerenciamento de usuário do Linux e para outras informações com a ferramenta **`zarafa-admin`**.

As seguintes informações podem ser alteradas no arquivo **`/etc/passwd`** ou com as ferramentas padrão de gerenciamento de usuário do Linux:

- Nome do usuário
- Senha
- Nome completo
- Tipo de Caixa de Entrada (ativa ou não-ativa)
- Participação em um grupo

A outra informação seguinte precisa ser mudada e configurada com a ferramenta **`zarafa-admin`**.

- Endereço de email
- Bandeira do Administrador
- Quota
- Permissões de Sendas

8.4.4. Deletando usuários com o plugin Unix

Para deletar um usuário de um servidor, utilize o seguinte comando do Linux:

```
userdel <username>
```

O usuário será deletado do banco de dados. Contudo, a armazenagem será mantida no banco de dados, mas não é acessível. Ver *Seção 8.2, "Uso geral da ferramenta Zarafa-admin"* para mais informações sobre o manejo desse tipo de armazenagem.

8.4.5. Configurando permissões de "Send as"

O ZCP suporta dois tipos de delegação de envio:

Permissões de Send on Behalf

Se um usuário cede a permissão apropriada para outro usuário, este último pode enviar itens em nome do outro usuário. Neste caso, um email ou requisição de reunião será enviado com o seguinte campo "De": **`<delegate>`** on behalf of **`<user>`**. Este item somente pode ser definido à partir do WebAccess ou cliente Outlook.

Permissões de Send As

Se o administrador do sistema der os direitos ao usuário B para "enviar como" o usuário A, o recebedor de um email não verá que o usuário B o enviou. O recebedor somente verá o endereço de email do usuário A no campo "de".

Adicione um usuário à lista do delegado que está sendo atualizado como um usuário de "enviar como". O delegado pode agora enviar mensagens como o nome do usuário atualizado, a menso que

o usuário atualizado defina o delegado como um delegado baseado no usuário. Esta opção somente é válida com a ação de atualização **-u**.

```
zarafa-admin -u <delegate> --add-sendas <user>
```

Por exemplo:

```
zarafa-admin -u helpdesk --add-sendas john
```

Remova um usuário da lista do delegado que está sendo atualizado como um usuário de "enviar como". Esta opção somente é válida como a ação de atualização **-u**.

```
zarafa-admin -u <delegate> --del-sendas <user>
```

Liste todos os usuários que estão na lista do delegado.

```
zarafa-admin --list-sendas helpdesk
Send-as list (1) for user helpdesk:
Username Fullname
----------------------------
john John Doe
```

Nota

Com o plugin Unix, as permissões de sendas não podem ser configuradas em grupos.

Nota

Quando as permissões "em nome de" e "sendas" forem configuradas ao mesmo tempo no mesmo usuários, o email será sempre enviado com "em nome de".

8.4.6. Grupos com o plugin Unix

O servidor suporta grupos. Usuários podem pertencer a qualquer número de grupos. Todo usuários sempre pertence ao grupo especial Todos. A definição de configurações de segurança em pastas e itens são as mesmas tanto para usuários quanto para grupos.

Por exemplo, o grupo Todos tem acesso de leitura para a Caixa de Entrada de Peter. Neste ponto, todo usuário pode ler o email na caixa de entrada de Peter, pois todos os usuários são membros do grupo Todos.

Quando um novo usuário Zarafa é criado, somente a informação free/busy está aberta para acesso a leitura para o grupo Todos, por padrão.

8.4.6.1. Criando grupos com o plugin Unix

Grupos podem ser criados e usuários adicionados ou removidos dos grupos pelas ferramentas de gestão de usuário padrão do Linux. No exemplo seguinte, a administração do grupo é criada e o usuário john é adicionado à administração do grupo.

```
groupadd administration
```

```
usermod -a -G administration john
```

Utilizando-se as opções **`-l`** ou **`-L`**, uma lista de usuários ou grupos pode ser listada à partir do servidor.

Todos os usuários criados serão membros do grupo "Todos" e isto não pode ser mudado. Grupos criados com o plugin Unix podem ser utilizados tanto para configurar permissões quanto para enviar emails para um grupo específico.

8.5. Gestão do Usuário com o LDAP ou Active Directory

O servidor Zarafa utiliza um sistema no qual o administrador de um servidor pode especificar um servidor baseado em LDAP para obter as informações de usuário, grupo e companhia. Isto significa que a gestão do usuário pode ser simplificada para instalações e ferramentas administrativas de LDAP padrão podem ser utilizadas para a gestão do usuário. Além disso, ao utilizar um servidor LDAP torna-se possível integrar o Zarafa em um ambiente existente.

Vários sistemas de servidores LDAP são suportados, e o Zarafa se comunicará com qualquer protoloco de servidor LDAP padrão de versão 3 ou superior. Isto significa que o Zarafa trabalha em combinação com soluções de padrão industrial como o Microsoft Active Directory, OpenLDAP e o eDirectory.

Este capítulo descreve vagamente como o Zarafa utiliza o servidor LDAP como uma fonte para informações do usuário, grupo, contato e companhia. Na maioria dos casos, a definição particular utilizada irá requerer outras opções e definições ao invés daquelas descritas neste documento. Dessa forma, assume-se que o leitor tenha um bom entendimento de como as raízes LDAP funcionam e como elas são configuradas em sua rede.

Para mais informações, favor ver os exemplos de configurações e páginas do manual disponíveis em todos os sistemas nos quais o Zarafa for instalado.

8.5.1. O princípio da sincronização do usuário do Zarafa

Em qualquer servidor do Zarafa, há um banco de dados que contém os dados necessários enquanto se roda o Zarafa. Separado dos dados da pasta e item atual, o banco de dados também mantém informação sobre os direitos de acesso a dados, definições de usuários e meta-dados do usuário para usuários e grupos. Muito desses dados se refere a uma ID de um usuário específico. Por exemplo, um ACL (Access Control List - Lista de Controle de Acesso) para a "caixa de entrada" para o usuário A será armazenado no banco de dados como uma gravação na tabela de ACL. Esta gravação detém os reais direitos de acesso para os objetos e a ID de usuário para o qual a entrada de controle de acesso foi designada.

A ID de usuário citada acima é então uma referência para uma ID de usuário dentro do banco de dados do Zarafa. Esta ID é armazenada na tabela de "usuários", junto com uma referência à ID do usuário no banco de dados de usuário externo (neste caso, um servidor LDAP). por exemplo, o usuário A pode ter a ID de usuário **5** no sistema Zarafa e pode se referir ao item (**`dn=cn=user,dc=example,dc=com`**) no servidor LDAP.

Manter uma lista de usuários desta maneira também resolve o problema da criação de armazenagem para um usuário. Não há forma de provocar um evento de criação de armazenagem no zervidor Zarafa sempre que um usuário é adicionado no servidor LDAP. A tabela de "usuários" fornece uma maneira conveniente para rastrear quais usuários são novos no sistema e assim requerer uma nova armazenagem. O mesmo vale para a remoçao de usuários, já que a armazenagem de usuário precisa ser removida quando o usuário é removido.

Assim, a tebela de "usuários" no Zarafa é quase exclusivamente um mapeamente entre a ID de usuário, a qual é utilizada internamente no Zarafa, e uma referência externa para um usuário no banco de dados LDAP. Naturalmente, quando quaisquer novos usuários são adicionados ou usuários são removidos do servidor LDAP, esta tabela precisa ser mantida em sincronia com as mudanças.

Há várias maneiras de manter a tabela de "usuáriuos" sincronizada com o servidor LDAP, mas o Zarafa escolheu por padrão uma abordagem "no ato". Isto significa que a qualquer hora que um usuário é requisitado do sistema, primeiramente é verificado no servidor LDAP para sua existência, e então é verificado na tabela de "usuários" por sua existência. Se o usuário não existir localmente no servidor Zarafa, então o usuário é criado às pressas, antes que a informação seja retornada para o chamador.

Isto significa que para usuários e administradores, a sincronização pareça ser em tempo real; nunca haverá um atraso entre a adição ou remoção de usuários do servidor LDAP e os usuários que aparecem no Zarafa.

Como todos os componentes do Zarafa usam a mesma interface MAPI para se conectar ao servidor de apoio, não pode surgir problema com qualquer uma das ferramentas do Zarafa onde o banco de dados do usuário esteja fora de sincronização. Por exemplo, enviar um email para um usuário que acabou de ser criado nunca falhará devido ao usuário não existir na tabela de usuários do Zarafa.

Para otimizar esta sincronização com Catálogos globais de Endereços muito grandes no LDAP, há uma definição opcional *sync_gab_realtime* no arquivo de configuração **`server.cfg`**. Quando esta opção é definida para *no*, não há uma sincronização em tempo real entre o diretório do LDAP e o servidor Zarafa. Neste caso, todas as entradas do Catálogo Global de Endereços serão recuperadas à partir do cache do servidor Zarafa. Isto é especialmente útil para definições que têm grandes catálogos de endereços (mais do que 10000 entradas no catálogo de endereços).

A sincronização entre o LDAP e o servidor Zarafa precisa ser forçada com o seguinte comando:

```
zarafa-admin --sync
```

Este comando pode ser executado diariamente ou a cada hora à partir de um cronjob.

8.5.1.1. Adicionar/Remover eventos

O mecanismo acima cria uma situação na qual há seis eventos que podem ser sinalizados:

- Criação de usuário
- Criação de grupo
- Criação de Companhia
- Remoção de usuário
- Remoção de grupo
- Remoção de companhia

Estes seis eventos podem ser relacionados a um script (o qual será descrito posteriormente) para que os administradores do sistema possam realizar ações específicas em seus servidores com estes eventos. Por padrão, o Zarafa somente executará as ações absolutamente necessárias durantes estes eventos; por exemplo, criação e remoção de armazenagem. Quaisquer outros eventos podem ser feitos com script pelo administrador do sistema. Isto significa que, por padrão, nenhuma ação é executada durante a criação de grupo e remoção de grupo.

8.5.1.2. Participação em um grupo

O Zarafa sincroniza usuários, grupos e companhias de forma que possa designar ID's de usuários para eles, mas a participação em um grupo para usuários nunca é armazenada no servidor Zarafa. Isto significa que mudanças na participação em um grupo são também em tempo real e o servidor Zarafa irá examinar a participação no grupo para um usuário ou lista de usuário para um grupo diretamente do servidor LDAP. A maneira como o mapeamento entre membros do grupo e usuários é feita será discutida posteriormente.

8.5.1.3. Dependência do servidor LDAP

Devido ao fato que o banco de dados de "usuários" do Zarafa não manter, de fato, as informações do usuário ou grupo, mas somente uma referência para o servidor LDAP, o servidor Zarafa não pode funcionar sem um servidor LDAP acessível e rodando. Se o servidor LDAP cair enquanto o Zarafa está sendo rodado, as ferramentas do Zarafa não serão capazes de fazer ação alguma, já que todas as ações do lado do servidor requerem algum tipo de interação com o servidor LDAP. Por exemplo, simplesmente abrir um email requer um exame para o servidor LDAP para os grupos nos quais o usuário atual foi inserido. Somente após recuperar esta informação o Zarafa pode determinar se o usuário atual tem os direitos de acesso para abrir a mensagem.

Ao utilizar o OpenLDAP como uma fonte LDAP, é recomendável utilizar a replicação de LDAP para garantir que um servidor LDAP esteja disponível a todo tempo, rodando um servidor OpenLDAP na mesma máquina que o Zarafa. Isto certificará que o servidor LDAP local será sempre alcançável e o Zarafa sempre rodará normalmente.

8.5.1.4. Definindo o repositório de LDAP

Embora, em princípio, quase qualquer repositório LDAP possa ser utilizado com o Zarafa, este capítulo descreve como o Zarafa solicita os dados do servidor e como os dados são utilizados pelo servidor Zarafa e pelas ferramentas.

A seguinte informação é solicitada pelo servidor LDAP:

- Detalhes do usuário (nome, endereço de email, etc)
- Contatos (nome, endereço de email)
- Detalhes do grupo (nome do grupo)
- Detalhes da companhia
- Relacionamentos de Usuário/Grupo (participação no grupo)
- Membros da companhia (participação de usuários e grupos)
- Relacionamentos de companhias (vista entre companhias e permissões de administrador)

Os objetos que são classificados como usuários, contatos, grupos, grupos dinâmicos, listas de endereços ou companhias e os atributos que coontém os dados podem ser configurados através dos arquivos de configuração do Zarafa, de forma que o Zarafa possa se adequar às necessidades do esquema LDAP. Contudo, aqui vão algumas dicas para manter a reposição de LDAP limpa e de fácil manuseio:

- Sempre utilize algum tipo de interface de usuário gráfica para o controle do usuário e grupo. Há muitas ferramentas de configuração de LDAP. (Por exemplo, *phpLDAPadmin*[1] para OpenLDAP como uma interface baseada na web)

- Se houver usuários que utilizarão o Zarafa enquanto outros usuários não utilizarão, tente agrupar estes usuários em "pastas" diferentes. Uma gravação **OU** ou qualquer outro objeto **dc-type** pode ser utilizado para criar estas pastas.

- Se o Active Directory da Microsoft for rodado, tenha certeza que os usuários reais estejam em pastas de LDAP separadas para que o Zarafa não precise importar os usuários padrões como "Administrador" e "Convidado" para o banco de dados. Também é possível filtrar os usuários utilizando um questionamento de busca LDAP, mas estes questionamentos de busca podem ser tornar insatisfatoriamente grandes ao se utilizar o ADS.

Como uma regra geral, utilize sempre o protocolo LDAPS (SSL) quando contatar o servidor de LDAP. Quando o SSL não é utilizado, a informação será transmitida como texto limpo pelo fio. Isto abre possibilidades de se descobrir as senhras de usuários (e administradores!) pela conexão de rede. O Zarafa suporta conexão através de LDAP via SSL e um certificado especificado em **/etc/ldap/ldap.conf**, o qual é compatível tanto com o Active Directory da Microsoft quanto com servidores OpenLDAP. O Zarafa ainda não suporta encriptação do tipo **STARTTLS**. Mais informações sobre configuração do Active Directory com suporte SSL podem ser encontradas em *http://wiki.zarafa.com*.

8.5.2. Manuseio de usuário à partir do ADS

8.5.2.1. Criando grupos utilizando o ADS

Novos usuários podem ser criado utilizando-se a ferramenta padrão de criação de usuário do Active Directory. Ao se criar o usuários, certifique-se que o endereço de email padrão do usuário seja sempre exclusivo.

Para configurar informações específicas do Zarafa para o usuário, selecione a aba **Zarafa** do usuário no Active Directory.

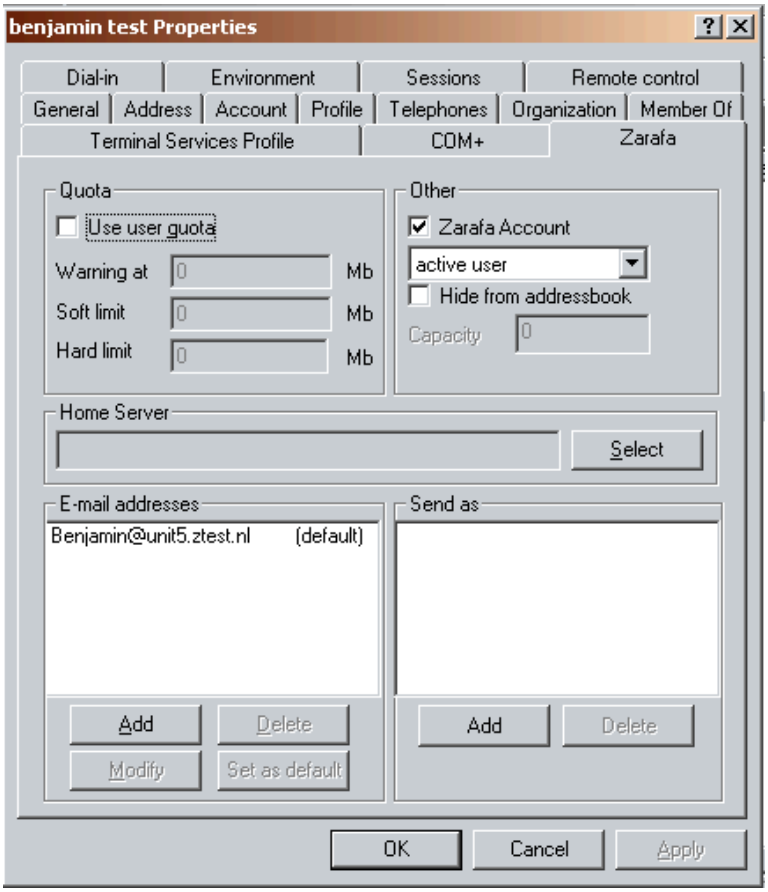


Figura 8.1. Aba de usuário do Zarafa

8.5.2.2. Criando grupos utilizando o ADS

No Active Directory, tanto grupos de segurança quanto de distribuição podem ser criados. Os grupos de segurança podem ser usados para definir permissões e enviar emails. Os grupos de distribuição somente podem ser usados para enviar emails e **não** serão mostrados ao se definir as permissões de segurança de uma pasta.

O ZCP 6.40 e versões superiores possuem suporte para grupos próprios.

Os grupos podem ser criados utilizando-se a ferramenta de criação padrão de grupo no Active Directory.

8.5.2.3. Criando contatos utilizando ADS

O Catálogo Global de Endereços pode ser aumentado com contatos. Os contatos são endereços SMTP externos os quais são mostrados no Catálogo Global de Endereços e podem ser utilizados como membros de lista de distribuição.

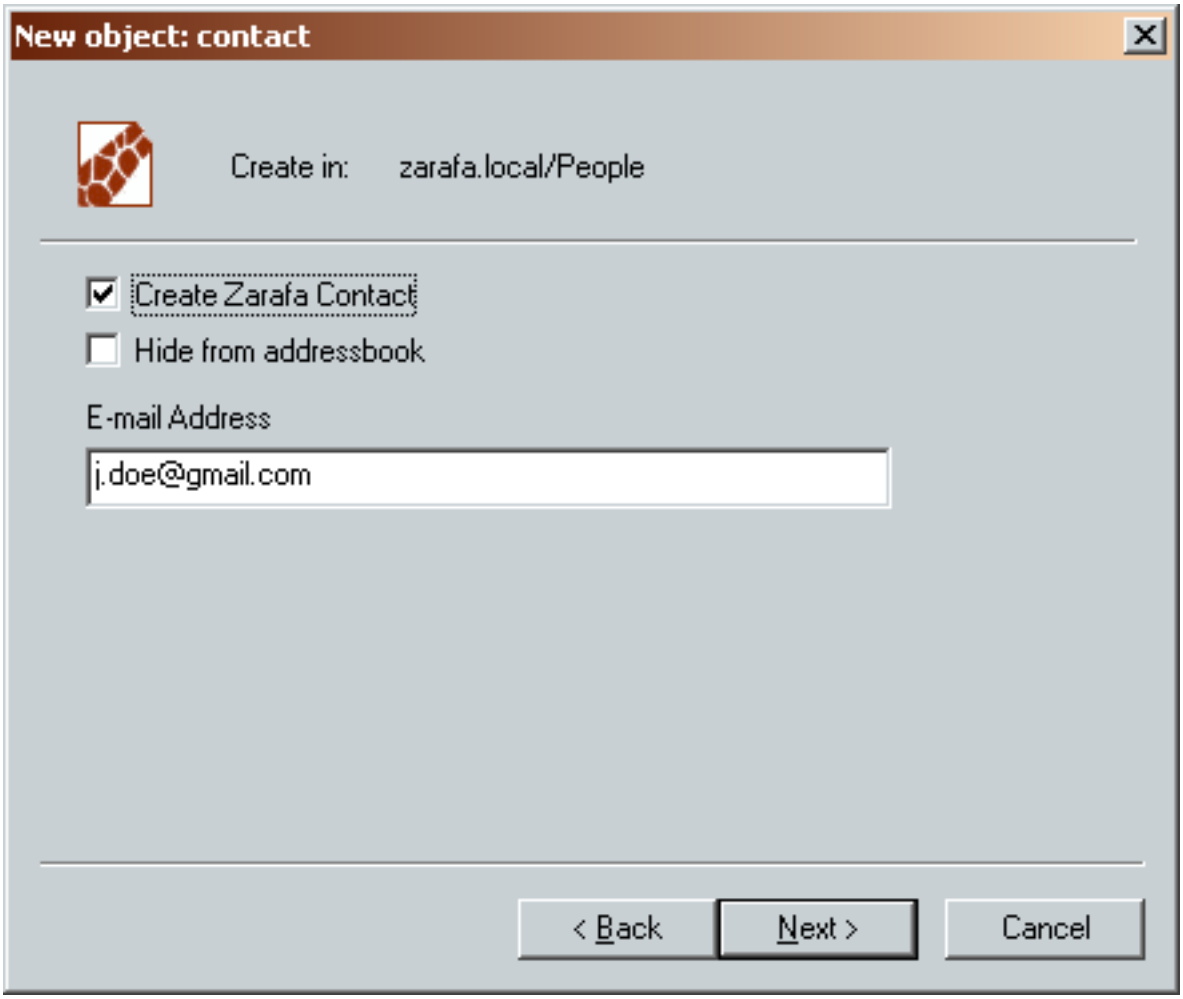


Figura 8.2. Ferramenta de criação de contato

8.5.2.4. Configurando permissões sendas utilizando o ADS

As permissões sendas podem ser configuradas tanto por usuários como por contatos. Os usuários ou grupos devem ser capazes de sendas um endereço específico precisam ser adiconados na lista de privilégio sendasdo usuário ou contato.

Para verificar se as permissões estão definidas corretamente, utilize:

```
zarafa-admin --list-sendas <username>
```

Por exemplo:

```
zarafa-admin --list-sendas helpdesk
Send-as list (1) for user helpdesk:
Username Fullname
----------------------------
john John Doe
```

Os usuários que têm as permissões sendas devem agora ser capazes de adicionar o outro endereço no campo "De" e "sendas" esta conta.

Desde o ZCP 6.40 o sistema de sendas está modificado:

- Configurar as permissões de sendas é a outra maneira de resolver isto ao invés das versões anteriores do Zarafa. As permissões sendas agora precisam ser configuradas no usuário que está selecionado como o endereço DE.
- Ver *Seção 3.5.1, "Etapas de pré atualização da versão 6.40 do Zarafa"* sobre conversão de permissões sendas.
- Os Grupos agora podem também ser usados para definir permissões sendas.

8.5.2.5. Enviando como pseudônimo de usuário

In Active Directory multiple email addresses can be added to each user via the Zarafa tab. These aliases will automatically be available for use with the send-as functionality of Zarafa.

8.5.2.6. Configurando listas de endereços no ADS

As listas de endereços são subpartes do Catálogo Global de Endereços que seguem um critério específico. Por exemplo, você pode criar uma lista de endereço que conhenha todos os usuários em Manchester e outra que contenha todos os usuários de Stuttgart.

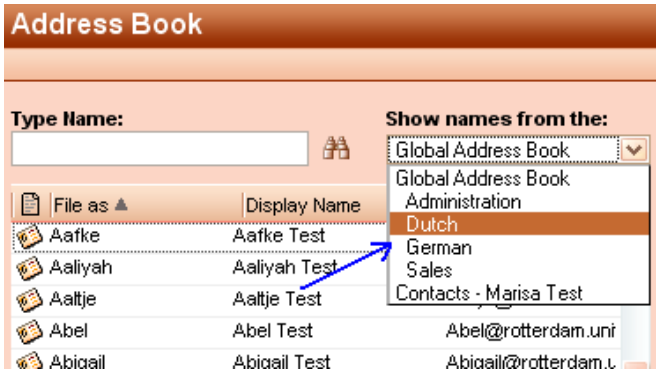


Figura 8.3. Listas de endereços no Catálogo de endereços

To setup an addresslist in Active Directory, it is required to have the Zarafa ADS plugin installed.

1. Selecione uma pasta na raiz do Active Directory do console de Usuários ou Grupo
2. Crie a nova lista de endereços clicando em **`Action > New > Zarafa Addresslist`**
3. Insira o nome da lista de endereços
4. Abra as propriedades da lista de endereços recém criada
5. Adicione um filtro de pesquisa para o endereço, ver *Seção 8.6, "Exemplos de Condição do LDAP"* para exemplo de questões de condição.

8.5.2.7. Omita informações do Catálogo Global de Endereços com o ADS

From ZCP 6.40, it is possible to hide users, contacts or groups from the Global Address Book. Hiding information from the Global Address Book can be done by the checkbox **`Hide from addressbook`** option in the Zarafa tab in Active Directory .

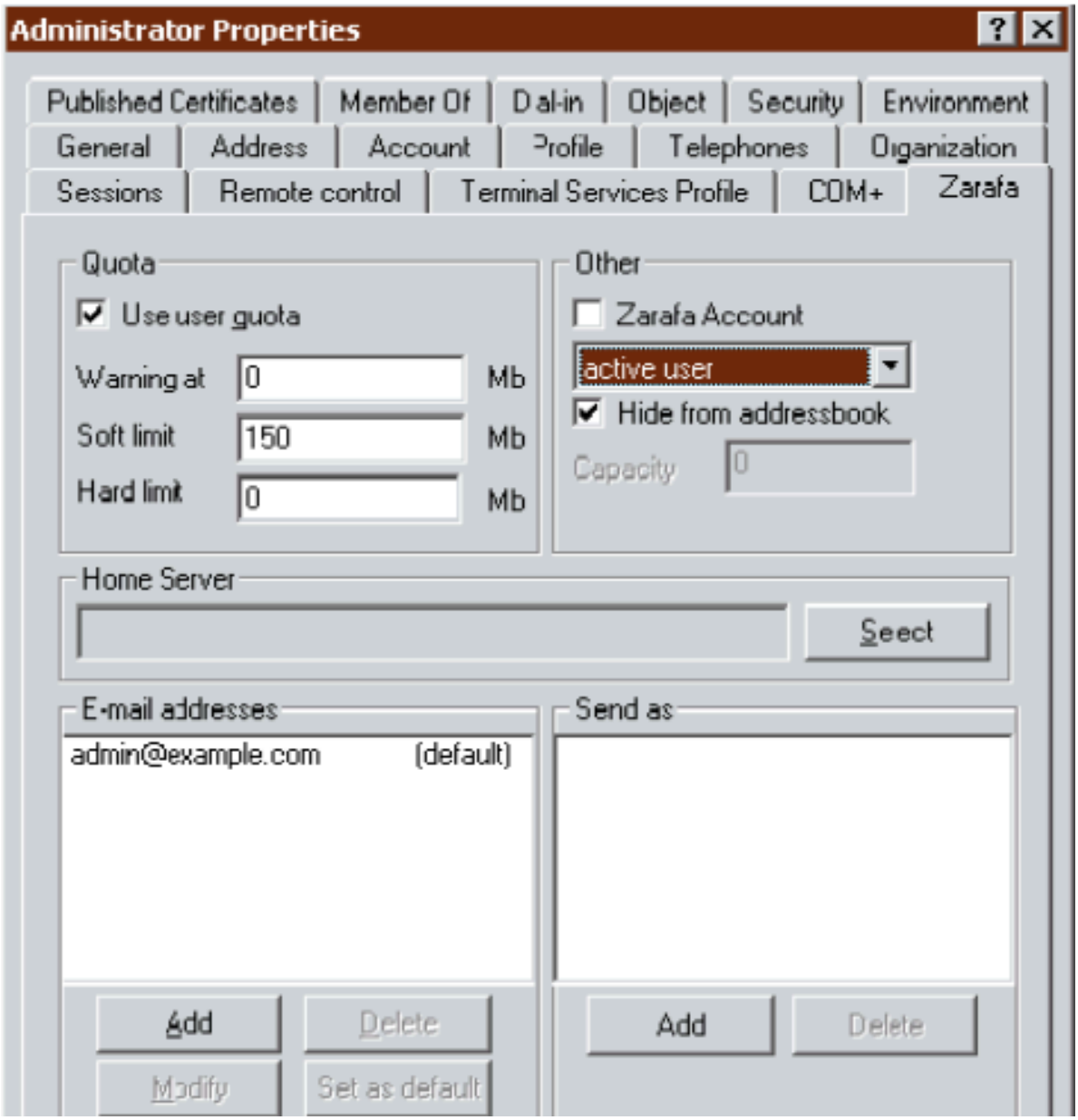


Figura 8.4. Omita um usuário do Catálogo Global de endereços utilizando o Active Directory

Nota

O usuário interno do Sistema e o grupo Todos podem ser omitidos no **`/etc/zarafa/server.cfg`**.

8.5.3. Manuseio de usuário à partir do OpenLDAP

8.5.3.1. Criando grupos utilizando o OpenLDAP

Usuários e grupos podem ser criados utilizando-se uma administração padrão OpenLDAP, por exemplo **`phpldapadmin`** ou a ferramento do Windows **`ldapadmin`**.

Para configurar informações específicas do Zarafa para o usuário, o objectClass zarafa-user precisa ser adicionado ao usuário. Adicionar este objectClass permite que você adicione atributos do Zarafa ao usuário, como definições de quotas, permissões sendas, tipo de caixa de correio.

8.5.3.2. Criando grupos utilizando o OpenLDAP

Os grupos criados no OpenLDAP serão utilizados por padrão como grupos de segurança no ZCP. Os grupos de segurança podem ser utilizados para definir permissões e enviar emails. Grupos de distribuição somente podem ser utilizados para enviar emails e **não** serão mostrados quando as permissões de segurança de uma pasta forem definidas.

Para trocar um grupo para um grupo de distribuição o atributo **`zarafaSecurityGroup`** precisa ser definido para **`0`**.

8.5.3.3. Criando contatos utilizando o OpenLDAP

O Catálogo Global de Endereços podem ser extendidos com contatos. Os contatos são endereços SMTP externos típicos e podem ser utilizados como membros de lista de distribuição.

Os contatos precisam ter o mesmo atributo exclusivo dos usuários. Favor verificar o **`ldap_unique_user_attribute`** no arquivo ldap.cfg para o atributo correto.

8.5.3.4. Configurando permissões sendas utilizando o openLDAP

As permissões sendas podem ser configuradas tanto para usuários quanto para contatos. Os usuários ou grupos que devem ser capazes de sendas um endereço específico precisam ser adicionados na lista de privilégio do sendas.

Para verificar se as permissões estão definidas corretamente, utilize:

```
zarafa-admin --list-sendas <username>
```

Por exemplo:

```
zarafa-admin --list-sendas helpdesk
Send-as list (1) for user helpdesk:
Username Fullname
----------------------------
john John Doe
```

Os usuários que têm as permissões sendas devem agora ser capazes de adicionar o outro endereço no campo "De" e "sendas" esta conta.

Desde o ZCP 6.40 o sistema de sendas está modificado:

- Configurar as permissões de sendas é a outra maneira de resolver isto ao invés das versões anteriores do Zarafa. As permissões sendas agora precisam ser configuradas no usuário que está selecionado como o endereço DE.
- Ver *Seção 3.5.1, “Etapas de pré atualização da versão 6.40 do Zarafa”* sobre conversão de permissões sendas.
- Os Grupos agora podem também ser usados para definir permissões sendas.

Nota

Ao utilizar grupos para permissões sendas, tenha certeza que o **`ldap_sendas_attribute_type`** esteja definido para **`dn`**. Veja a seguinte configuração LDAP:

```
ldap_sendas_attribute = zarafaSendAsPrivilege
ldap_sendas_attribute_type = dn
ldap_sendas_relation_attribute =
```

8.5.3.5. Configurando listas de endereços no OpenLDAP

As listas de endereços são subpartes do Catálogo Global de Endereços que seguem um critério específico. Por exemplo, você pode criar uma lista de endereço que conhenha todos os usuários em Manchester e outra que contenha todos os usuários de Stuttgart.

Para configurar uma lista de endereços no OpenLDAP, siga estes passos:

1. Crie uma **`Organisation Unit`** para todas as listas de endereços na árvore LDAP.
2. Crie um novo objeto LDAP e adicione o objectClass **`zarafa-addresslist`**
3. Defina o atributo cn para o nome exclusivo da lista de endereços
4. Crie uma pergunta de condição no atributo **`zarafaFilter`**, ver *Seção 8.6, "Exemplos de Condição do LDAP"* para exemplos de perguntas de condição.

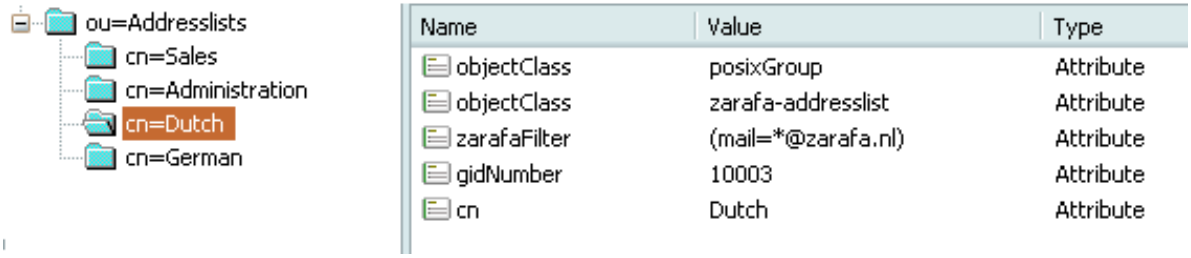


Figura 8.5. Listas de endereço no LDAP

Após reiniciar o **`zarafa-server`**, as listas de endereço devem ser visíveis no livro catálogo global de endereços.

8.5.3.6. Omita informações do Catálogo Global de Endereços com o OpenLDAP

From ZCP 6.40, it is possible to hide users, contacts or groups from the Global Address Book.

Para omitir uma informação do Catálogo Global de Endereços, defina o atributo **`zarafaHidden`** no OpenLDAP para **`1`** em um objeto específico.

Nota

O usuário interno do Sistema e o grupo Todos podem ser omitidos no **`/etc/zarafa/server.cfg`**.

8.6. Exemplos de Condição do LDAP

Tanto para as listas de endereços quanto para grupos dinâmicos um filtro LDAP precisa ser especificado. Por exemplo, o Catálogo Global de Endereços contém usuários holandeses e alemães. É possível ver estes usuários por país criando duas listas de endereços da árvore do LDAP. Todos os usuários alemães possuem o domínio *example.de* em seus endereços de email e todos os holandeses tem *example.nl*.

Nesta situação, a condição (**mail=*@example.de**) é usada para a lista de endereços alemã e (**mail=*@example.nl**) para a lista de endereços holandesa.

Qualquer combinação com atributos do LDAP é aplicável. Este exemplo seguinte seleciona todos que são administradores do Zarafa e têm o caracter **p** no valor de **cn**.

```
(&(cn=*p*)(zarafaAdmin=1))
```

Este exemplo seleciona todos os usuários com endereço de email *piet@example.com* or *klaas@example.com*.

```
(|(mail=piet@example.com)(mail=klaas@example.com))
```

8.7. Manuseio do Zarafa Feature

Algumas características dentro do ZCP podem ser desabilitadas. Por padrão, todas as características são desabilitadas. A habilitação pode ser feita globalmente ou na base por usuário. Quando uma característica tiver sido desabilitada globalmente, você pode habilitá-la em uma base por usuário também. Atualmente as únicas características que podem ser controladas são ‘imap’ and ‘pop3’.

Se a característica ‘pop3’ for desabilitada, os usuários não serão capazes de acessar utilizando o protocolo POP3. O mesmo vale para a característica ‘imap’, mas isto também tem um efeito extra. Quando um usuário recebe email quando a característica ‘imap’ está habilitada, o email original e outros dados otimizados de imap também serão salvos no banco de dados e diretório de anexos do Zarafa. Isto tornará os serviços IMAP disponibilizados pelo zarafa-gateway mais confiáveis. Por outro lado, também utilizará mais espaço do disco. Desabilitar a característica ‘imap’ então economizará espaço do disco.

A seguinte tabela mostrará quando um usuário pode utilizar o IMAP ou o POP3.

Tabela 8.1. Visão geral do controle do Access

	Serviço habilitado para o usuário	Serviço desabilitado para o usuário	Nada configurado para o usuário
Serviço listado no disable_feature no arquivo server.cfg			
Serviço não listado em disable_feature no arquivo server.cfg			

8.7.1. Habilitando características globalmente

Para habilitar uma característica específica, edite a definição **disabled_features** na configuração de seu servidor:

```
disabled_features = imap pop3
```

8.7.2. Habilitando ou desabilitando características por usuário

O manuseio da característica por usuário depende do plugin de usuário utilizado. Para os plugins **db** e **unix** a ferramenta **zarafa-admin** precisa ser utilizada para controlar as características:

```
zarafa-admin -u john --enable-feature imap
 zarafa-admin -u john --disable-feature pop3
```

Para definições do Active Directory ou OpenLADP(utilizando o plugin de usuário **`ldap`** ou **`ldapms`**), as características serão manuseadas à partir do dois atributos LDAP **`zarafaEnabledFeatures`** e **`zarafaDisabledFeatures`**. Tenha certeza que o último arquivo de esquema ou plugin do Acvice Directory esteja instalado antes de usar estes atributos. Estes atributos de multi-valor podem conter qualquer sequência, mas somente as características reconhecidas pelo Zarafa serão de fato disponibilizadas através do sistema.

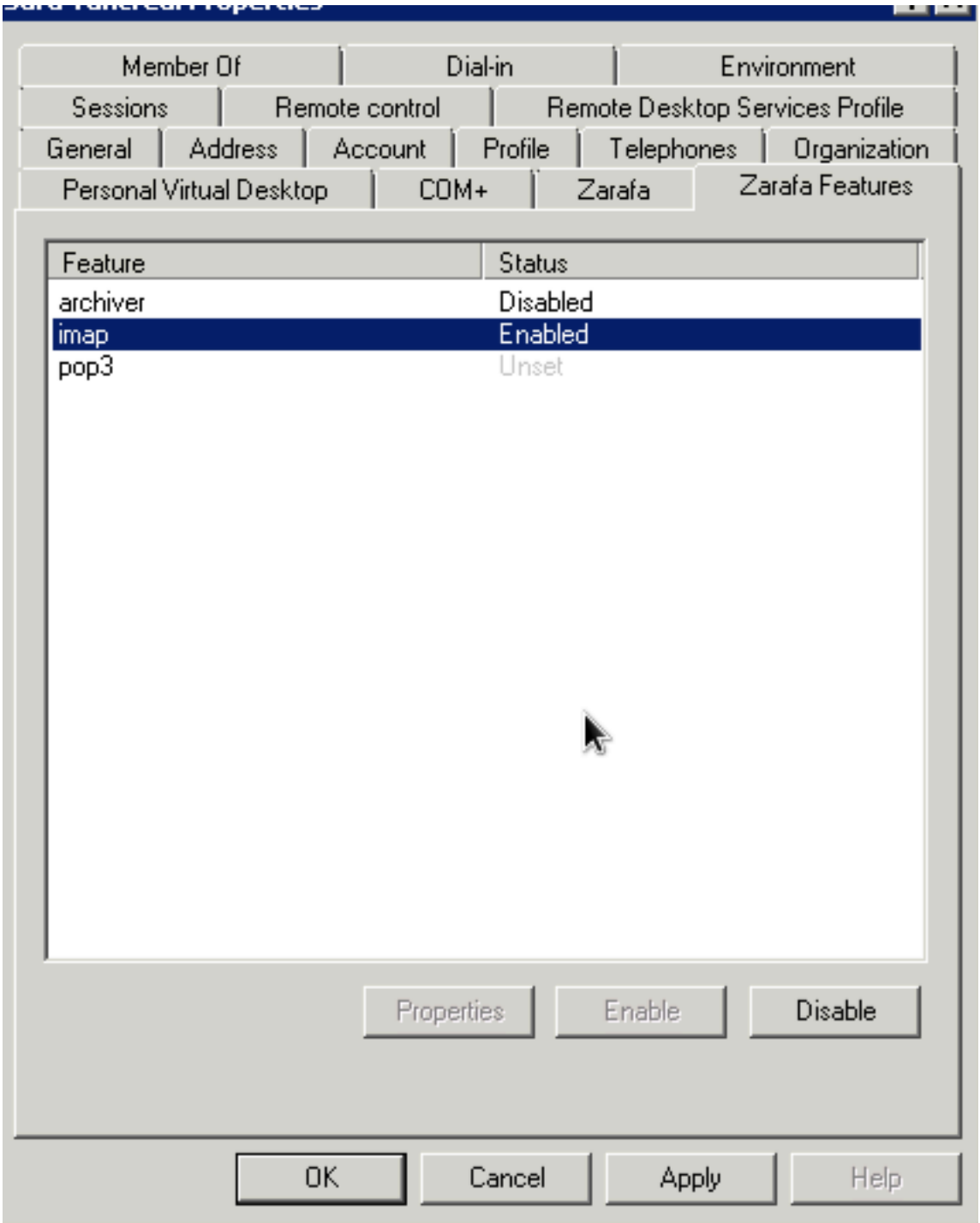


Figura 8.6. Aba de características em ADS

Nota

Tenha certeza que uma característica particular não esteja listada tanto no zarafaEnabledFeatures quanto no zarafaDisabledFeatures. A consistênia não será garantida.

8.8. Configuração de recurso

ZCP soporta reserva automática de recursos, como projetores, salas, ou outros equipamentos. Para criar um recurso, adicionae uma nova caixa de correio desativada ou selecione no Active Directory ou OpenLADP o tipo de usuário do recurso.

Antes que um recurso possa ser inscrito por usuários, o recurso precisa ser configurado para aceitar requisições de reuniões automaticamente. A aceitação automática de requisição de reunião pode ser configurada de duas maneiras; utilizando a ferramenta zarafa-admin ou utilizando o cliente Outlook.

Para configurar o recurso à partir do Outlook, siga os seguintes passos:

- Faça o recurso temporariamente ativo
- Acesse como o recurso no Outlook
- no menu Ferramentas, clique em Opções e depois em Opões de Calendário.
- Abaixo de opções avançadas, clique Agendamento de Recurso
- Habilite a aceitação automática de requisição de reunião
- Se o recurso não aceitar a inscrição dupla do recurso ou inscrições de reuniões recorrentes, as opções "Decline recurrencing meeting request" e "Decline conflicting meeting requests" devem ser habilitadas.
- Configure as permissões no calendário do recurso, para que os usuários possam inscrever o recurso. Os usuários devem ao menos escrever permissões para o calendário do recurso.

Para configurar o recurso com a ferramenta zarafa-admin, use o seguinte comando:

```
zarafa-admin -u <resource name> --mr-accept 1
```

O recurso agora aceitará automaticamente as requisições de reunião. Para não aceitar a inscrição dupla ou reunião recorrente, use:

```
zarafa-admin -u <resource name> --mr-decline-conflict 1
 zarafa-admin -u <resource name> --mr-decline-recurring 1
```

After the automatic acception of meeting requests is configured, make sure the users have at least write permissions on the calendar of the resource. The permissions can be configured by opening the resource mailbox to an administrator user and setting the permissions.

Para reservar automaticamente um recurso, tenha certeza que a opção do recurço esteja realmente selecionada nos horários Freebusy quando agendar uma reunião.

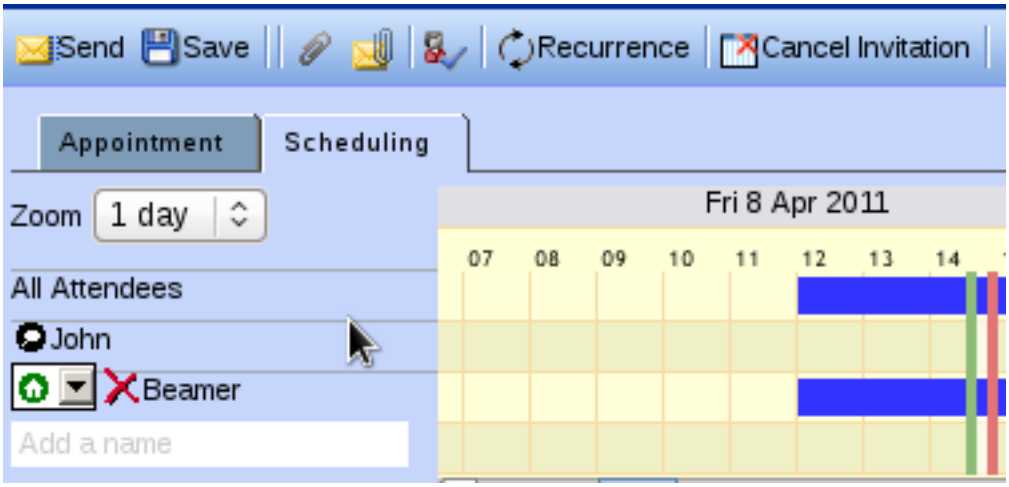


Figura 8.7. Opção do recurso nas horas Freebusy

8.8.1. Metodos de reserva de recurso

Há dois métodos para se reservar recursos:

1. Reserva direta
2. Reserva de requisição de reunião

Ambos os métodos são utilizados para reservar recursos. O resultado final é que o usuário pode reservar um recurso, depois do qual o calendário mostrará que ele está ocupado para o horário alocado. Ambos os métodos suportam o não-aceite de reuniões recorrentes e conflitantes, mas a forma que eles funcionam diferem em várias maneiras:

Tabela 8.2. Tabela Comparação dos métodos de reserva de recurso

Reserva direta	Reserva MR
Reservas diretamento no calendário alvo	Envia requização de reunião a qual é respondida para
Precisa de acesso para ler/escrever para o calendário do recurso	Não precisa de permissão para ler ou escrever para o calendário do recurso
Possível limitar os reservantes através de permissões	Não é possível limitar os reservantes
Não suporta múltiplos recursos utilizando o mesmo calendário	Possível definir limite de reserva dupla para 2 ou mais por equipamento
Não funciona com reservantes externos	Funciona com reservantes externos

8.8.1.1. Reserva direta

Reserva direta é o método de reserva de recurso padrão para:

- Outlook 2000 - Outlook 2007
- Zarafa WebAccess

A maneira que isto funciona é que a aplicação do cliente:

1. Abre o calendário do recurso
2. Verifica o calendário para disponibilidade

3. Cria um compromisso no calendário

4. Notifica o usuário que o recurso já foi reservado

Isto tem a contrapartida que o cliente precisa ter acesso de escrita para o calendário. Isto, contudo, significa que o usuário fazendo a reserva poderia, teoricamente, também reservar outros compromisso no calendário do resurso sem aderir aos requirimentos (e.g. reserva dupla de uma sala).

No outlook 2010, o método de reserva padrão foi mudado para reserva baseada em MR. Ele pode ser reabilitado em uma base por usuário adicionando a seguinte chave de registro:

```
HKEY_CURRENT_USER\Software\Microsoft\Office\14.0\Outlook\Options\Calendar\EnableDirectBooking
 = (DWORD) 0x00000001
```

Outras versões do Outlook também suportam a chave de registro para *desabilitar* a reserva direta.

Para mais informaões, veja *http://support.microsoft.com/kb/982774*

8.8.2. Reserva de requisição de reunião (MR)

Nota

A reserva MR foi introduzida no Zarafa 7.0.3. Tentar utilizar a reserva MR em versões anteriores à 7.0.3 resultará na não confirmação de todas as requisições de recurso restantes, assim como os utens que não estão sendo reservados, no calendário do resurso.

Reserva oir requerimentos de reunião funciona exatamente da mesma forma que enviar uma requisição de reunião para outro usuário. Ao reservar o recurso, um usuário envia uma requisição de reunião para o recurso em um e-mail. O recurso então recebe o e-mail, verifica seu própria disponibilidade e responde à requisição da reunião da mesma forma que um usuário humano faria; o reservante recebe uma resposta de reunião *Aceita* ou *Negada* por e-mail.

Isto significa que quando a reunião é enviada aos participantes,o recurso ainda não foi de fato reservado; é possível que outro usuário tenha reservado o recurso durante o processo, resultando em uma resposta de *negado* do recurso. O reservante precisa então remarcar e enviar uma atualiz~ção para todos os participantes.

A principal vantagem deste método é que o reservante não precisa ter permissões para escrever no calendário do recurso. Também, o Métoro MR permite um manuseio mais flexível de requisições de reunião. Por exemplo, se um usuário tem 5 projetores, os quais foram cirados como um *resource*, então eles poderiam ser criados como 5 diferentes recursos, cada um seria normalmente reservado diretamente. Contudo, isto iria requerer que o usuário procurasse por um projetor libre e reservasse aquele projetor específico.

Com a reserva MR, o administrador pode definir a capacidade do equipamento para um número diferente de 1, por exemplo 5 neste cado. O administrador então precisa somente de um recurso com uma capacidade de 5 para representar todos os projetores. Quando o MR é processado pelo recurso, ele checa se *todos* os projetores foram reservados naquele momento, somente negando quando todos os 5 projetores não estiverem disponíveis naquele momento.

Favor notar que você *precisa* utilizar o tipo de *equipamento* para seu recurso se você deseja utilizar a característica de capacidade. A capacidade dos recursos da *sala* é ignorada (você não pode reservar duas vezes uma sala).

A reserva MR é processada pelo script zarafa-mr-accept, o qual é instalado por padrão. Este script é ativado pelo zarafa-dagent tanto no modo direto quanto LMTP quando a definição mr-accept do usuário de destino é definipara para VERDADEIRO e a mensagem de volta é uma requisição de reunião ou cancelamento de reunião. Se o script zarafa-mr-accept falhar, o processamento de entrega é feito como de costume, possivelmente ativando regras de entrega e mensagem de out-of-office.

Nota

In rare cases *zarafa-mr-accept* prints out a warning about using localtime(). This relates to the - per default - unspecified *date.timezone* variable of php.ini. Setting it to for example *date.timezone = Europe/Berlin* fixes these messages.

8.8.3. Definindo o método de reserva de recurso

No Outlook, o método de reserva pode ser definido donfigurando-se

```
HKEY_CURRENT_USER\Software\Microsoft\Office\<OUTLOOK VERSION>\Outlook\Options\Calendar
\EnableDirectBooking = (DWORD) 0x00000001
```

Isto irá habilitar ou desabilidar a reserva direta. Desabilitar a reserva direta implica que a reserva MR será utilizada.

Para o Zarafa WebAccess, você pode definir o método de reserva definindo

```
define('ENABLE_DIRECT_BOOKING', true)
```

em config.php

Isto habilitará ou desabilitará a reserva direta, espelhando o comportamento no Outlook. Se você desabilitar a reserva direta, a Reserva MR será utilizada.

8.9. Out of office management

Users can normally manage their out of office replies from the Outlook, webclients and certain mobile devices. Sometimes users forget to turn on their out of office reply or out of office replies should be enabled for shared mailboxes.

For these purposes ZCP 7.1 is shipping a commandline utility to manage out of office replies.

To use the utility, use the following command:

```
zarafa-set-oof -u <username> -m 1|0 -t "Out of office subject" -n <path to out of office
 text>
```

To enable an out of office reply for the user john use:

```
zarafa-set-oof -u john -m 1 -t "I'm on holiday till the 30th of June" -n /tmp/oof.txt
```

Other options can be gathered from the help of the script. This can be reached when the script is called without any arguments.

8.10. Realocador de estoque da caixa de correio

Para se mover caixas de correio entre diferentes nódulos de multi-servidor, o realocador de armazenagem de caixa de correio está disponível. A ferramenta **zarafa-msr** deve ser utilizada para reallocar caixas de correio de um nódulo de multi-servidor para outro.

A ferramenta **zarafa-msr** se conectará ao servidor de apoio do usuário (LDAP/AD) como definido no arquivo do Zarafa server.cfg. Ela pedirá a atual configuração do servidor doméstico daquele usuário para o servidor de apoio. Ela então se conectará àquele servidor doméstico e migrará toda a armazenagem de correio para o novo servidor doméstico especificado no arquivo de configuração msr. Após a migração, a ferramenta **zarafa-msr** manterá as duas armagenagens de correio em sincronia uma com a outra.

The **zarafa-msr** is not only migrating items and folders, but also permissions, rules and settings.

Nota

O **zarafa-msr** somente pode ser utilizado em configurações multi-servidor. O suporte a multi-servidor está disponível nas esdições Zarafa Enterprise e Hosted.

Importante

When the **zarafa-msr** will be used for large scale migrations, please contact Zarafa Professional Services for advise on the recommended setup.

8.10.1. Pré-requisitos

- Python 2.5 ou superior
- Python MAPI binding
- Zarafa 6.40.5 ou superior

8.10.2. Invocação

O único argumento requerido pelo **zarafa-msr** é um arquivo de configuração especificando os detalhes da operação de realocação.

```
zarafa-msr msr.cfg
```

Quando o **zarafa-msr** tiver terminado de realocar todas as caixas de correio, ele irá imprimir a seguinte mensagem:

```
"x migrações foram completadas com sucesso, mantendo a sinc."
```

onte x significa o número de caixas de correio migradas. O administrador pode agora parar o **zarafa-msr** pressionando Ctrl-C.

zarafa-msr pode ser parado de maneira segura a qualquer momento pressionando-se Ctrl-C. Da próxima vez que for rodado, ele continuará à partir de onde parou.

Se ele não foi parado pressionando-se Ctrl-C, zarafa-msr manterá a sincronização rodando sem fim.

A ferramenta **zarafa-msr** pode ser rodada em qualquer destino ou no servidor da fonte. Ou, embora sem a devida eficiência, em qualquer outro nódulo na configuração multi-servidor.

Nota

It's recommended to disable mailbox quotas on the destination server during the migration.

8.10.3. Atualizando o LDAP/ADS

Há duas situações nas quais é seguro atualizar os servidores domésticos para os usuários dos quais as caixas de correio tenham sido realocadas:

1. **zarafa-msr** está ainda em andamento. Neste caso todas as mudanças na caixa de correio original continuação a ser propagadas para a nova caixa de correio.

2. Nenhuma mudança da caixa de correio original pode acontecer.

8.10.4. Configuração

Um arquivo de configuração típico se parece com este:

```
[Connection]
serverpath: file:///var/run/zarafa
sslkey_file: ssl.cert
sslkey_pass: pass

[Servers]

[Mapping]
user1: https://server2:237/zarafa
user2: https://server1:237/zarafa

[Logging]
log_file: /var/log/zarafa/msr.log
```

Nota

In the directory /usr/share/doc/zarafa-multiserver/example-config an example msr.cfg can be found.

8.10.4.1. Seção de Conexão

A seção de **Connection** contem informações sobre como se conectar a um nódulo em particular no grupo multi-servidor. Esta seção é obrigatória.

Tabela 8.3. Opções da seção de conexão

Opção	Valor padrão	Descrição
atalho do servidor	*file:///var/run/zarafa*	Atalho para o servidor. Pode ser qualquer nódulo no grupo.
sslkey_file	-	Atalho para o arquivo chave SSL.
sslkey_pass	-	Senha para a chave SSL especificada com **sslkey_file**.

Opção	Valor padrão	Descrição
bidirectional	yes	When enabled changes in the destination mailbox will get synced back.
force_source	no	When enabled the msr won't redirect to source server from LDAP information
workers	4	Amount of concurrent sync worker threads

8.10.4.2. Seção de Servidores

A seção de **`Servers`** é uma seção opcional que conté uma lista de pseudonimos de servidor. Esses pseudônimos podem ser usados na seção de **`Mapeamento`** quando muitas caixas de correio são realocadas para o mesmo servidor.

A seção **`Servidores`** não tem opções pré-definidas. Ao invés disso, o formato é

```
ever_alias: server_path
```

Podem ser colocados nesta seção tantos itens quanto forem necessários.

8.10.4.3. Seção de Mapeamento

A seção **`Mapping`** contém a lista de nomes de usuários e o nódulo de destino de suas caixas de correio. O nódulo de destino pode ser um caminho de servidor completo ou um pseudônimo especificado na seção **`Servidores`**.

A seção de **`Mapeamento`** não possui opções pré-definidas. Ao invés disso, o formato é

```
username: destination_node
```

Podem ser colocados nesta seção tantos itens quanto forem necessários.

Para realocar o armazenamento público, um nome especial deve ser utilizado para o nome de usuário:

1. Em um ambiente multi-locatário, o nome do locatário para o qual se dejesa realocar a armazenagem pública precisa ser usado.

2. Em um ambiente multi-locatário, o nome especial **`__public__`** tem que ser usado.

8.10.4.4. Seção de Log

The **`Logging`** section is optional and contains logging specific settings. Currently the only setting is the **`log_file`** setting, which allows an alternate log file to be selected. By default a file called **`zarafa-msr.log`** will be created in the working directory.

8.10.5. Post migration steps

The **`zarafa-msr`** will migrate the complete mailbox including all settings to the destination node. However the **`zarafa-msr`** will not migrate the sync state of the user. The sync state is used for Z-Push users, Blackberry users and offline Outlook users.

This means all Z-Push users need to reinitialize their device after they are migrated. On some mobile devices a full resync can be done, however on iPhones and Ipads the whole Activesync profile has to be deleted and recreated. Users with a Blackberry device need to be removed and added again in the Blackberry Enterprise Server administration console.

Users with an offline Outlook profile will get an automatic resync triggered after the msr migration. The resync will reinitialize the sync state on the new serverm, so all changes get synced to the Outlook client.

As the **zarafa-msr** will not remove the source mailbox when the migration is finished, the administrator should remove it. On the source server the following commands can be used to cleanup the migrated mailboxes:

```
zarafa-admin --unhook-store <username>
zarafa-admin --list-orphans
```

Now use the store GUID to completely remove the mailbox:

```
zarafa-admin --remove-store <store GUID>
```

Ajuste de desempenho

Ao instalar um servidor Linux com o Zarafa, é impressindível que o MySQL esteja configurado corretamente para obter a máxima performance no servidor; quase todos os gargalos de desempenho estão dentro do próprio acesso ao banco de dados, portanto, assim que começarem as peerguntas do SQL, é muito importante configurar para que a execução de dê o mais rápido possível.

Para grandes instalações, é altamente recomendável ajustar os parâmetros de cache do Zarafa também; Estes são normalmente set bastante baixo para se certificar de que Zarafa possa ser executado em servidores de baixo custo relativo, mas a não ser em no caso de as instalações muito pequenas, esses padrões precisam ser elevados. Qualquer instlação com 50 usuários ou mais, deve definir os parâmetros de cache para a máxima performance.

Esse documenta considera que a principal função do servidor é executar o Zarafa. Certifique-se sempre que outros fatores são levados em conta - por exemplo, um sistema anti-spam ou um servidor rodando um site diferente do Zarafa WebAccess.

Maiores informações sobre o ajuste de performance podem ser encontradas em *http://wiki.zarafa.com*.

9.1. Considerações sobre hardwares

Também existem diferentes configurações de hardwares a serem consideradas ao configurar o servidor Zarafa. Discutiremos os vários tipos de hardware que afetam o desempenho do Zarafa.

9.1.1. Uso da memória

Ajustar o uso da memória é uma das melhores formas de aumentar a performance do servidor; como a memória RAM geralmente é barata, usar, corretamente, uma grande quantidade de RAM no servidor pode melhorar o desempenho por ordem de magnitude.

Por outro lado, configurando um uso muito alto de RAM pode fazer com que o servidor troque partes da memória que precisarão ser trocadas de volta mais tarde, causando uma desaceleração grande em todas as partes do servidor. Portanto, é importante definir o uso de RAM de vários componentes para uma configuração alta o suficiente para usar a memória RAM disponível e, ao mesmo tempo, não definir o uso de RAM muito alto.

Para fazer uso da memória RAM disponível da melhor forma possível, o Zarafa é projetado para usar apenas uma quantidade fixa de memória RAM física, o uso de memória não aumenta proporcionalmente a cada usuário que se conecta, mas apenas em uma pequena quantidade - a maior parte do uso de memória é devido ao cache de configurações no arquivo de configuração. Isto torna muito fácil o controle da quantidade exata de memória que será usada em uma situação ao real, e pode se ter certeza que a quantidade real de memória RAM usada nunca superará em muito os valores estabelecidos.

Então, em geral, o uso otimizada de memória RAM é o mais alto possivel, desde que o sistema não precise trocar importântes partes de memória disponível.

É muito difícil da fixar um valor para o qual a distribuição de uso de memória é otimizado para determinado servidor, já que padrões de acesso a dados variam muito de servidor para servidor. Iremos descrever alguns parâmetros e fazer com que os padrões de uso memória RAM sejam esclarecidos da melhor forma possível.

9.1.2. Considerações sobre Hardwares

Em servidores que executam o Zarafa, o principal gargalo de desempenho é a rota entre os dados no disco rígido, e o tempo que leva para chegar ao cliente. Isto significa que, em geral, o desempenho de I/O é mais importante que o desempenho da CPU. Usando isto como base, as seguintes indicações podem ajudar na escolha do hardware correto para o sistema:

9.1.3. Mais memório significa mais velocidade

Mais memória RAM significa melhor cache e, portanto, maior velocidade.

O Zarafa foi especificamente desenvolvido para fazer uso da maior quantidade de memória RAM que estiver disponível nos servidores modernos. Por outro lado, é preciso ter em mente que num servidor Linux normal a maior quantidade de memória disponível em um servidor 32-bits que tem 3 gigas de memória PAE (physical address extension) disponível é suportada no kernel do CPU e placa-mãe. Se for necessário mais de 3Gb, sem algum tipo de limitação, use um sistema de 64 bits, um de 64 bits Linux OS, e um pacote de 64 bits Zarafa.

9.1.4. RAID 1/10 é mais rápdo que RAID 5

In general, a RAID1 or RAID10 array is faster at database accesses than RAID5 and RAID6. Zarafa strongly recommends not use the RAID5 or RAID6 configuration to prevent performance issues.

9.1.5. Alta velocidade de rotação (RPM) para um melhor desempenho do banco de dados

Os discos high-end SCSI ou SAS, em geral, têm alta velocidade de rotação; de 10K ou mesmo 15K RPMs. A velocidade de rotação dos discos afeta os tempos de busca. Embora, o formato de banco de dados do Zarafa seja otimizado para ter dados disponíveis sobre o disco de maneira sequenciada, e a maioria das as leituras sejam feitas na região exata do disco, o tempo de busca é ainda um fator de relativo á velocidade do I/O. Quanto maior a velocidade de rotação, menor o tempo de busca.

9.1.6. Hardware RAID

Controladores RAID de hardware geralmente possuem uma grande quantidade de RAM usado para cache. Isso pode aumentar a performance e throughput do subsistema de I/O. Se usado um controlador de RAID de hardware, tenha certeza que o cache de write-back não é usado ou que existe um sistema funcional de UPS e processos de shutdown do servidor estejam disponíveis já que em caso da falta de energia os dados contidos no write-cache poderão ser perdidos. Isso não afeta somente os dados que foram escritos neste momento, mas também poderá corromper os dados do innodb no disco.

9.2. Configuração do uso de memória

Existem basicamente quatro grandes partes da configuração do servidor que utilizam a memória do servidor:

- Cache do Zarafa no caso de aparelhos celulares (caches de dados individuais de aparelhos celulares dentro de uma exibição da tabela)
- Tamanho do buffer do MySQL (lê e grava os caches a partir do arquivo ibdata)
- Consulta de cache do MySQL (armazena as consutas SQL repitidas de maneira exata)

No caso de um servidor que executa exclusivamente o Zarafa, certifique-se estes caches estejam configurados para usar em torno de 80% da memória RAM do servidor. Os outros 20% devem estar livres para os processos do sistema, para outros processos (como MTA) e para o servidor web.

Uma regra geral impera que deve-se utilizar a seguinte distribuição de memória RAM:

Caches do Zarafa:

- **cache_cell_size:** cerca de 25% do tamenho total da memória RAM
- **cache_object_size:** CERCA DE 100KB por usuário
- **cache_indexedobject_size:** em torno de 512KB por usuário

These cache settings need to be configured in the **/etc/zarafa/server.cfg** file. To activate the cache size changes the Zarafa Server need to be restarted.

Definições do MySQL:

- **innodb_buffer_pool_size:** around 50% of total RAM size
- **mysql_query_cache:** 32MB
- **innodb_log_file_size:** 25% do innodb_buffer_pool_size
- **innodb_log_buffer_size:** 32M
- **innodb_file_per_table**
- **max_allowed_packet**: 16M
- **table_cache: 1000**

These settings need to be configured in the **/etc/my.cnf** or **/etc/mysql/my.cnf** file below the [mysqld] section.

It's recommended to change these MySQL settings before starting the Zarafa Server and migrating user data.

The most important settings will now shortly be described to illustrate the need of each of these cache settings.

9.2.1. Cache do Zarafa em relação a telefones celulares (cache_cell_size)

Os dados que são mostrados ao usuário nas exibições de tabela, passam através do *cache do celular*. Isto significa que qualquer ponto de vista de uma tabela no Outlook só vai recuperar as informações do banco de dados das células que não estão no cache. A vida útil Cache do é tão longa quanto o tempo de vida do servidor inteiro. Assim, ao abrir uma caixa de entrada duas vezes sucessivas no caso do segundo acesso provavelmente o disco não será acessado. É interesssante definir o cache de aparelhos celulares tão alto quanto poderem ser gerenciados, geralmente do mesmo tamanho que o tamanho do buffer do MySQL.

9.2.2. Cache dos objetos do Zarafa (cache_object_size)

O cache de objeto do Zarafa é usado para armazenar tabela de hierarquia. Cada objeto acessado será colocado no cache, o que torna mais rápido o processo de recuperar as informações novamente

sem que haja a necessidade de acessar o banco de dados.Quanto mais itens o usuários tiverem em suas pastas, mais importante se torna o cache. Desde que as informações sejam pequenas, o cache não precisa ser grande. Cerca de 1MB para 10 usuários é mais que o suficiente.

9.2.3. Zarafa indexedobject cache (`cache_indexedobject_size`)

Para abrir um item específico, o programa precisa enviar ao servidor uma chave única, chamada de **`entryid`**. Esse cache é um índice de 2 vias, da chave MAPI de banco de dados e vice-versa. A tradução das chaves é muito importante. Esse cache é preechido por pasta, de modo que pastas grandes podem empurrar para fora informações importantes. O uso normal é de cerca d MB por usuário.

9.2.4. MySQL `innodb_buffer_pool_size`

O buffer do MySQL é usado para armazenar leituras e gravações no arquivo ibdata. Numa máquina dedicada ao MySQL, isso seria algo entre 50% e 80% do tamanho fisico da memória RAM. Já quando o MySQL é executado na mesma máquina em que se executa o Zarafa, recomenda-se o uso de cerca de 25% do tamanho fisico da RAM (de modo que Cache Zarafa do celular também possa ser ajustado para este valor)

9.2.5. MySQL `innodb_log_file_size`

O **`innodb_log_file_size`** é o tamanho dos registros de transações. Por padrão, há dois logfiles. O tamanho preferencial para o **`innodb_log_file_size`** é de 25% do **`innodb_buffer_pool_size`**.

Importante

Customers the innodb_log_file_size tuning their existing MySQL installation are recommended to read this article *http://www.zarafa.com/wiki/index.php/MySQL_tuning* before performing this tasks. Tuning MySQL the wrong way can result in a database corruption.

9.2.6. MySQL `innodb_log_buffer_size`

O tamanho do **`innodb_log_buffer_size`** que o InnoDB utiliza para gravar os arquivos de registro no disco. Um grande buffer de registro permite que grandes transações sejam executadas sem que seja preciso gravar o registro no disco antes da submeter a transação. Se estiverem envolvidas grandes transações, fazendo com que o buffer de registro seja maior o disco I/O será salvo. Este valor deve ser de 25% do **`innodb_log_file_size`** .

9.2.7. MySQL `query_cache_size`

A cache de consultas do MySQL normalmente se encontra desativado. Ao ativar o cache de consultas é possivel que se obtenha um pequeno aumento de desempenho, mas não é necessário mais que alguns MBs, já que em geral as consultas SQL são bastante pequenas.

9.2.8. MySQL `innodb_buffer_pool_size`

A opção innodb_file_per_table criará um arquivo de dados innodb para cada banco de dados, ao invés de utilizar um grande arquivo ibdata para todos os dados. Ter um arquivo por lista dará mais flexibilidade para mover listas para diferentes partições do sistema de arquivos para melhor performance.

9.2.9. MySQL `max_allowed_packet`

The max_allowed_packet defines the maximum size of a single packet which can be inserted in the database. Customer changing this value to a higher value, should keep in mind the Outlook offline database is also using MySQL, which can cause client issues in case packets are larger than 16Mb.

9.3. Configuração de módulos em diferentes servidores

Existem várias partes do servidor Zarafa que podem ser hospedadas em diferentes serviidores. De fato, cada parte do servidor pode ser executada em um sistema diferente. No entanto, na prática, dividir todos os módulos do servidor em diferentes servidores não implica num aumento do desempenho. As principais partes que podem ser considaradas são:

- *Server1:* MySQL server
- *Server2:* Zarafa server
- *Server3:* MTA + AntiSpam/AntiVirus
- *Server4:* WebServer

se estas quatro peças forem hospedadas em quatro servidores, cada servdor iria se comunicar com os outros para trabalhar como um único sistema. Esta configuração pode ser feita com bastante facilidade, simplesmente configurando a várias partes do sistema para se comunicar com outro servidor.

No caso do servidor MySQL, isso só pode ser acessado através do processo do **`zarafa-server`** no *Server2*. Isso pode ser realizado facilmente, definindo o login correto e configurando o host no **`server.cfg`** do Zarafa.

O Servidor Zarafa vai se conectar automaticamente através do Cliente Outlook, *Server3* (MTA), e *Server4* (WebServer). Isso pode ser feito porque o processo do **`zarafa-server`** está sendo executado na porta **236** no *Server2*, e os outros servidores podem se conectar com a ele.

O *Server3* receberá e-mails na porta **25** ou buscará e-mail através de algum protocolo como o POP3. Depois de passar pelo antispam e pelo antivirus, o e-mail passara pelo processo **`zarafa-dagent`**. O processo **`zarafa-dagent`** pode ser configurado para se conectar a um certificado SSL através do *Server2*. Esse certificado SSL é necessário porque o **`zarafa-dagent`** precisa ser autenticado, já que ele está se conectando a diferentes servidores pela porta **236**. Quando essa configuração é realizada para ambos, o*Server3* e *Server2*, o e-mail pode ser entregue diretamente ao *Server2* pelo *Server3*.

O *Server4* é o servidor do WebAccess que executa o Apache e aceita conexões na porta **80** (ou **443**, no caso do SSL). O Zarafa WebAccess pode ser configurado (em**`config.php`**) para se conectar pela porta **236** (ou porta **237** no, caso do SSL) para o *Server2* para os dados reais. Uma vez que essa configuração foi realizada, este servidor está pronto para servir aos usuários. Nenhuma configuração adicional é necessária.

Backup & Restauração

Atualmente, o Zarafa conta com três formas de restaurar itens:

- Through the softdelete restore system
- Usando o brick-level backup system
- With a full database backup

10.1. Softdelete restore

The softdelete restore can be used by users from Outlook with the *Restore deleted items* dialog from the *Tools* menu to restore deleted items. This will cover most accidental deletions.

Os itens que foram deletados pelo usuário (através do processo de esvaziar a pasta de itens deletados ou com o comando Shift+Delete no Outlook), são simplesmente colocados no cache de itens deletados. Isso significa que o item não será realmente removido do banco de dados até que o tempo de retenção do item expire. Este tempo de expiração pode ser especificado na configuração do tempo do **`server.cfg`**, que por padrão está configurada para **`30`**.

Observe que a caixa de dialogo *restaurar itens excluidos* funciona na pasta atualmente selecionada.

In the following overview, which possibilities can be performed by whom, and when it is most likely used, can be seen.

Tabela 10.1. Opções de recuperação

Solicitação de restauração	Percentual de tempo gasto	Solução de Backup	Executante
Items < com até 30 dias	80 %	Sistema Softdelete	Usuário e Administrador
Items >= 30 dias	10 %	Bricklevel	Administrator
Itens de um remetente específico	5 %	Bricklevel	Administrator
Itens durante um período de tempo específico	3 %	Bricklevel	Administrator
Recuperação em caso de desastres	2 %	Despejo do MySQL	Administrator

Como se pode ver, as solicitações de restauração mais comuns podem ser executadas pelo próprio usuário. Isso se deve ao fato do sistema softdelete ser acessível através do Outlook.

Quando for solicitada a recuperação de mensagens mais antigas, o administrador terá que consultar os backups. Não é possível restaurar um item individual com o despejo do MySQL, portanto, esse é o ponto em que a ferramenta **`zarafa-backup`** atua.

O backups bricklevel da ferramenta **`zarafa-backup`** não contêm informações suficientes para recuperação de desastres. Um despejo de memória completo do banco de dados MySQL será necessário para realizar esse tipo de recuperação.

10.2. Despejo de memória do banco de dados completo

Todos os dados que são armazenados pelo Servidor Zarafa são encaminhados para o banco de dados do MySQL. Isso significa que caso seja necessária uma recuperação por desastre, tudo o que

é preciso pe uma restauração (backup) completa do banco de dados em questão. Isso também pode ser feito de várias maneiras, mas nesse manual nós explicaremos apenas duas maneira de realizar tal oeração.

10.2.1. Despejo do SQL através do mysqldump

O conteúdo de um banco de dados inteiro pode ser salvo como um arquivo usando o comando **`mysqldump`**. Existem, entretanto, algumas opções que são importantes nesse caso: a opção **`--single-transaction`** deve sempre ser especificada para o **`mysqldump`**. Quando o isso for realizado o **`mysqldump`** gravará uma única imagem do banco de dados no disco. Isso garantirá que quaisquer gravações feitas no banco de dados durante o backup não serão incluídas no backup. Com efeito, o despejo que é feito é "instantâneo" referente ao banco de dados no momento em que o despejo começou.

Quando estiver usando o **`mysqldump`**, é muito importante não fazer qualquer bloqueio da tabela. isso significa que as opçõess **`--opt`** e **`--lock-tables`** nunca devem ser usadas enquanto se estiver realizando o despejo de memória no banco de dados do Zarafa. a razão disso é que tais opções 'bloqueiam' as tabelas enquanto elas estão sendo despejadas no disco, o que impede quaisquer acessos ao banco de dados para "congelar" enquanto o backup é executado. Isto é, em primeiro lugar desnecessário e em segundo lugar pode fazer com que os e-mails que chegarem durante o backup sejam devolvidos (dependendo das configurações do MTA).

Um exemplo simples:

```
mysqldump --skip-opt ---single-transaction -p <database> > <dumpfile>
```

will make a consistent dump of the database.

10.2.2. Despejo de dados binários via LVM Snapshotting

Esta técnica utiliza o recurso 'LVM Snapshot' para, efetivamente, 'congelar' uma exibição binário do arquivo do banco de dados enquanto o banco de dados se mantém em execução. Esta visão "congelada" é, então, simplesmente binária copiada para um servidor remoto. Isso funciona porque o innodb garante que uma única imagem instantânea de um banco de dados seja sempre coerente (ou seja, ele será capaz de recuperar o banco de dados mysql, quando é iniciado nesta configuração de dados.)

Como o setup do LVM e a configuração do LVM para snapshots é um processo complexo, recomendamos ao leitor a documentação e ferramentas para a configuração de um volume LVM para os dados MySQL e como criar e remover partições de snapshot.

10.2.3. Backup de anexos

Ao usar o armazenamento de anexos num banco de dados externo, certifique-se de que tais anexos tambem sejam recuperados.

Alguns métodos de backup que podem ser usados para recuperar anexos:

- Rsync
- Copie todos os arquivos para o servidor de backup ou servidor externo anexo ao HD
- Utilize um agente de backup para Linux (comercial), como o SEP, Bacula, Arkeia entre outros

10.3. Backups Brick-level

As edições comerciais do Zarafa contam com uma ferramenta de Backup brick-level. Essa ferramenta criará um backup das caixas de e-mail para arquivos separados. Na segunda vez que um backup é realizado, somente as mudanças e os novos itens são adicionados ao backup.

Observe que esse tipo de backup não é significativo no caso de uma recuperação ocasionada em função de um desatre. Somente itens são gravados no backup. As informações sobre usuários, ou informações específicas sobre usuários, como regras, não são recuperáveis.

10.3.1. Formatar Backup

A ferramenta de backup cria dois arquivos para cada conta de e-mail: um arquivo de dados e um arquivo índice.

O arquivo indice contém informações sobre as pastas, a hierarquia e mensagens. Os campos são separados por vírgulas. Existem três tipos de entradas no arquivo índice:**`R`**, **`C`** e **`M`**. O **`R`** representa **`Root`**, e é sempre a primeira e a única entrada **`R`** do índice. Ela contém uma chave que as pastas usam como sua 'chave-mestra' para indicar que eles estão diretamente ligados ao contentor da conta.

O **`C`** representa **`Container`**, que pode ser qualquer tipo de pasta. Ela possui duas chaves, uma mestra e uma outra para identificar o próprio recipiente. Ela possui também uma chave única de restauração. Esta chave pode ser usada para selecionar a pasta para a ferramenta de restauração. Tem um indicador de quantos itens existem na pasta, uma modificação recente do unix timestamp, e um tipo de pasta (Por exemplo **`IPF.Note`**, para uma pasta de correio, **`IPF.Appointment`** para um calendário). A última parte de uma entrada **`C`** é o nome da pasta, que pode conter uma vírgula , e por isso é a última parte na entrada. Uma listagem detalhada dos campos para um **`Container`** pode ser encontrada no apêndice.

O **`M`** do índice significa **`Message`**, que pode ser qualquer tipo de mensagem ou item. Ele tem uma chave principalque corresponde a uma pasta-chave . Ele também dispõe de uma chave de restauração, que pode ser usada para restaurar essa mensagem específica. Um unix timestamp segue esta última modificação da **`mensagem`**. Se um usuário alterou a mensagem, esse registro vai ser atualizado. A entrada de índice continua com o tipo de **`mensagem`** (e-mail, calendário solicitação de reunião, etc). A entrada contém um deslocamento onde o item começa no arquivo de dados e, finalmente, contém o assunto do item. Uma vez que este assunto pode conter vírgulas, sua posição é no final da entrada. Uma listagem detalhada dos campos de uma **`mensagem`** pode ser encontrada no apêndice.

O arquivo de dados é um dump binário de todas as propriedades da mensagem, destinatários e anexos. As pastas são apenas um conjunto no arquivo de índice, portanto, apenas se faz o backup do nome da pasta, já que é o suficiente para recriá-la.

10.3.2. Processo de Backup

Quando o primeiro buckup de uma conta é criado, a ferramenta de backup executará as seguintes ações:

- Criar uma listagem de todas as pastas da conta e seus respectivos conteúdos
- Gravar no disco todos os itens encontrados

Como o primeiro passo é a criação da lista de tudo o que está contido na conta, os itens recém-criados durante o backup não vai serão visualizados e, portanto, não serão incluídos no backup. Os Itens movidos ainda estarão no backup, mas no local original em que foram encontrados.Não será realizado o backup dos itens delatados pelo comando 'Shift+delete' durante o backup, isso porque eles não podem mais ser abertos.

Quando o backup for iniciado novamete, o backup anterior será identificado e automáticamente terá início um backup complementa. Serão executadas as seguintes ações.

- Leitura do arquivo índice e criação de uma arvore do backup anterior
- Criar uma listagem de todas as pastas da conta e seus respectivos conteúdos
- Identificação dos itens cujo backup já foi realzado para que tais itens não sejam removidos da lista; esse processo é feito por pasta.
- Remoção do arquivo índice antigo
- Realização do backup dos itens listados e anexação destes ao arquivo de dados

Para iniciar o processo de backup utilize:

```
zarafa-backup -u <username>
```

ou para todos os usuários e pastas públicas:

```
zarafa-backup -a
```

To speed up the backup process multiple threads can be configured in the **`backup.cfg`**. The default option is 1 thread, so for larger environment increasing this number is recommended.

Há algumas coisas a se observar sobre o comportamento da ferramenta de backup. Quando o conteúdo da lista do arquivo índice anterior e o da atual são comparados, esse processo se faz por pasta correspondente. Isto significa que se o usuário moveu itens de uma pasta para outra, eles não serão encontrados, assim será feito o backup novamente, eles serão marcados como 'novo' na outra pasta para a qual ele foi movido.

If a message was changed by the user since the last backup, the item will have a new ‘last modification date’, and will be backed up again in its totality, since the backup would become unbearably slow if it would need to check all the properties of a message to see which properties have changed and which have not. Overwriting the old message is also problematic, because the new message may be bigger than the old one, and it will not fit into the old space of the message.

Then when the actual backup process starts, it will first remove the old index. The index file will then be rebuild while the backup processes each message found in the list. The changed data will be placed in a new data file with an incrementing counter in its filename, keeping the old information which was still available and did not need to be stored again.

Para mais opções da **`Ferramenta de Backup Zarafa`** utilize:

```
man zarafa-backup
```

10.3.3. Processo de restauração

Para restaurar os itens da **`Ferramenta de backup Zarafa`**, use a ferramenta **`Zarafa-restore`**. Para restaurar os itens ou pastas completas, encontre a chave de restauração correspondente no arquivo **`user.index.zbk`**.

This index file isn’t humanly readable with a text editor. Instead, use the **`readable-index.pl`** Perl script, which can be found in **`/usr/share/zarafa-backup/`**. To identify items, use the folder name field or the subject to find the items needed to be restored.

```
/usr/share/zarafa-backup/readable-index.pl username.index.zbk
```

When the items are found, place the restore keys in a separated file, or give them as parameters to the **zarafa-restore** tool. If the restore key of a folder is entered, the complete folder with all its items will be restored on one level. If the sub folders of the selected folder need to be restored, add the **-r** parameter to the command. The following example restores the inbox with sub folders from **userA**. The restore key **AF000000** is found in the **userA.index.zbk** file and needs to be defined at the end of the command.

```
zarafa-restore -u userA -r -f userA.index.zbk AF000000
```

The **-f** parameter as a reference for the index file is not necessary when using an index file from the same user. For example, if using **zarafa-restore --u userA**, the **zarafa-restore** tool will automatically use the **userA.index.zbk** file when **index.zbk** is in the same directory as where the command is executed.

No próximo exemlo será utiizado um arquivo (**keys.txt**) conntendo múltiplas chaves de restauração de múltiplos itens e pastas para o usuário **userA**. Cada chave de restauração do arquivo necessita ser separada numa linha nova.

```
zarafa-restore -u userA --r --i keys.txt
```

Para realizar a restauração de uma conta de e-mail inteira, pode-se usar o seguinte script.

```
/usr/share/zarafa-backup/full-restore.sh <username>
```

Certifique-se de que o script seja executado a partir do diretório de backup. Para restaurar uma conta inteira de outro usuário, utilize:

```
/usr/share/zarafa-backup/full-restore.sh <username> <destination_username>
```

Para mais opções sobre o **zarafa-restore**, verifique a página 'man' da ferramenta.

```
man zarafa-restore
```

Apêndice A; estratégias de atualização pré-5.2x

11.1. Atualizações do banco de dados à partir de 4.1 ou 4.2

Antes que se possa iniciar o Zarafa novamente, o banco de dados deve ser atualizado. Existem vários scripts necessários, dependendo da versão partir da qual de realiza a atualização. Scripts de atualização só são necessários quando a atualização se dá a partir da versão 5.0x ou de uma versão anterior. Os scripts são os seguintes:

```
db-convert-4.1-to-4.2
```

Este script perl atualiza o banco de dados do 4.1 para o formato 4.20. Estas são mudanças a respeito de que modo os usuários são armazenados no banco de dados. Este script é necessário e deve ser executado da seguinte forma:

```
perl /usr/share/doc/zarafa/db-convert-4.1-to-4.2 \
       <dbuser> <dbpass> <dbname>
```

Substituir **<dbuser>** com o nome de usuário usado para conectar ao banco de dados. Substituir **<dbpass>** com a senha do usuário do banco de dados. Se não houver senha, digite 2"aspas simples aqui. Substituir **<dbname>** com o nome do banco de dados do Zarafa. Isso resultará em algo como:

```
perl /usr/share/doc/zarafa/db-convert-4.1-to-4.2 root '' zarafa
```

```
db-convert-4.20-to-4.21
```

Este script perl atualiza o banco de dados de 4.20 para o formato 4.21. Ele irá substituir uma chave de indexação para melhorar a velocidade do banco de dados. Este script é altamente recomendável, e deve ser executado como explicado para o script **db-convert-4.1-to-4.2**.

Depending on the size of the database and the speed of the system, this script might take a while, but it will probably complete within 10 to 30 minutes.

```
db-convert-4.20-to-innodb.sql
```

Este script SQL converte o que foi convertido para o formato de banco de dados 4,20 InnoDB. Instalações que iniciaram na versão 4.0 criando tabelas MyISAM. No entanto, o layout atual banco de dados SQL é otimizado para InnoDB. Portanto, convertendo o banco de dados MyISAM para InnoDB irá resultar em um aumento de velocidade enorme. Além disso, InnoDB é menos propenso a erros e tem o bloqueio de tabela menos abrangente. É altamente recomendado para converter o banco de dados InnoDB. No prompt do MySQL, importe o script:

```
mysql> source /usr/share/doc/zarafa/db-convert-4.20-to-innodb.sql
```

Dependendo do tamanho do banco de dados e da velocidade do sistema, este script vai demorar um longo tempo . Reserve até 8 horas de tempo para completar essa conversão para um banco de dados com vários gigabytes de dados. Se as configurações de memória MySQL forem otimizadas antes desse script ser iniciado, ele vai correr muito mais rápido.

```
db-convert-4.2x-to-5.00
```

Este script perl atualiza o banco de dados do formato das versões 4.2x para o formato das versão 5.0. Este script calcula e acrescenta uma coluna de caixas para a tabela de propriedades. Isso seleciona a tabela no disco aumentando o rendimento dos dados. Para executar esse script, faça como descrito no caso do script **db-convert-4.1-to-4.2**.

Dependendo do tamanho do banco de dados e da velocidade do sistema, este script pode demorar um pouco, mas provavelmente será concluído dentro de 10 a 30 minutos em uma máquina rápida.

Nota

É aconselhável começar esse script em tela cheia, para que este script possa continuar em segundo plano.

11.2. Atualização da versão 5.0 para as versões 5.1x e posteriores

O servidor Zarafa 5.10 pode atualizar o próprio banco de dados. Ele pode fazer isso a partir do banco de dados que é necesário para a versão 5.10. Quando se atualiza o Zarafa das versões 4.x para a versão 5.0 ou posterior, é necessário antes de tudo atualizar o banco dados (com o script descrito abaixo) para o formato 5.0. Assim, o servidor 5.1 pode . Em seguida o servidor 5.10 pode ser iniciado e poderá finalizar a atualização automaticamente da versão 5.0 para a 5.10.

Versões posteriores do Zarafa sempre podem realizar atualizações a partir de um formato de banco de dados 5.0 ou mais recente.

11.3. Mudanças importantes a partir das versões 4.x e 5.x

Uma opção de configuração no **server.cfg** foi alterada desde a versão 4.20. A opção server_name foi renomeada para server_bind. Um arquivo de configuração com erros de digitação nos nomes das opções ou opções não existentes e que, consequentemente, prestariam um serviço inoperante, não serão inicializadas. Todos os erros encontrados no arquivo de configuração serão impressos.

Na versão 5.0 do Zarafa algumas opções não utilizadas foram removidas da configuração do servidor. O suporte SQLite foi removido, de modo que a opção **internal_path** também foi removida. Se esta opção estiver no arquivo **server.cfg**, por favor, remova esta linha antes de iniciar o processo de Zarafa-server.

Opções não definidas em um arquivo de configuração manterão seu valor padrão. Valores padrões podem ser encontrados exemplo de arquivo de configuração encontrado em **/ usr / share / doc / Zarafa / example-config**. Outra alternativa é ler a página de manual específico para o serviço:

```
man zarafa-<service>.cfg
```

Os serviços Zarafa não realizava daemonise em versões antes da 5.0. No entanto, as versões 5.0 e mais recentes fazem daemonise, executando em segundo plano. Para reverter esse comportamento, use o **-F** switch de um serviço para mantê-lo funcionando em primeiro plano.

Outras alterações de configuração dizem respeito ao gateway. Os padrões para o **ssl_private_file_key** e **ssl_certificate_file** foram alterados. O diretório padrão agora é **/ etc / Zarafa / gateway /**, para distingui-lo do serviço de arquivos ssl.

Apêndice B; descrição dos atributos do LDAP

Este apêndice descreve todos os atributos LDAP disponíveis e acessíveis no Zarafa. O esquema Zarafa está disponível no kit de ferramentas de integração no Active Directory e no diretório **`/usr/share/doc/zarafa`**.

Tenha em mente que as configuração de arquivos LDAP no Zarafa são muito flexíveis, de maneira que estes atributos não são utilizados em todos os casos.

zarafaQuotaOverride

Este atributo é usado para subscrever a cota padrão que é configurado no **`/etc/zarafa/server.cfg`**. Este atributo sempre precisa ser habilitado para o uso da cota customizada.

OID	1.3.6.1.4.1.26278.1.1.1.1
Sintaxe	Total
Valor únivo ou Multi-valor	Valor único

zarafaQuotaWarn

Este atributo contém o nível de alerta de quota em MB.

OID	1.3.6.1.4.1.26278.1.1.1.2
Sintaxe	Total
Valor únivo ou Multi-valor	Valor único

zarafaQuotaSoft

Este atributo contém o nível de quota flexível em MB.

OID	1.3.6.1.4.1.26278.1.1.1.3
Sintaxe	Total
Valor únivo ou Multi-valor	Valor único

zarafaQuotaHard

Este atributo contém o nível da quota rígida em MB.

OID	1.3.6.1.4.1.26278.1.1.1.4
Sintaxe	Total
Valor únivo ou Multi-valor	Valor único

zarafaUserDefaultQuotaOverride

Este atributo vai subscrever o valor da cota padrão para os sistema para todos os usuários de uma empresa.

OID	1.3.6.1.4.1.26278.1.1.1.5
Sintaxe	Total

Valor únivo ou Multi-valor	Valor único

zarafaUserDefaultQuotaWarn

Este atributo contém o nível de alerta da cota em MB para todos os usuários de uma empresa.

OID	1.3.6.1.4.1.26278.1.1.1.6
Sintaxe	Total
Valor únivo ou Multi-valor	Valor único

zarafaUserDefaultQuotaSoft

Este atributo contém o nível de cota "soft" em MB para todos os usuários da empresa.

OID	1.3.6.1.4.1.26278.1.1.1.7
Sintaxe	Total
Valor únivo ou Multi-valor	Valor único

zarafaUserDefaultQuotaHard

Este atributo contem o nível da cota "hard" em MB para todos os usuários da empresa.

OID	1.3.6.1.4.1.26278.1.1.1.8
Sintaxe	Total
Valor únivo ou Multi-valor	Valor único

zarafaAdmin

Este atributo tornará do usuário um administrador do Zarafa.

OID	1.3.6.1.4.1.26278.1.1.2.1
Sintaxe	Total
Valor únivo ou Multi-valor	Valor único

zarafaSharedStoreOnly

Este atributo irá configurar uma caixa de correio como uma conta comum. Você não poderá efetuar login em contas compartilhadas.

OID	1.3.6.1.4.1.26278.1.1.2.2
Sintaxe	Total
Valor únivo ou Multi-valor	Valor único

zarafaAccount

Este atributo pode ser utilizado nos filtros de pesquisa LDAP para filtrar usuários grupos.

OID	1.3.6.1.4.1.26278.1.1.2.3
Sintaxe	Total
Valor únivo ou Multi-valor	Valor único

zarafaSendAsPrivilege

Este atributo incluirá usuários ou grupos que devem ter permissões de "enviar como" para o usuário, para quem este atributo é adicionado.

OID	1.3.6.1.4.1.26278.1.1.2.4
Sintaxe	DN ou DirectoryString
Valor únivo ou Multi-valor	Multi-valor

zarafaMrAccept

Este atributo irá configurar aceitação automática de solicitações de reunião. Este atributo **não** é utilizado nas versões do Zarafa atual.

OID	1.3.6.1.4.1.26278.1.1.2.5
Sintaxe	Total
Valor únivo ou Multi-valor	Valor único

zarafaMrDeclineConflict

Com este atributo não irá aceitar outro agendamento caso o calendário já contem um agendamento. Este atributo **não** é usado em versões atuais do Zarafa.

OID	1.3.6.1.4.1.26278.1.1.2.6
Sintaxe	Total
Valor únivo ou Multi-valor	Valor único

zarafaMrDeclineRecurring

Este atributo irá diminuir as solicitações de reunião quando estas forem definidas como recorrentes. Este atributo **não** é utilizado nas versões atuais do Zarafa

OID	1.3.6.1.4.1.26278.1.1.2.7
Sintaxe	Total
Valor únivo ou Multi-valor	Valor único

zarafaId

Este atributo pode ser usado como um unique-id genérico, por exemplo para usuários e grupos. Este atributo **não** é usado pelo Zarafa por padrão.

OID	1.3.6.1.4.1.26278.1.1.2.8
Sintaxe	Total
Valor únivo ou Multi-valor	Valor único

zarafaResourceType

Este atributo irá configurar o tipo de recurso de uma conta comum. As opções disponíveis são **`Room`** ou "Equipment"

OID	1.3.6.1.4.1.26278.1.1.2.9

Sintaxe	DirectoryString
Valor únivo ou Multi-valor	Valor único

zarafaResourceCapacity

Este atributo indica a capacidade da sala ou de equipamentos disponíveis.

OID	1.3.6.1.4.1.26278.1.1.2.10
Sintaxe	Total
Valor únivo ou Multi-valor	Valor único

zarafaHidden

This attribute will hide the object in the Global Address Book. This will also hide the object for administrator users.

OID	1.3.6.1.4.1.26278.1.1.2.11
Sintaxe	Total
Valor únivo ou Multi-valor	Valor único

zarafaEnabledFeatures

Controla quais recursos são explicitamente habilitados para um usuário, e substitui quaisquer recursos desativados nas definições do servidor disabled_features.

OID	1.3.6.1.4.1.26278.1.1.2.13
Sintaxe	Corda
Valor únivo ou Multi-valor	Multi-valor

zarafaDisabledFeatures

Controla quais recursos são explicitamente desativados para determinado usuário.

OID	1.3.6.1.4.1.26278.1.1.2.14
Sintaxe	Corda
Valor únivo ou Multi-valor	Multi-valor

zarafaAliases

Este atributo conterá todos os outros endereços eletrônicos e apelidos para detreminado usuário.

OID	1.3.6.1.4.1.26278.1.1.3.1
Sintaxe	DirectoryString
Valor únivo ou Multi-valor	Multi-valor

zarafaUserServer

Este atributo será o homeserver de um usuário quando em execução de modo multi-servidor.

OID	1.3.6.1.4.1.26278.1.1.4.1

Sintaxe	DirectoryString
Valor únivo ou Multi-valor	Valor único

zarafaSecurityGroup

Este atributo especifique se um grupo tem privilégios de segurança. Se este atributo e setado em **`0`** o grupo será interpretado como uma lista de distribuição.

OID	1.3.6.1.4.1.26278.1.2.2.1
Sintaxe	Total
Valor únivo ou Multi-valor	Valor único

zarafaViewPrivilege

Este atributo conterá empresas com privilégios de vista em relação à empresa selecionada.

OID	1.3.6.1.4.1.26278.1.3.2.4
Sintaxe	DirectoryString
Valor únivo ou Multi-valor	Multi-valor

zarafaAdminPrivilege

Este atributo conterá usuários de diferentes empresas administradoras da empresa selecionada.

OID	1.3.6.1.4.1.26278.1.3.2.5
Sintaxe	DirectoryString
Valor únivo ou Multi-valor	Multi-valor

zarafaSystemAdmin

This attribute will specify the users who are system administrators for this company.

OID	1.3.6.1.4.1.26278.1.3.2.6
Sintaxe	DirectoryString
Valor únivo ou Multi-valor	Multi-valor

zarafaQuotaUserWarningRecipients

Este atributo conterá os usuários que receberão um email notificação quando um usuário excede sua cota.

OID	1.3.6.1.4.1.26278.1.3.1.5
Sintaxe	DirectoryString
Valor únivo ou Multi-valor	Multi-valor

zarafaQuotaCompanyWarningRecipients

Este atributo irá conter o endereço de e-mail que irá receber um email de notificação quando uma empresa ultrapassa a sua quota.

OID	1.3.6.1.4.1.26278.1.3.1.6

Sintaxe	DirectoryString
Valor únivo ou Multi-valor	Multi-valor

zarafaCompanyServer

Este atributo irá conter o servidor padrão de uma empresa de quando esta estiver trabalhando no modo multi-servidor.

OID	1.3.6.1.4.1.26278.1.3.4.1
Sintaxe	DirectoryString
Valor únivo ou Multi-valor	Valor único

zarafaHttpPort

Este atributo conterá a porta para as conexões HTTP quando a empresa estiver trabalhando no modo multi-servidor.

OID	1.3.6.1.4.1.26278.1.4.4.1
Sintaxe	Total
Valor únivo ou Multi-valor	Valor único

zarafaSslPort

Este atributo conterá a porta para as conexões HTTPS quando a empresa estiver trabalhando no modo multi-servidor.

OID	1.3.6.1.4.1.26278.1.4.4.2
Sintaxe	Total
Valor únivo ou Multi-valor	Valor único

zarafaFilePath

Este atributo conterá o UNIX socket ou o pipe escolhido para o servidor quando a empresa estiver trabalhando no modo multi-servidor.

OID	1.3.6.1.4.1.26278.1.4.4.3
Sintaxe	DirectoryString
Valor únivo ou Multi-valor	Valor único

zarafaContainsPublic

Este atributo permitirá o armazenamento público para um nó de multi-servidor específico. Certifique-se que somente um nó tenha ativado este atributo.

OID	1.3.6.1.4.1.26278.1.4.4.4
Sintaxe	Total
Valor únivo ou Multi-valor	Valor único

zarafaFilter

Este atributo conterá o filtro LDAP para solicitar uma lista de endereços ou dinâmica de grupo.

OID	1.3.6.1.4.1.26278.1.5.5.1
Sintaxe	DirectoryString
Valor únivo ou Multi-valor	Valor único

zarafaBase

Este atributo irá conter a base de pesquisa LDAP para solicitar uma lista de endereços ou dinâmica de grupo.

OID	1.3.6.1.4.1.26278.1.5.5.2
Sintaxe	DirectoryString
Valor únivo ou Multi-valor	Valor único

Appendix C: Example LDIF

The LDIF below shows an example of LDAP configuration for a single tenant setup.

```
dn: dc=example,dc=com
objectClass: dcObject
objectClass: organization
dc: zarafa
description: My LDAP Root
o: example.com

dn: cn=Manager,dc=example,dc=com
objectClass: simpleSecurityObject
objectClass: organizationalRole
cn: admin
cn: Manager
userPassword: secret
description: LDAP administrator

dn: ou=Addresslists,dc=example,dc=com
objectClass: organizationalUnit
objectClass: top
ou: Addresslists

dn: ou=People,dc=example,dc=com
objectClass: organizationalUnit
objectClass: top
ou: People

dn: ou=Groups,dc=example,dc=com
objectClass: organizationalUnit
objectClass: top
ou: Groups

dn: ou=Contacts,dc=example,dc=com
objectClass: organizationalUnit
objectClass: top
ou: Contacts

dn: cn=Mary Poppins,ou=Contacts,dc=example,dc=com
objectClass: inetOrgPerson
objectClass: top
objectClass: zarafa-contact
uidNumber: 1001
sn: Poppins
cn: Mary Poppins
mail: mary@poppins.org

dn: uid=john,ou=People,dc=example,dc=com
objectClass: posixAccount
objectClass: top
objectClass: zarafa-user
objectClass: inetOrgPerson
gidNumber: 1000
cn: John Doe
homeDirectory: /home/john
mail: john@example.com
uidNumber: 1000
zarafaAliases: j.doe@example.com
zarafaUserServer: node1
uid: john
zarafaAccount: 1
zarafaAdmin: 0
sn: Doe
userPassword: john
```

```
zarafaQuotaOverride: 1
zarafaEnabledFeatures: imap
zarafaDisabledFeatures: pop3
zarafaQuotaWarn: 1000000000
zarafaQuotaSoft: 1100000000
zarafaQuotaHard: 1200000000

dn: cn=Example addresslist,ou=Addresslists,dc=example,dc=com
objectClass: zarafa-addresslist
objectClass: top
cn: Example addresslist
zarafaFilter: (mail=*@example.com)

dn: cn=Example security group,ou=Groups,dc=example,dc=com
objectClass: posixGroup
objectClass: top
objectClass: zarafa-group
zarafaHidden: 0
cn: Example security group
gidNumber: 1000
memberUid: john
zarafaAccount: 1
description: Example security group
zarafaSecurityGroup: 1

dn: cn=Example distribution group,ou=Groups,dc=example,dc=com
objectClass: posixGroup
objectClass: top
objectClass: zarafa-group
zarafaHidden: 0
cn: Example distribution group
memberUid: john
zarafaAccount: 1
gidNumber: 1001
description: Example distribution group
zarafaSecurityGroup: 0
```