Sun Java™ System

Messaging Server 6
配備計画ガイド

2004Q2

Sun Microsystems, Inc.
4150 Network Circle
Santa Clara, CA 95054
U.S.A.

Part No: 817-7102

目 次

図目次

表目次

はじめに

『Sun Java System Messaging Server 配備計画ガイド』には、Sun™ Java System Messaging Server 6 2004Q2 と付属のソフトウェアコンポーネントを配備するのに必要な情報があります。Messaging Server は、強力で柔軟性に富んだ複数のプラットフォームをサポートするソリューションを備えており、オープンなインターネット標準を使用して、企業の電子メールとあらゆるサイズのメッセージングホストのニーズを満たします。

この章には、以下の節があります。

対象読者

このマニュアルは、管理するサイトで、Messaging Server の配備に対し、責任ある立場の方を対象としています。

注	Netscape Messaging Server または Sun Internet Mail Server 製品から Sun Java System Messaging Server に、既存のメールボックスとメッセージキューを直接移行することはできません。 Netscape Messaging Server または Sun Internet Mail Server から Sun ONE Messaging Server 5.2 への移行については、『Sun ONE Messaging Server 5.2 Migration Guide』を参照してください。Messaging Server 5.2 から Sun Java System Messaging Server 6 へのアップグレードについては、『Sun Java System Messaging Server 管理ガイド』の指示に従ってください。

お読みになる前に

このマニュアルは、Messaging Server の配備計画に関する責任者を対象としており、以下のことに関する一般的な知識を持っていることを前提としています。

- インターネットおよび WWW (ワールドワイドウェブ)
- IMAP、POP、SMTP、HTTP、および LDAP プロトコル
- Sun Java™ Enterprise System
- Sun Java™ System 管理サーバー
- Sun Java™ System Identity Server
- Sun Java™ System Web Server
- Sun Java™ System Directory Server
- Sun Java™ System Console
- Solaris™ システム管理およびネットワーキング

表記

以下の表は、このマニュアルで使用されている書体表記規則です。

表 1　書体表記規則

書体	意味	例
AaBbCc123 (固定幅)	API および言語要素、HTML タグ、Web サイトの URL、コマンド名、ファイル名、ディレクトリパス、画面上のコンピュータ出力、サンプルコード	.login ファイルを編集する ls -a を使用してすべてのファイルを一覧する % 受信メールがあります
AaBbCc123 (固定幅太字)	画面上のコンピュータ出力と対比した入力内容	% **su**
AaBbCc123 (斜体)	実際の名前または値で置き換えられるコマンド行変数	これらは、*class* オプションと呼ばれる ファイルは、*ms_svr_base*/sbin ディレクトリにある

以下の表は、このマニュアルで使用されているプレースホルダーの表記規則です。

表 2　プレースホルダーの表記法

項目	意味	例
product_base	製品がインストールされているディレクトリのプレースホルダー	*ms_svr_base*/bin ディレクトリは、下記のとおり /opt/SUNWmsgsr/

以下の表は、このマニュアルで使用されている記号の表記規則です。

表 3　記号の表記法

記号	意味	表記法	例
[ ]	任意のコマンドオプションがある	O[*n*]	O4, O
{ }	コマンドオプションとしての選択肢のセットを含む	d{y|n}	dy
|	コマンドオプションの選択肢を区切る		

表 3　記号の表記法 (続き)

記号	意味	表記法	例
+	グラフィカルユーザーインタフェースで使用されるキーボードショートカットに、同時に押すキーを追加する		Ctrl+A
-	グラフィカルユーザーインタフェースで使用されるキーボードショートカットに、連続して押すキーを追加する		Esc-S
>	グラフィカルユーザーインタフェースで、メニューの選択を示す		ファイル > 新規 ファイル > 新規 > テンプレート

Web 上のリソース

このマニュアルのほかに、Messaging Server には、管理者用の補足情報およびエンドユーザーや開発者用のマニュアルもあります。次の URL を使用すると、Messaging Server のすべてのマニュアルを参照できます。

http://docs.sun.com/db/prod/entsys?l=ja

Messaging Server 製品のセットには、Sun Java System Console、Sun Java System Directory Server、および Sun Java System 管理サーバーなどのコンポーネント製品も含まれています。これらの製品およびその他の製品のマニュアルは、次の URL で参照できます。

http://docs.sun.com/db/prod/entsys?l=ja

このマニュアルには、関連する詳細情報が提供されているサードパーティーの URL が含まれています。

注　Sun は、このマニュアルに記載されているサードパーティの Web サイトの可用性について責任を負いません。Sun は、サードパーティのサイトやリソース上またはこれらを通じて利用できるコンテンツ、広告、製品、その他の素材について保証せず、いかなる責任も負いません。こうしたサイトやリソース上またはこれらを通じて利用できるコンテンツ、製品、またはサービスを利用または信用したことに伴って発生した (あるいは発生したと主張される) いかなる損害や損失についても、Sun は一切責任を負いません。

特定の Messaging Server 製品に関する技術サポートについては、Sun Java System Messaging Server Software Forum も参照してください。以下の URL をご利用ください。

http://swforum.sun.com/jive/forum.jsp?forum=15

問題の報告方法

Messaging Server に関して問題が発生した場合は、以下の方法を使用してご購入先のカスタマサポートに連絡してください。

- Sun Software Support のオンラインサービス

  http://www.sun.com/service/sunone/software

  このサイトには、保守プログラムとサポートの連絡先のほかに、Knowledge Base、Online Support Center、および ProductTracker へのリンクがあります。

- 保守契約に関連する電話番号

問題の解決をサポートするために、カスタマサポートにご連絡の際には以下の情報の提供をお願いいたします。

- 問題が発生した状況と運用への影響などの、問題の詳細
- マシンのタイプ、パッチを含むオペレーティングシステムのバージョンおよび製品のバージョン、および問題に影響を与えている可能性のある他のソフトウェア
- 問題が再現したときに行っていた詳細な手順
- エラーログまたはコアダンプ

コメントをお寄せください

Sun は、マニュアルをよりよいものにしていくために、皆様からのコメントやご意見をお待ちしています。Web ベースのフォームを使用して、Sun へのフィードバックをお願いします。

http://www.sun.com/hwdocs/feedback/

マニュアルの完全なタイトル名と該当するフィールドのパート番号をお知らせください。パート番号は、マニュアルのタイトルページまたは表紙にある 7 桁または 9 桁の番号です。たとえば、この『Sun Java System Messaging Server 配備計画ガイド』のパート番号は 817-7102 です。

第 1 章

Messaging Server ソフトウェアの紹介

Sun Java™ System Messaging Server 6 2004Q2 は、企業とサービスプロバイダの両方で要求される大容量で信頼性の高いメッセージング処理のために設計された、強力な標準ベースのインターネットメッセージングサーバーです。サーバーはモジュール化された、個別に構成可能な複数のコンポーネントから成り立っています。これらのコンポーネントは、さまざまな標準ベースの電子メールプロトコルをサポートしています。

Messaging Server は、ユーザー、グループ、およびドメインについての情報を格納するために一元化された LDAP データベースを使用します。サーバー設定についてのいくつかの情報は LDAP データベースに格納されます。また、ローカル設定ファイルに格納される情報もあります。

Messaging Server 製品群には、ユーザーのプロビジョニングやサーバーの構成をサポートするツールが含まれています。

この章には、以下の節があります。

標準プロトコルのサポート

Messaging Server は、電子メッセージングに関連するほとんどの国内規格、国際規格、および業界規格をサポートしています。完全なリストは、『Sun Java System Messaging Server Administration Reference』の付録 A を参照してください。

http://docs.sun.com/doc/817-6267

ホストされているドメインのサポート

Messaging Server は、ISP にアウトソースされた電子メールドメインのようなホストされているドメインを完全にサポートしています。つまり、ISP は組織の電子メールサービスをリモートで操作および管理することにより組織をホスティングする電子メールドメインを提供します。ホストしているドメインは、ほかのホストしているドメインと同じ Messaging Server ホストを共有することができます。初期の LDAP ベースの電子メールシステムでは、1 つのドメインが 1 つまたは複数の電子メールサーバーホストによってサポートされていました。Messaging Server では、複数のドメインを単一のサーバーでホストできます。ホストされている各ドメインには、そのドメインのユーザーとグループのコンテナを指し、さまざまなドメイン固有のデフォルト設定を提供する LDAP エントリがあります。

ユーザーのプロビジョニングのサポート

Messaging Server は、ユーザー、グループ、およびドメインについての情報を格納するために一元化された LDAP データベースを使用します。現在、Messaging Server は Sun Java™ System Schema 1 (Schema 1) と Sun Java™ System Schema 2 (Schema 2) の 2 つのスキーマオプションをサポートしています。プロビジョニングオプションは、選択されたスキーマにより異なります。詳細は、第 11 章「インストール前の考慮事項と手順」を参照してください。

『Sun Java System Communications Services User Management Utility Administration Guide』で説明されているように、現在、スキーマ 2 の Messaging Server プロビジョニングは、`commadmin` ユーティリティを使用した場合にだけ実行できます。

http://docs.sun.com/doc/817-5703

スキーマ 1 は、メッセージング用 iPlanet™ Delegated Administrator 製品によってサポートされています。メッセージング用 iPlanet Delegated Administrator 製品には、組織内のユーザー、グループ、およびドメインを管理するために、グラフィカルユーザーインタフェースとコマンドラインユーティリティが用意されています。スキーマ 1 におけるユーザー、グループ、およびドメイン管理については、以前のリリースのソフトウェアに関する以下のマニュアルを参照することもできます。

- 『iPlanet Messaging Server Provisioning Guide』- LDAP を使ってドメイン、ユーザー、グループ、または管理者のエントリを作成する方法を説明しています。

  http://docs.sun.com/doc/816-6018-10

- 『iPlanet Messaging and Collaboration Schema Reference Manual』- Messaging Server のスキーマ 1 について説明しています。

  http://docs.sun.com/doc/816-6021-10

- 『iPlanet Messaging Server Reference Manual』- ユーザー、グループ、およびドメインを管理するための iPlanet Delegated Administrator コマンドラインユーティリティについて説明しています。

  http://docs.sun.com/doc/816-6020-10

- iPlanet Delegated Administrator オンラインヘルプ

注

Sun Java System Identity Server コンソールには、Identity Server Service を使用した最小インストールの Messaging Server と Calendar Server LDAP ユーザーエントリプロビジョニングが用意されています。インタフェースには入力を確認する機能がないため、電子メールを受け取ることができないユーザーエントリや動作しないユーザーエントリが、エラーが報告されることなく作成されてしまいます。そのため、このインタフェースはデモの目的でだけ使用します。

Messaging Server ユーザープロビジョニングのメカニズムには、『Sun Java System Communications Services User Management Utility Administration Guide』で説明されている `commadmin` インタフェースを推奨します。

統一されたメッセージングのサポート

Messaging Server は完全な統一されたメッセージングソリューションの基盤となります。統一されたメッセージングとは、電子メール、ボイスメール、FAX、およびその他の通信形態に関して単一のメッセージストアを使用するという概念です。

Web メールのサポート

Messaging Server は、エンドユーザーが HTTP を使用したインターネットに接続されているコンピュータシステム上で、自分のメールボックスにアクセスすることができる、Web で利用可能な電子メールプログラム Messenger Express を含みます。Messenger Express クライアントは、Messaging Server の一部である特殊な Web サーバーにメールを送信します。HTTP サービスは、ルーティングまたは配信のために、そのメッセージをローカルの MTA またはリモートの MTA に送信します。

注 Sun Java™ System Communications Express は Messenger Express クライアントもサポートしています。詳細は、Communications Express のマニュアルを参照してください。

http://docs.sun.com/db/prod/entsys?l=ja

強力なセキュリティとアクセス制御

Messaging Server には、次のセキュリティとアクセス制御の機能があります。

- POP、IMAP、HTTP、または SMTP へのパスワードによるログインおよび証明書に基づくログインのサポート
- 標準セキュリティプロトコルのサポート : TLS (Transport Layer Security)、SSL (Secure Socket Layer)、および SASL (Simple Authentication and Security Layer)
- アクセス制御命令を使用した委任管理 (スキーマ 1 のみ)
- POP、IMAP、SMTP および HTTP へのクライアントアクセスのフィルタリング
- システム全体、ユーザーごと、およびサーバー側のルールによる不特定多数宛のメールのフィルタリング

使いやすいユーザーインタフェース

Messaging Server はモジュール化された、個別に構成可能な複数のコンポーネントから成ります。これらのコンポーネントは、電子メールの転送とアクセスプロトコルをサポートしています。

Messaging Server には、MTA (Message Transfer Agent) を設定するために、コマンドラインユーティリティと設定ファイルの完全なセットが用意されています。このセットはサーバーとコマンド行ユーティリティにローカルに格納されています。また、メッセージストアおよびメッセージアクセスサービスを設定するために、コンソールグラフィカルユーザーインタフェースとコマンドラインユーティリティの完全なセットが用意されています。

第 2 章

要件の分析

Messaging Server の配備を計画する場合は、最初に組織の業務および技術の要件を分析する必要があります。この章では、要件を収集し評価する方法と、それに基づいて構築する Messaging Server アーキテクチャーを決定する方法を説明します。

この章には、以下の節があります。

配備目標の確認

Messaging Server ソフトウェアまたはハードウェアを購入したり配備したりする前に、配備の目標を確認しておく必要があります。組織内のさまざまなソースから、配備の要件があがってきます。多くの場合、要件はあいまいな言葉で表現されますが、それを特定の目標に向けた明確な定義に変える必要があります。

配備を計画する前に検討の必要がある要件には、以下のものがあります。

- ビジネスの要件
- 技術の要件
- 財務の要件
- サービスレベル契約 (SLA)

要件分析の結果は、明確で簡潔な言葉で定義し、配備による成果を評価できる目標としてまとめる必要があります。プロジェクト関係者からの同意を得た明確な目標がなければ、先に進んでも成功するのは困難です。

業務の要件

ビジネスの目標は、配備の決定に大きく影響します。具体的には、ユーザーの行動、サイトの配布、配備に影響を与える潜在的な政治的要因について理解しておく必要があります。これらのビジネス要件を理解していない場合は、誤った前提に基づいて的確な配備計画に悪影響をおよぼす結果を招きかねません。

運用の要件

直接的な目標を持った一連の機能上の要件として、運用要件を明確にします。通常、以下の項目が該当します。

- エンドユーザー機能
- エンドユーザー応答時間
- 可用性 / 稼働時間
- 情報の保存と保持

たとえば、「適切なエンドユーザー応答時間」という表現を評価可能な用語に変換して、関係者全員が何が「適切」で応答時間がどのように評価されるかを理解できるようにします。

カルチャーとポリティクス

配備を考える場合、企業のカルチャーとポリティクスを考慮する必要があります。需要というものは、結局はビジネス要件そのものから生み出されてくるものです。
例 :

- サイトの中には、配備されたソリューションを独自に管理する必要があるものもある。そのような需要が、プロジェクトのトレーニング費用、複雑さなどを発生させる元となる
- LDAP ディレクトリに個人情報が含まれている場合、人事部門はそのディレクトリを自己の管理下におきたいと考えるはず

技術の要件

技術の要件 (または機能の要件) は、組織のシステムニーズの詳細です。

既存の利用率パターンのサポート

既存の利用率パターンを、配備実現のための明確で評価可能な目標として定義します。以下の質問が、目標を定義する際に参考となります。

- 現在のサービスはどのように利用されているか。
- ユーザーは分類可能か (一時的なユーザー、常用ユーザー、ヘビーユーザーなど)。
- ユーザーが通常送信するメッセージのサイズはどのくらいか。
- ユーザーが通常送信する 1 日あたりまたは 1 時間あたりのメッセージ数はどのくらいか。
- ユーザーがメッセージを送信するのは、社内のどのサイトか。

サービスにアクセスするユーザーについて調査します。ユーザーはいつ既存のサービスを使うのかといった要素が、配備の要件、ひいては配備の目標を定める重要なポイントとなります。組織の今までの事例からこれらのパターンを得ることができない場合は、他の組織の事例を研究し、推測します。

利用率のきわめて高い部署では、専用のサーバーが必要になる場合もあります。一般に、ユーザーが実際のサーバーから遠く離れている場合、応答時間が長くなります。応答時間が適切であるかどうかを検討する必要があります。

サイトの分散

以下の質問を検討して、サイトの分散が配備目標に与える影響を理解します。

- サイトが、物理的にどのように分散されているか。
- サイト間の帯域幅はどれか。

  集中化方式を採用する場合は、分散化方式よりも広い帯域幅が必要です。ミッションクリティカルなサイトには、専用サーバーが必要です。

ネットワーク

以下の質問を検討することで、ネットワークの要件を理解します。

- 内部ネットワーク情報をわかりにくくしたいと考えるか。
- ネットワークサービスに冗長性を持たせたいと考えているか。
- レイヤーホストにアクセスする場合に、利用可能なデータを制限したいと考えているか。

- エンドユーザー設定を簡略化したいと考えていますか。たとえば、変更が不要なシングルメールホストにエンドユーザーを加入させるなどの方法があります。
- ネットワークの HTTP トラフィックを削減したいと考えていますか。

注 これらの質問にはいと答えた場合は、2 階層アーキテクチャを採用することが考えられます。詳細は、第 3 章「メッセージングアーキテクチャの開発」を参照してください。

既存のインフラストラクチャ

より信頼性の高い高可用性帯域幅が利用できる場合は、集中化サーバーを採用できます。

- 既存のインフラストラクチャと設備で、この配備が可能か。
- DNS サーバーは追加の負荷を処理できますか。ディレクトリサーバー、ネットワーク、ルーター、スイッチ、ファイアウォールはどうか。

サポート要員

24 時間、週に 7 日 (24 x 7) 体制のサポートは、特定のサイトでのみ提供されます。少数のサーバーによる簡単なアーキテクチャの場合は、サポートが容易です。

- 運用グループと技術サポートグループに十分な能力があり、この配備を促進できる状況にあるか。
- 運用グループと技術サポートグループは、配備期間中に増大する負荷に対処できるか。

財務の要件

財務上の制約により、配備の構築がどの程度影響を受けますか。財務上の要件は全体的な視点から明確に定義される場合が多く、配備の限界や目標が明確になります。

ハードウェア、ソフトウェア、および保守のための明確なコスト以外に、次のような他のコストがプロジェクト全体に影響を与えます。

- トレーニング
- ネットワーク帯域幅やルーターなどのサービスや設備のアップグレード
- 配備のコンセプトを検証するのに必要な人員やリソースのような配備コスト
- 配備されたソリューションを管理する人員のような運用コスト

プロジェクトの要件に関連する数多くの要素を注意深く分析することで、プロジェクトに関連する財務上の問題を回避することができます。

サービスレベル契約 (SLA)

サービスレベル契約には、稼働時間、応答時間、メッセージ配信時間、および障害回復のような領域に関連する配備を盛り込む必要があります。サービスレベル契約自体には、システムの概要、サポート組織の役割と責任、応答時間、サービスレベルの評価方法、要求の変更などの項目が網羅されています。

サービスレベル契約の範囲を決定する際には、システムの可用性に対する組織の予測が重要なポイントとなります。システムの可用性は、システム稼働時間に対するパーセンテージで表されます。システムの可用性を表す公式は次のとおりです。

```
可用性 = 稼働時間 / ( 稼働時間 + 停止時間 ) * 100
```

たとえば、サービスレベル契約で稼働時間が 99.99 パーセントと規定されている場合、1 か月に許されるシステムが使用できない時間は、約 4 分間となります。

さらに、システムの停止時間とは、システムが使用できない時間の合計を意味します。この合計には、システム障害やネットワークの停止などの予期しない停止時間だけでなく、計画された停止時間、予防的保守、ソフトウェアのアップグレードやパッチを当てる時間なども含まれます。システムが 7x24 (週 7 日、24 時間) 稼働を前提としている場合、アーキテクチャに冗長性を持たせて計画された停止や予期しない停止に備え、高可用性を確保する必要があります。

プロジェクト目標の決定

まず、調査と分析を行って、プロジェクトの必要要件を明確にする必要があります。次に、明確で評価可能な目標を決定します。プロジェクトに直接関与しない人員でも理解可能な形で目標を設定し、プロジェクトの評価方法も明確にしておきます。

関係者による目標の容認導入後の検証によりプロジェクトを評価し、その成果を決定する必要があります。

拡張に向けた計画

現在要求されている許容量を決定するだけでなく、計画できる時間枠内で将来必要とされる能力も算出しておく必要があります。拡張のスケジュールは、通常 6 か月から 12 か月です。拡張の例外と利用率特性の変化を考慮して、拡張を検討する必要があります。

ユーザー数とメッセージの数の増加に対応して、容量計画のガイドラインを策定する必要があります。さまざまなサーバーのメッセージトラフィックの増大、ユーザー数の増加、メールボックスサイズの増大なども想定して計画を立てる必要があります。収容ユーザー数の増加に伴い、その間の利用率特性も変化します。配備目標 (そして配備設計) は、将来に向けても実現可能なように、状況に応じて対応できるものでなければなりません。

アーキテクチャが将来の拡張を容易に吸収できるよう設計しておくのが理想的です。稼働段階に入ったら、配備状態を監視して、配備ニーズがいつどのように増加しているかを認識することも重要です。

総所有コスト (TCO) の理解

総所有コスト (TCO) もまた、許容量の計画に影響を与える要素です。これには、Messaging Server の配備で選択するハードウェアが含まれます。以下の表で、数を多くした小規模なハードウェアシステム、または少数の大規模ハードウェアシステムのどちらを配備するかに関しての検討項目をまとめています。

表 2-1 総所有コスト (TCO) の検討

ハードウェアの選択	利点	欠点
数を多くした小規模なハードウェアシステム	• 小規模なハードウェアシステムは一般にコストが低い • 数を多くした小規模なハードウェアシステムは多くの拠点に配備が可能で、分散型ビジネス環境をサポートする • 数を多くした小規模なハードウェアシステムでは、サーバーが保守のため停止している場合でも、トラフィックを別のサーバーにルーティングすることでシステム保守やアップグレード、移行のための停止時間を短縮することが可能	• 小規模なハードウェアシステムでは許容量に限界があるため、より多くのシステムが必要になる ハードウェアシステムの数が増加するにつれて管理と保守のコストも上昇する • 数を多くした小規模なハードウェアシステムでは管理台数が多いため、管理の手間が増大する

表 2-1 総所有コスト (TCO) の検討 (続き)

ハードウェアの選択	利点	欠点
少数の大規模ハードウェアシステム	• 少数の大規模ハードウェアシステムでは、サーバーあたりの固定管理コストが少額となる 管理するハードウェアシステムの数が少ないため、管理コストが月次で発生する場合、それが社内からのものであっても ISP からのものであっても、少額となる • ハードウェアシステムの数が少ないということは、保守が必要なシステムの数が少ないため、保守、アップグレード、移行の作業が容易になる	• 大規模なハードウェアシステムでは、通常、導入時のコストが大きくなる • 少数のハードウェアシステムでは、保守、アップグレード、移行のための停止時間も長くなる

第 3 章

メッセージングアーキテクチャの開発

この章では、Messaging Server 配備のアーキテクチャの設計方法について説明します。

この章には、以下の節があります。

メッセージングシステムアーキテクチャの目的

すぐれたメールシステムアーキテクチャでは、電子メールが埋め込まれたサウンド、画像、ビデオファイル、HTML 形式、Java アプレット、デスクトップアプリケーションと共に迅速に配信され、将来のアップグレードへの対応とスケーラビリティを提供します。単純化すると、Messaging Server アーキテクチャは以下の機能を備える必要があります。

- 外部サイトからのメールを受信する
- メッセージが配信されるユーザーメールボックスを判断し、そこにルーティングする
- 内部ホストからのメールを受信する
- メッセージが配信される宛先システムを判断し、そこにルーティングする

電子メールシステムの中心は Messaging Server で、これはメッセージの送信と配信に使用されるコンポーネントの集合体です。Messaging Server のコンポーネントとは別に、電子メールシステムでは LDAP サーバーと DNS サーバーも必要となります。多くの企業には既存の LDAP サーバーとデータベースが存在し、それらは Messaging Server と共に利用できます。そうでない場合、Java Enterprise System が提供する LDAP サーバー (Sun Java System Directory Server) を利用することができます。DNS サーバーは、電子メールシステムを配備する前に配置しておく必要があります。

この章の後半では、効率的でスケーラブルなメッセージングシステムの設計に使われた Messaging Server のコンポーネントと Messaging Server ソフトウェアアーキテクチャについて説明します。

効率性とスケーラビリティ以外にも、いくつかの要素が Messaging Server アーキテクチャに影響を与えます。これらの要素を次に示します。

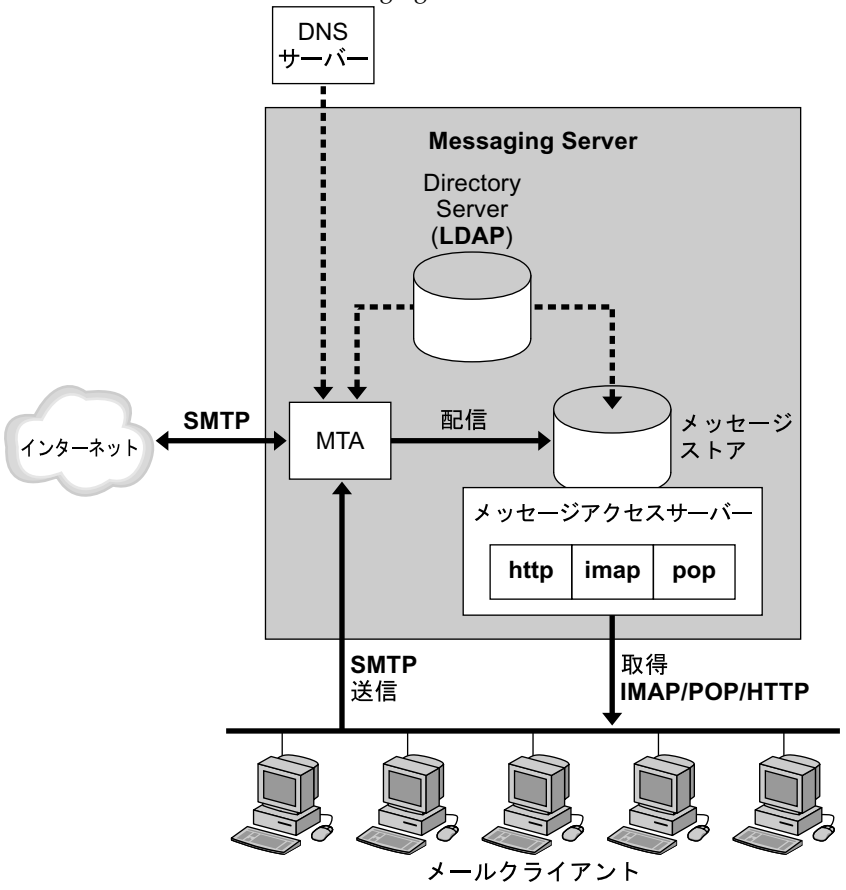
- ロードバランシング
- ファイアウォール
- 高可用性

これらについては、後の節で説明します。

Messaging Server のソフトウェアアーキテクチャ

35 ページの図 3-1 は、スタンドアロンな Messaging Server を簡略化して示しています。この特別な配備は、サイズの都合でサーバーの個別のコンポーネントが省略されていますので、そのまま使うことはお勧めできません。

図 3-1　スタンドアロンな Messaging Server の簡略化したコンポーネント表示



メッセージフロー

DNS/ディレクトリ情報フロー

太字 = メッセージングプロトコル

前の図は、以下の Messaging Server ソフトウェアコンポーネントを示します。

- **メッセージ転送エージェントまたは MTA:** SMTP プロトコルを使用して、メールメッセージの受信、ルーティング、転送、および配信を行う。MTA は電子メール配信者のように、メッセージを電子メールボックスまたは別の MTA に配信する

- **メッセージストア**：メールクライアントのメッセージの保管、取得、および操作を行うコンポーネントで構成される。メールは POP クライアント、IMAP クライアント、または HTTP クライアントにより取得される。POP クライアントは、メッセージをクライアントマシンにダウンロードして、読み出しと保管を行う。IMAP クライアントと HTTP クライアントは、サーバー上でメッセージの読み出しと保管を行う。メッセージストアは、電子メールボックスのようにユーザーのためのメールの保管と取得を行う
- **LDAP ディレクトリ**：Messaging Server のメールディレクトリ情報の保管、取得、および配信を行う。これには、ユーザーのルーティング情報、配信リスト、設定データ、および電子メールの配信とアクセスのサポートに必要なその他の情報などがある。LDAP ディレクトリは、電子メールアドレス、エイリアス、ルーティング情報、パスワード、および MTA または メッセージストアがメッセージの配信と取得を行うのに必要なその他の情報のディレクトリである
- **DNS サーバー**：ドメイン名を IP アドレスに変換する。このコンポーネントは Messaging Server をインストールする前に必要となる

簡略化した Messaging Server システムをパススルーするメッセージ

インターネットまたはローカルクライアントからの受信メッセージは、Simple Mail Transport Protocol (SMTP) を通じて MTA によって受信されます。内部アドレスの場合、すなわち Messaging Server ドメイン内の場合は、MTA はメッセージをメッセージストアに配信します。外部アドレスの場合、すなわち Messaging Server ドメイン外の場合は、MTA はメッセージをインターネット上の別の MTA にリレーします。

Messaging Server の前のバージョンのように、メールを UNIX システムでのみ使用される `/var/mail` ファイルシステムに送信することは可能ですが、ローカルメッセージは通常、より最適化された Messaging Server メッセージストアに配信されます。次に、IMAP4、POP3、または HTTP メールクライアントプログラムがメッセージを取得します。

ディレクトリサーバーは、アドレス、代替メールアドレス、メールホストのようなローカルユーザーとグループの配信情報の格納と取得を行います。MTA はメッセージを受信すると、このアドレス情報を使用してメッセージの配信先と配信方法を決定します。

メッセージを格納するだけでなく、メッセージストアはディレクトリサーバーを使用して、メールクライアントがメールにアクセスする場合のユーザーのログイン名とパスワードの検証も行います。ディレクトリには、割り当て制限、デフォルトのメッセージストアタイプなどの情報も格納されます。

メールクライアントからの送信メッセージは MTA に直接送られ、そこでインターネット上の適切なサーバーに送信されます。アドレスがローカルの場合は、MTA はメッセージをメッセージストアに送信します。

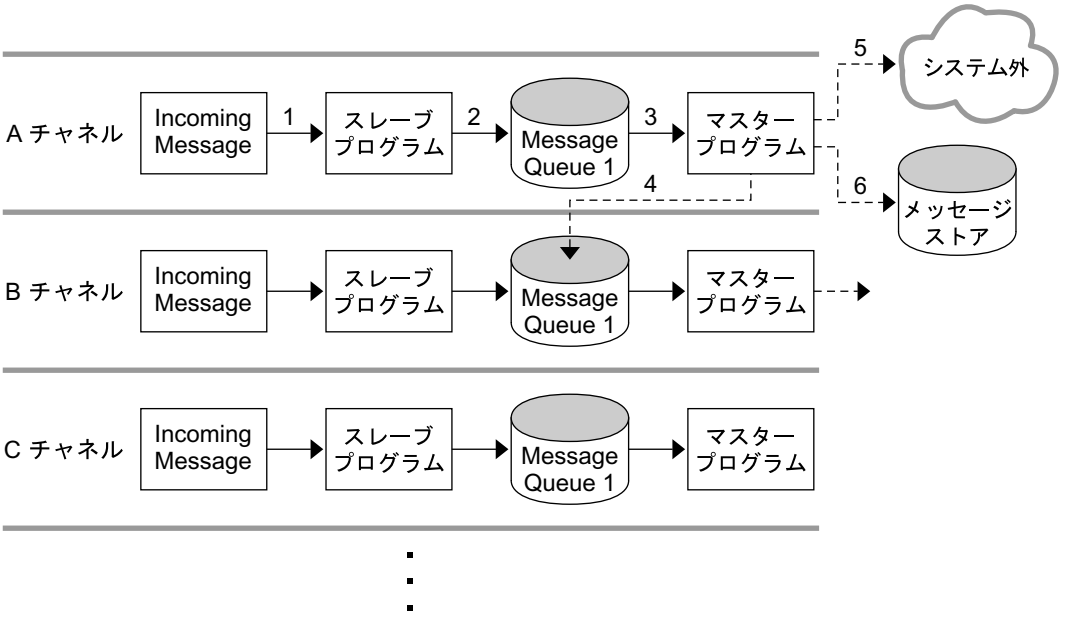
新規ユーザーとグループは、ディレクトリにユーザーとグループを追加することで作成されます。エントリは、ユーザー管理ユーティリティを使用するか、LDAP を使用してディレクトリを変更することで作成および変更できます。

Messaging Server コンポーネントは、管理サーバーコンソールによって管理されています。さらに、Messaging Server にはコマンド行インタフェースと設定ファイルも用意されています。Messaging Server ホストに接続されたマシンで、管理者が適切なアクセスを行った場合には、管理タスクを実行できます。より一般的な管理タスクとしては、メールシステムへのユーザーやグループの追加、変更、削除や、MTA、ディレクトリサーバー、およびメッセージングストアの操作の設定があります。

メッセージ転送エージェント (MTA)

MTA は、Messaging Server に宛てられたインターネットメールメッセージのルーティング、転送、および配信を行います。メールフローは、チャネルと呼ばれるインタフェースを通じて行われます。チャネルは、チャネルプログラムのペアと一連の設定情報とで構成されます。38 ページの図 3-2 にそのプロセスを示します。チャネルを個別に設定し、アドレスに基づいてメールを特定のチャネルに送ることもできます。

図 3-2　チャネルアーキテクチャ



それぞれのチャネルは、最大 2 つまでのスレーブプログラムと呼ばれるチャネルプログラムで構成されます。このプログラムは、チャネルに送信されてくるメールを処理します。マスタープログラムは、チャネルから送信されるメールを処理します。チャネルに関連付けられた 1 つ以上のインタフェースに送られるメッセージを格納するための送信メッセージキューもあります。チャネルプログラムは、以下の 2 つの機能の 1 つを実行します。

- スレーブプログラムは (1) 他のインタフェースからのメッセージを受け入れ、(2) それをメッセージキューに入れて MTA による次の処理に備えるか、システムに受け入れることのできないメッセージを拒否する
- マスタープログラムは、(3) キュー領域からのメッセージを処理し、(4) それを同じシステムのキューに入れて、別のチャネルによる処理に備える。または、(5) 他のインタフェースに送信し、送信後にキューから削除する。または、(6) そのメッセージをメッセージストアのようなシステム上の最終の宛先に配信する

チャネルの設定は、`imta.cnf` 設定テキストファイルを使用して行います。チャネル設定を通じて、さまざまなチャネルキーワードを設定してメッセージの処理方法を制御できます。チャネルキーワードは、パフォーマンスの調整とシステムのレポーティング面に影響を与えます。たとえば、複数のチャネルを定義してトラフィックをグ

ループ別または部門別に分類し、メッセージサイズを制限してトラフィックを制限し、業務のニーズに応じて配信ステータス通知ルールを定義します。診断属性もチャネル単位で設定可能です。かなりの数の設定パラメータが、チャネルベースで設定可能です。詳細は、『Sun Java System Messaging Server 管理ガイド』を参照してください。

Messaging Server には、以下のような数多くのチャネルがデフォルトで用意されています。

- **SMTP チャネル** : TCP/IP ベースのメッセージ配信と受信を有効にする。マスターチャネルとスレーブチャネルが用意される
- **LMTP チャネル** : MTA から メッセージストアへのメッセージの直接ルーティングを有効にする。これらのチャネルは、SMTP ではなく LMTP を使用してメッセージストアと通信を行う
- **パイプチャネル** : 代替メッセージ配信プログラムで使用する。メッセージをユーザーの受信ボックスに直接送るのではなく、メールソーターのようなプログラムへの配信を行う
- **ローカルチャネル** : メールを `/var/mail` に配信する。古い UNIX メールクライアントとの互換性を提供する
- **再処理チャネル** : 再送信されたメッセージの処理に便利に使われる
- **再組立チャネル** : メッセージの一部を再度組み立てて、MIME message/partial content type をサポートする元の完全なメッセージにする
- **変換チャネル** : メッセージを本文ごとに変換する。アドレスの再書き込みまたはメッセージの再フォーマットに便利に使われる

MTA の概念の詳細については、『Sun Java System Messaging Server 管理ガイド』を参照してください。

ダイレクト LDAP 検索

バージョン 5.2 以前の Messaging Server は、`dirsync` モードで実行されていました。`dirsync` モードでは、MTA が使用するユーザーとグループに関するディレクトリ情報は、ディレクトリキャッシュと総称される数多くのファイルとデータベースを通じてアクセスされていました。データそのものは LDAP ディレクトリに格納されていましたが、実際の情報はキャッシュからアクセスされていました。キャッシュ内のデータは、LDAP ディレクトリへの変更を監視してそれに伴ってファイルとデータベースを更新する `dirsync` プログラムにより更新されていました。

Messaging Server 5.2 からは、MTA を設定して LDAP サーバーから情報を直接検索できるようになりました。このダイレクト検索により、LDAP が予想する通常のクエリのタイプを使用することで、LDAP を有効に利用できます。ダイレクト検索により MTA と LDAP サーバー間の関係が、よりスケーラブルで、若干高速で、より自由に設定できるようになります。LDAP クエリの結果は設定可能なサイズと時間でプロセスにキャッシュされるため、パフォーマンスの調整が可能です。詳細は、『Sun Java System Messaging Server 管理ガイド』を参照してください。

Messaging Server 6.0 の導入に伴い、`dirsync` のサポートは終了し、削除されています。

書き換えルール

メールは、ドメイン書き換えルール または省略用の書き換えルールが適用された宛先アドレスに基づいて、チャネルにルーティングされます。書き換えルールは、アドレスを真のドメインアドレスに変換し、それに対応するチャネルを決定するのに使用されます。これらのルールは、トランスポートレイヤーとメッセージヘッダーの両方に表示されるアドレスを書き換えるのに使用されます。トランスポートレイヤーは、メッセージのエンベロープです。ルーティング情報はユーザーには見えない形で含まれていますが、実際の情報はメッセージを適切な受信者に配信するのに使用されます。

書き換えルールとチャネルのテーブルは、協力してそれぞれのアドレスの処置を決定します。書き換えプロセスの結果により、アドレスとルーティングシステム、すなわちメッセージが送信されるシステム (チャネル) が書き換えられます。ネットワークのトポロジ次第で、ルーティングシステムはメッセージが宛先までにたどるパスの最初のステップである場合もあれば、最終の宛先システムである場合もあります。

書き換えプロセスが終了すると、`imta.cnf` ファイルのチャネル部分に対してルーティングシステムの検索が行われます。それぞれのチャネルには、チャネルに関連付けられた 1 つ以上のホスト名があります。ルーティングシステム名がそれぞれのホスト名と比較されて、メッセージがどのチャネルのキューに入れられるかが決定されます。次に簡単な書き換えルールを示します。

```
example.com      $U%example.com@tcp_siroe-daemon
```

このルールは、ドメイン `example.com` のアドレスにだけ一致します。一致したアドレスは、以下に示すテンプレート `$U%$D` を使用して、に書き換えることができます。

$U	アドレスのユーザーの部分またはアドレスの左側 (@ の前) を示す
%	@ 符号を示す
$D	アドレスのドメインの部分またはアドレスの右側 (@ の後ろ) を示す

このように、wallaby@thor.example.com の形式のメッセージが wallaby@example.com に書き換えられ、tcp_siroe-daemon と呼ばれるチャネルに送信されます。

書き換えルールは、マッピングテーブル、LDAP ディレクトリ検索、およびデータベース参照に基づいて、高度な置換を行うこともできます。暗号のようなわかりにくいものになる場合もありますが、書き換えルールが低レベルで動作し、処理サイクルへの直接のオーバーヘッドがほとんどない点が便利です。書き換えルールの詳細と書き換えプロセスで利用できる機能については、『Sun Java System Messaging Server 管理ガイド』を参照してください。

ジョブコントローラ

ジョブコントローラは、マスター、ジョブコントローラ、送信側の制御と、チャネルプログラムのキューからの削除を行います。ジョブコントローラは、メッセージキューを制御し、実際のメッセージ配信を行うプログラムを実行するプログラムです。ジョブコントローラは、マルチスレッドプロセスとして実行され、Messaging Server システムで常に実行されている数少ないプロセスの 1 つです。チャネル処理ジョブ自体は、ジョブコントローラにより作成されますが、一時的なジョブで、実行する作業がない場合は存在しなくなります。

ジョブコントローラを設定して、チャネル処理プログラムのインスタンスが少なくとも 1 つ常駐するかどうかを決めることができます。多くの場合、すぐに実行する作業がない場合でも、ジョブコントローラは少なくとも 1 つのサービスプログラムのインスタンスが常に存在するように設定されます。それ以外の場合は、行う作業がなくなるまで何らかの作業の実行を継続する期間を設定するインスタンスがあります。

スレーブチャネルは、外部の誘発要因に応答し、ジョブコントローラに新しく作成されたメッセージファイルを通知します。ジョブコントローラは、この情報を内部データ構造に入力し、必要に応じてそのメッセージを処理するマスターチャネルジョブを作成します。ジョブコントローラで、現存しているチャネルジョブが新しく作成されたメッセージファイルを処理できるように設定されている場合は、このジョブを作成する必要はありません。マスターチャネルジョブは、ジョブが開始されると、ジョブコントローラからメッセージ割り当てを取得します。メッセージの処理を終了すると、マスターチャネルはその処理のステータスに応じてジョブコントローラを更新します。そのステータスは、メッセージが正常にキューから削除されたか、メッセージの再配信スケジュールが組まれたかのいずれかになります。ジョブコントローラは、メッセージの優先度と失敗した配信に関する情報を維持し、チャネルジョブに優先的なスケジュールを許可します。ジョブコントローラは、各ジョブの状態の追跡も行います。ジョブの状態は、アイドル、アイドルの時間、ジョブがビジーであるかどうかです。状態の追跡により、ジョブコントローラはチャネルジョブの最適なプールを維持できます。

Local Mail Transfer Protocol (LMTP)

Sun ONE Messaging Server 6.0 リリースでは、複数階層配備におけるメッセージストアに配信を行う LMTP 設定が可能になりました。受信リレーとバックエンドメッセージストアが使用されるこのような環境では、アドレス拡張、自動返信や転送などの配信方法、およびメーリングリストの拡張などに関してリレーが重要な役割を果たします。バックエンドストアへの配信はこれまで SMTP 上で行われてきました。SMTP では、バックエンドシステムで LDAP ディレクトリの受取人アドレスを再度調べる必要があるため、MTA の全機能が使用されます。速度と効率性を向上するために、MTA では SMTP ではなく LMTP を使用してバックエンドストアにメッセージを配信できます。詳細は、『Sun Java System Messaging Server 管理ガイド』を参照してください。

注 Messaging Server への LMTP の実装は、汎用的な目的で行います。Messaging Server の LMTP は、2 階層アーキテクチャの Messaging Server MTA とメッセージストアコンポーネントとの間でだけ使用できます。1 つのマシンで構成される 1 階層アーキテクチャでは、LMTP は使用できません。

メッセージストア

メッセージストアは、インターネットメールメッセージの配信、取得、および操作のための専用のデータストアです。メッセージストアは IMAP4 および POP3 クライアントアクセスサーバーと共に動作し、サーバーにアクセスしてメッセージへの柔軟で容易なアクセスを提供します。メッセージストアは Webmail サーバーでも動作し、Web ブラウザにメッセージング機能を提供します。詳細については、この節のほかに、『Sun Java System Messaging Server 管理ガイド』を参照してください。

メッセージストアは、フォルダのセットまたはユーザーのメールボックスとして構成されます。フォルダまたはメールボックスは、メッセージのコンテナです。それぞれのユーザーには、新しく受信したメールが入る INBOX があります。それぞれのユーザーには、メールを格納できる 1 つ以上のフォルダがあります。フォルダには、他のフォルダを階層構造で含めることができます。個別のユーザーが所有するメールボックスは非公開フォルダです。非公開フォルダは、所有者の意志で、同じメッセージストア内の他のユーザーと共有できます。6.0 リリースでは、Messaging Server は複数のストア間でのフォルダの共有をサポートしています。

メッセージストアには、ユーザーファイルとシステムファイルの 2 つの一般領域があります。ユーザー領域では、それぞれのユーザーの INBOX の位置が 2 階層ハッシングアルゴリズムを使用して決定されます。それぞれのユーザーのメールボックスまたはフォルダは、その親フォルダ内の別のディレクトリとして表されます。それぞれのメッセージは、MIME 形式標準を使用してプレーンテキストで保存されます。フォルダ内に大量のメッセージがある場合は、システムによりフォルダのハッシュディレクトリが作成されます。ハッシュディレクトリを使用することで、フォルダに大量の

メッセージがある場合にファイルシステムが抱える負担が軽減されます。メッセージストアでは、メッセージ自体のほかに、メッセージヘッダー情報の索引とキャッシュ、およびその他の頻繁に使用されるデータが維持されるため、クライアントはメールボックスの情報を迅速に取得し、個別のメッセージファイルにアクセスすることなく一般的な検索を実行できます。

メッセージストアには、多くのメッセージストアパーティションを含めることができます。メッセージストアパーティションは、ファイルシステムボリュームに格納されます。ファイルシステムがいっぱいになると、それらのファイルシステムボリューム上に、追加のファイルシステムボリュームとメッセージストアパーティションを作成できます。

メッセージストアは、パーティションごとにメッセージそれぞれのコピーを 1 つずつだけ維持します。これは、シングルコピーメッセージストアとも呼ばれます。メッセージストアが複数のユーザー、グループ、または配信リストに宛てられたメッセージを受信した場合、それぞれのユーザーの INBOX にそのメッセージへの参照を追加します。メッセージのコピーをそれぞれのユーザーの INBOX に保存しないというより、メッセージストアでは同じデータの複製を保存しないようにしています。既読、返信済み、削除などの個別メッセージステータスのフラグは、それぞれのユーザーのフォルダごとに維持されます。

システム領域には、メッセージストア全体の情報が Berkeley データベース形式で格納されており、高速なアクセスを実現しています。システム領域内の情報は、ユーザー領域から再構築できます。Sun ONE Messaging Server 5.2 以降の製品には、データベーススナップショット機能があります。必要な場合には、データベースを元の状態に迅速に回復できます。現在の Messaging Server には、高速回復機能も追加されており、データベースが破損した場合には、データベース再構築のために長い時間待つことなく、メッセージストアをシャットダウンして、すぐに元の状態に戻すことができます。

メッセージストアは、IMAP 割り当て (QUOTA) 拡張 (RFC2087) をサポートしています。割り当ての拡張は有効にすることも無効にすることもできます。ユーザー割り当ては、バイト数またはメッセージ数を使用して設定できます。しきい値を設定して、割り当てがしきい値に達した場合には、ユーザーに警告を出すこともできます。ユーザーが割り当てを超過した場合は、猶予期間中の新規メッセージは保留され、再試行されます。猶予期間の後で、割り当てを超過したユーザーに送信されたメッセージは、未送信通知と共に送信者に返されます。

割り当てを使用する特別なアプリケーションで、ユーザーの割り当てステータスに関係なくメッセージが配信されなければならない場合には、保証メッセージ配信チャネルがあります。このチャネルは、割り当てステータスに関係なくすべてのメッセージを配信するのに使用できます。割り当て使用率のレポートと割り当て警告の送信を行うユーティリティも用意されています。

Directory サービス

Messaging Server は、Sun Java System Directory Server にバンドルされています。Directory Server は、Lightweight Directory Access Protocol (LDAP) ディレクトリサービスです。Directory Server は、Messaging Server の運用に不可欠な情報のための中央リポジトリを提供します。この情報には、ユーザープロファイル、配信リスト、およびその他のシステムリソースなどが含まれます。

ディレクトリ情報ツリー

ディレクトリは、ディレクトリ情報ツリー (DIT) として知られるツリー形式でデータを格納します。DIT は、ツリーの最上部に 1 つの主要ブランチがあり、その下にブランチおよびサブブランチがある階層構造です。DIT は、組織のニーズに合わせた配備の設計を可能にする柔軟性を備えています。たとえば、実際の業務組織構造に従った DTI の配置を選択することも、業務の地理的なレイアウトに従って選択することもできます。また、使用する DNS レイヤーに 1 対 1 でマッピングした DIT を設計することもできます。実稼動後の DIT の変更は大変な作業となるため、DIT の設計は慎重に行ってください。

DIT は、幅広い管理シナリオに適応する柔軟性も備えています。DIT は、集中型でも分散型でも管理できます。集中型の管理では、1 つの権限で DIT 全体を管理します。集中型管理の場合は、DIT 全体を 1 つのメールサーバー上に配置して使用します。分散型管理では、複数の権限で DIT を管理します。通常は、DIT がいくつかの部分、サブツリー、または異なるメールサーバーに分割された場合に分散型管理を用います。

DIT が大規模な場合、またはメールサーバーが地理的に分散されている場合は、DIT の一部の管理を委託することも検討します。通常は、DIT のそれぞれのサブツリーを管理する権限を割り当てます。Messaging Server では、1 つの権限で複数のサブツリーの管理が可能です。ただしセキュリティ上の理由で、権限は、その権限が所有する DIT のサブツリーの変更だけが可能となっています。

Identity Server が使用されない場合に Messaging Server が使用するデフォルトのスキーマは、Identity Server が使用するスキーマとは異なります。Messaging Server は、Sun Java System LDAP スキーマ 1 および 2 をサポートしており、スキーマの切り替えと移行も可能です。詳細は、第 7 章「Messaging Server スキーマとプロビジョニングオプションの理解」を参照してください。

ディレクトリのレプリケーション

Directory Server は、レプリケーションをサポートしており、冗長性と効率性を実現するさまざまな設定が可能です。1 つのホストから別のホストへの DIT の全部または一部をレプリケーションすることで、以下の設定機能が利用できます。

- ディレクトリ情報が 1 つのサーバー上にだけあるのではなく、複数のサーバーにレプリケートされるため、ディレクトリ情報へのアクセスがより容易になる

- ディレクトリ情報はローカルディレクトリサーバーにキャッシュされ、リモートディレクトリサーバーから情報にアクセスする手間を省いている。ディレクトリ情報のキャッシュにより、特に中央ディレクトリへのネットワーク帯域幅が限られている配備では、パフォーマンスが向上する
- 実際の設定次第で、複数のディレクトリサーバーは単独の集中型サーバーよりも、メールクライアントの要求をより高速に処理できる

ディレクトリのレプリケーション、ディレクトリパフォーマンスの調整、DIT 構造と設計の詳細については、以下の場所にある Sun Java System Directory Server のマニュアルを参照してください。

http://docs.sun.com/db/prod/entsys?l=ja

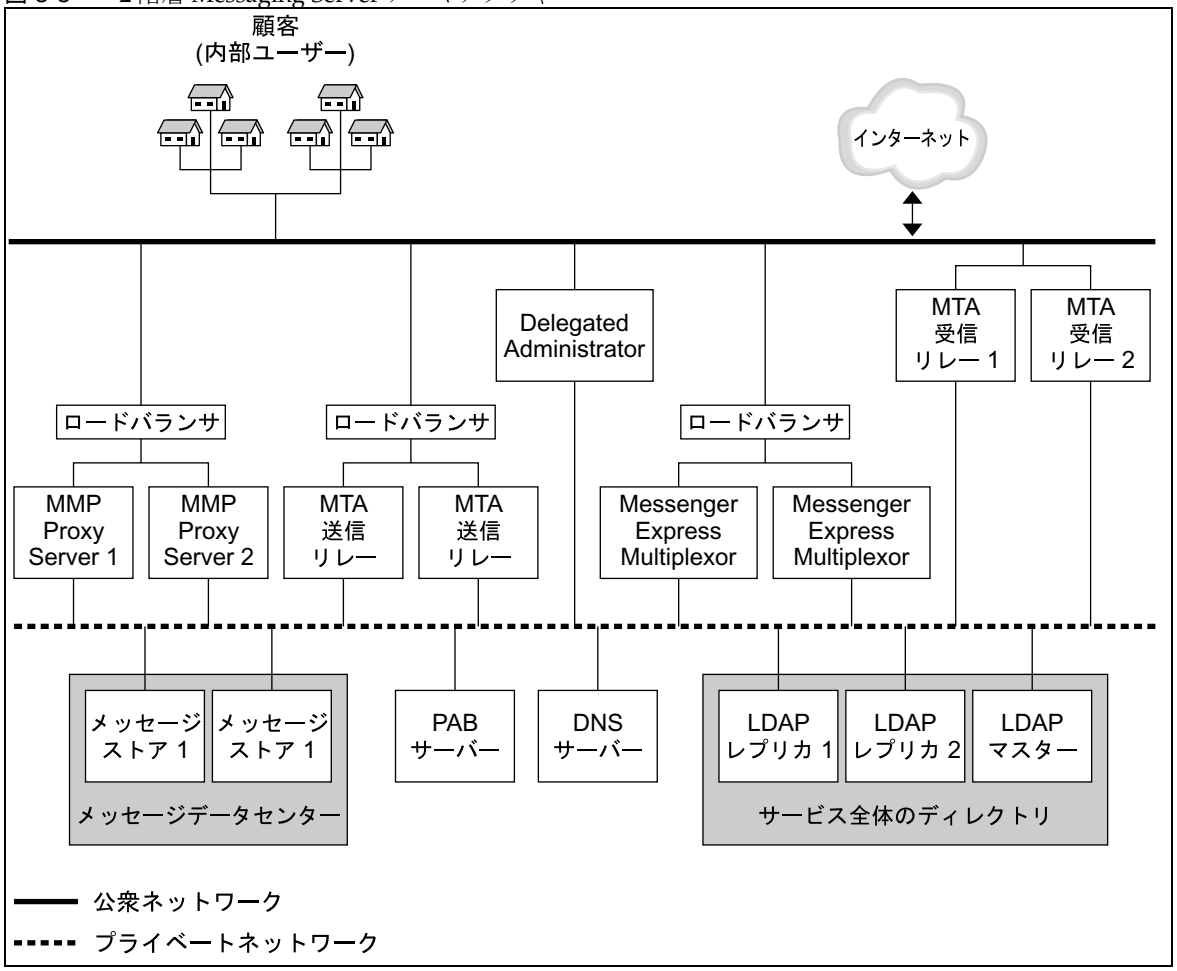
2 階層アーキテクチャの理解

2 階層アーキテクチャにより、スケーラビリティと信頼性のために最適化された設計が可能になります。単独のホストでメッセージングシステムのすべてのコンポーネントを実行する代わりに、2 階層アーキテクチャでは、コンポーネントを異なるマシンに分割します。このようなコンポーネントの分割により、特別な専用機能が実行されます。たとえば、追加のメッセージストレージが必要になるとか、より多くの送信リレーが必要になるなど、特定の機能コンポーネントの負荷が増大すると、サーバーを追加して増大する負荷に対応できます。

2 階層アーキテクチャは、アクセスレイヤーとデータレイヤーで構成されます。アクセスレイヤーは、配備の中で配信、メッセージアクセス、ユーザーログイン、および認証を処理する部分です。データレイヤーは、すべてのデータを維持する部分です。これには、LDAP マスターサーバーと、ユーザーメッセージを格納するよう設定された Messaging Server マシンが含まれます。

46 ページの図 3-3 は、簡略化した 2 階層アーキテクチャの説明です。

図 3-3　2 階層 Messaging Server アーキテクチャ



以下でこれらの機能部分について説明します。

公衆アクセスネットワーク : Messaging Server を内部ユーザーとインターネットに接続するネットワークです。それぞれの配備で独自のネットワーク要件が定義されますが、基本的な Messaging Server の要件は、SMTP、POP、IMAP、および HTTP のような標準のプロトコルを使用したエンドユーザーとインターネットへの接続性です。

プライベートデータネットワーク : このネットワークは、公衆アクセスネットワークと Messaging Server データの間で、セキュリティで保護された接続を提供します。セキュリティで保護されたアクセスレイヤーと、サービス全体のディレクトリ、メッセージデータセンター、および個人アドレスブック (PAB) サーバーが含まれるデータレイヤーで構成されます。

LDAP ディレクトリサーバー : ユーザーベースに関する情報の格納と取得に使用されるディレクトリサーバーです。ユーザーとグループエイリアス、メールホスト情報、配信設定などを格納します。設計の要件次第で、システムの同一ディレクトリを複数格納することも可能です。図 3-3 は、マスターディレクトリと 2 つのレプリカを示します。LDAP ディレクトリサーバーは、Messaging Server 製品の一部として提供されます。希望する場合は、既存のディレクトリのデータも使用できます。この例では、ユーザーとグループのデータを既存のディレクトリから取得し、それを Sun Java System Directory Server ディレクトリ内に置きます。既存のディレクトリのデータ形式は、Messaging Server スキーマに準拠している必要があります。

メッセージストア : ユーザーメールを保持し、格納します。「バックエンド」とも呼ばれます。メッセージストアは、IMAP サーバー、POP サーバー、および Messenger Express (Webmail) サーバーのような Message Access Component も参照します。46 ページの図 3-3 は、2 つのメッセージストアを持つ配備を示します。必要に応じて、さらにストアを追加することもできます。

個人アドレスブック (PAB) サーバー : Messenger Express のユーザーアドレスの格納と取得を行います。

DNS サーバー : ホスト名を IP アドレスにマップします。DNS サーバーは、メッセージを外部ホストにルーティングする時に、どのホストに接続するかを判断します。内部的には、DNS は実際のサービスをマシン名にマップします。DNS サーバーは、Messaging Server 製品ラインの一部ではありません。Messaging Server をインストールする前に、稼働状態の DNS サーバーをインストールする必要があります。

サーバーロードバランサ : ネットワーク接続について、均一にバランスを取るか、複数のサーバーにアルゴリズムを適用してバランスを取ります。ロードバランサを使用すると、1 つのネットワークアドレスで多数のサーバーを表すことができるため、トラフィックのボトルネックを解消し、トラフィックフローの管理と高いレベルのサービス保証が可能になります。46 ページの図 3-3 では、2 つのロードバランサを使用しています。1 つのバランサは MMP に接続され、もう 1 つは MTA 送信リレーに接続されています。ロードバランサは Java Enterprise System 製品ラインの一部ではありません。ロードバランサをメッセージストアまたはディレクトリマスター上で使用することはできません。ロードバランサは、MMP、MEM、Communications Express、受信 MTA または送信 MTA、ディレクトリコンシューマ、Messaging Server の MTA を使用しない Brightmail 製品、Brightmail サーバーに接続して使用します。

MTA 受信リレー : 外部 (インターネット) サイトからのメッセージを受信し、それをローカルのメッセージストアサーバーにルーティングする専用 MTA です。この MTA は外部からの最初のコンタクトポイントとなるため、MTA 受信リレーには、権限のないリレーを防ぎ、スパムをフィルタリングし、サービス拒否攻撃に対抗する機能が追加されます。

MTA 送信リレー : 内部または認証されたユーザーからのメールだけを受け取り、それをその他の内部ユーザーまたは外部 (インターネット) ドメインにルーティングする MTA です。単独のマシンを受信リレーと送信リレーとして使用できますが、インターネットに接続された大規模な配備では、これらの機能を 2 つの別のマシンに分割します。このようにすると、内部クライアントは、外部サイトから受信するメールと競合することなく、メールを送信できます。

送信リレーに内部配信をさせないようにする、別のルーティングオプションもあります。送信リレーは、ユーザーベースからの内部宛てのメールを単にルーティングのインスタンスとして認識し、そのようなメッセージをすべて受信 MTA に送信します。

Delegated Administrator サーバー : ユーザーと管理者のための GUI 管理コンソールを提供します。Delegated Administrator を使用して、ユーザーはパスワードの変更、休暇通知メールの設定などを行うことができます。管理者は、ユーザーの追加や削除など、より高度な管理タスクを行うことができます。Delegated Administrator は、現在スキーマ 1 の環境でのみ動作します。

Messaging Multiplexor または Mail Message Proxy または MMP : ユーザーのメールボックスを含む特定のマシンと、関連付けられた DNS 名との結合を解除して、複数の物理マシンにわたるメッセージストアの拡張を可能にします。クライアントソフトウェアは、メッセージストアのある物理的なマシンを知る必要はありません。このようにすると、ユーザーは、メールボックスが新しいマシンに移動されるたびにホストメッセージストアの DNS 名を変更する必要がなくなります。POP クライアントまたは IMAP クライアントがメールボックスへのアクセスを要求すると、プロキシはディレクトリサービスを参照してユーザーのメールボックスがある場所を検索し、そのメールボックスがある Messaging Server システムに要求を転送します。

Messenger Express Multiplexor : Webmail 用の HTTP アクセスサービスへの単一の接続ポイントとして機能する特別なサーバーです。すべてのユーザーがこのメッセージングプロキシサーバーに接続し、ここで該当するメールボックスに転送されます。このため、メールユーザーには複数の Messaging Server が単一のホスト名であるかのように表示されます。Messaging Multiplexor (MMP) は POP および IMAP サーバーに接続しますが、Messenger Express Multiplexor は HTTP サーバーに接続します。つまり、Messenger Express Multiplexor と Messenger Express との関係は、MMP と POP や IMAP との関係と同じです。

2 階層アーキテクチャ — メッセージングデータフロー

この節では、メッセージングシステム経由のメッセージフローについて説明します。メッセージフローがどのように機能するかは、実際のプロトコルとメッセージパス次第です。

メールの送信 : 内部ユーザーから別の内部ユーザーへ

概要 : 内部ユーザー -> ロードバランサ -> MTA 送信リレー 1 または 2 -> MTA 受信リレー 1 または 2 -> メッセージストア 1 または 2

注	送信リレーからストアにメールを直接配信させるために、LMTP を使用するのが一般的になってきています。2 階層配備では、この方法を選択できます。

内部ユーザーから別の内部ユーザー (すなわち同じ電子メールシステムのユーザー) へ宛てられたメッセージは、最初にロードバランサに送られます。ロードバランサは、基盤となるサイトアーキテクチャから電子メールユーザーを切り離し、高可用性電子メールサービスを提供します。ロードバランサは、その接続を MTA 送信リレー 1 または 2 のいずれかに送信します。送信リレーはアドレスを読み取り、メッセージが外部ユーザー宛てのものか内部ユーザー宛てのものかを判断します。外部ユーザー宛ての場合は、そのメッセージをインターネットに送信します。内部ユーザー宛ての場合は、MTA 受信リレー 1 または 2 に送信するか、設定によっては適切なメッセージストアに直接送信します。MTA 受信リレーは、そのメッセージを適切なメッセージストアに配信します。メッセージストアはそのメッセージを受け取り、メールボックスに配信します。

メールの取得 : 内部ユーザー

概要 : 内部ユーザー -> ロードバランサ -> MMP/MEM/Communications Express Proxy Server 1 または 2 -> メッセージストア 1 または 2

メールは POP、HTTP、または IMAP のいずれかを使用して取得されます。ユーザー接続がロードバランサに受信され、MMP、MEM、または Communications Express サーバーのいずれかに転送されます。次にユーザーは、接続したアクセスマシンにログイン要求を送信します。アクセスレイヤーのマシンは、ログイン要求とパスワードを検証し、ユーザー接続で指定されたものと同じプロトコルを使用して、適切なメッセージストア (1 または 2) に要求を送信します。そしてアクセスレイヤーのマシンは、クライアントとサーバー間の残りの接続を仲介します。例外は、進行中のユーザー要求を処理しているレベルの Communications Express にブラウザのレンダリングを処理させる場合です。

メールの送信 : 内部ユーザーから外部 (インターネット) ユーザーへ

概要 : 内部ユーザー -> ロードバランサ -> MTA 送信リレー 1 または 2 -> インターネット

内部ユーザーから外部ユーザー (すなわち異なる電子メールシステムのユーザー) へ宛てられたメッセージは、最初にロードバランサに送られます。ロードバランサは、電子メールユーザーを基盤となるサイトアーキテクチャから切り離し、高可用性電子メールサービスを提供します。ロードバランサは、メッセージを MTA 送信リレー 1 または 2 に送信するか、設定によっては適切なメッセージストアに直接送信します。送信リレーはアドレスを読み取り、メッセージが外部ユーザー宛てのものか内部ユーザー宛てのものかを判断します。外部ユーザー宛ての場合は、そのメッセージをインターネット上の MTA に送信します。内部ユーザー宛ての場合は、MTA 受信リレー 1 または 2 に送信します。MTA 受信リレーは、そのメッセージを適切なメッセージストアに配信します。メッセージストアはそのメッセージを受け取り、適切なメールボックスに配信します。

メールの送信 : 外部 (インターネット) ユーザーから内部ユーザーへ

概要 : 外部ユーザー -> MTA 受信リレー 1 または 2 -> メッセージストア 1 または 2

外部ユーザー (インターネット) から内部ユーザーへのメッセージは、MTA 受信リレー 1 または 2 (ロードバランサは不要) に送られます。受信リレーはアドレスを読み取り、メッセージが外部ユーザー宛てのものか (インターネットリレーが許可されている場合) 内部ユーザー宛てのものかを判断します。外部ユーザー宛ての場合は、受信リレーはそのメッセージをインターネット上の別の MTA に送信します。内部ユーザー宛ての場合は、受信リレーは LDAP 検索を使用して メッセージストア 1 または 2 のいずれに送信するかを判断し、それに従って配信します。メッセージストアはそのメッセージを受け取り、適切なメールボックスに配信します。

水平スケーラビリティと垂直スケーラビリティの理解

スケーラビリティは、メッセージングサービスの利用拡大に対応する配備の能力です。スケーラビリティにより、ユーザー数の急激な拡大をシステムがどのように受け入れられるかが決まります。またスケーラビリティにより、たとえば 1 か月の間にユーザーの多くが SSL の使用を希望するなどというような、ユーザーの行動の大きな変化にシステムがどのようにうまく適応できるかも決まります。

この節では、個別のサーバーとサーバー全体でサービスの拡大を吸収するために、アーキテクチャに追加する機能について確認します。以下のトピックについて説明しています。

- 水平的スケーラビリティの計画
- 垂直スケーラビリティの計画

水平的スケーラビリティの計画

水平スケーラビリティは、アーキテクチャにサーバーを追加することがどの程度容易であるかを示します。ユーザー数が拡大する、またはユーザーの行動が変化するにつれて、やがては既存のアーキテクチャのリソースを最大化しなければならなくなります。慎重に計画を立てて、アーキテクチャのスケールを適切に拡張する方法を決めます。

アーキテクチャの水平的拡張を行う場合には、リソースを複数のサーバーに分散します。水平スケーラビリティでは、2 つの方法が使用されます。

- 複数サーバーへのユーザーベースの分散
- 冗長コンポーネントへのリソース分散

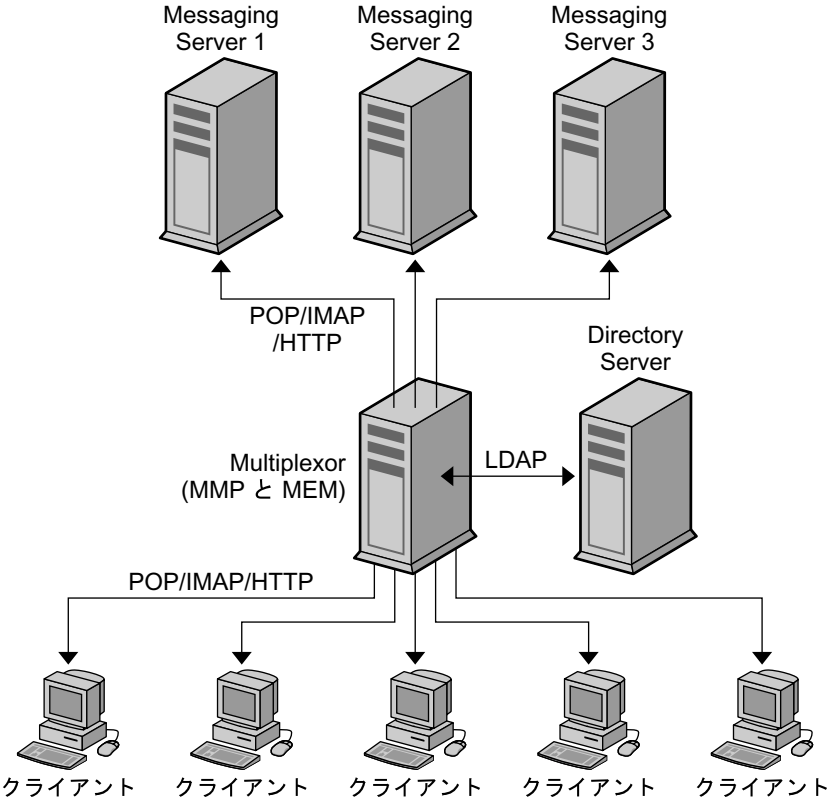
複数サーバーへのユーザーベースの分散

負荷を複数のサーバーに分散するには、クライアントのメールをいくつかのバックエンドメッセージストアに均等に分割します。ユーザーをアルファベット順に分けたり、サービスのクラス別、部門、または物理的な場所別に分けたりして、特定のバックエンドメッセージストアホストに割り当てます。

Messaging Multiplexor (MMP) は、複数のサーバーの受信クライアント接続を処理するマルチスレッドサーバーです。MMP は POP 接続または IMAP 接続を受け入れ、LDAP 検索を実行して認証を行い、その接続を適切なメッセージングサーバーにルーティングします。HTTP 接続では、Messenger Express Multiplexor (MEM) を有効にして、複数のサーバーの受信クライアント接続を処理します。Communications Express も同様に機能します。

MMP と Messenger Express Multiplexor は、管理を容易にする目的でしばしば同じマシンに置かれます。52 ページの図 3-4 は、ユーザーが複数のバックエンドサーバーに分割され、受信クライアント接続の処理に Multiplexor を使用するサンプルアーキテクチャを示します。

図 3-4 複数サーバーへのユーザーベースの分散



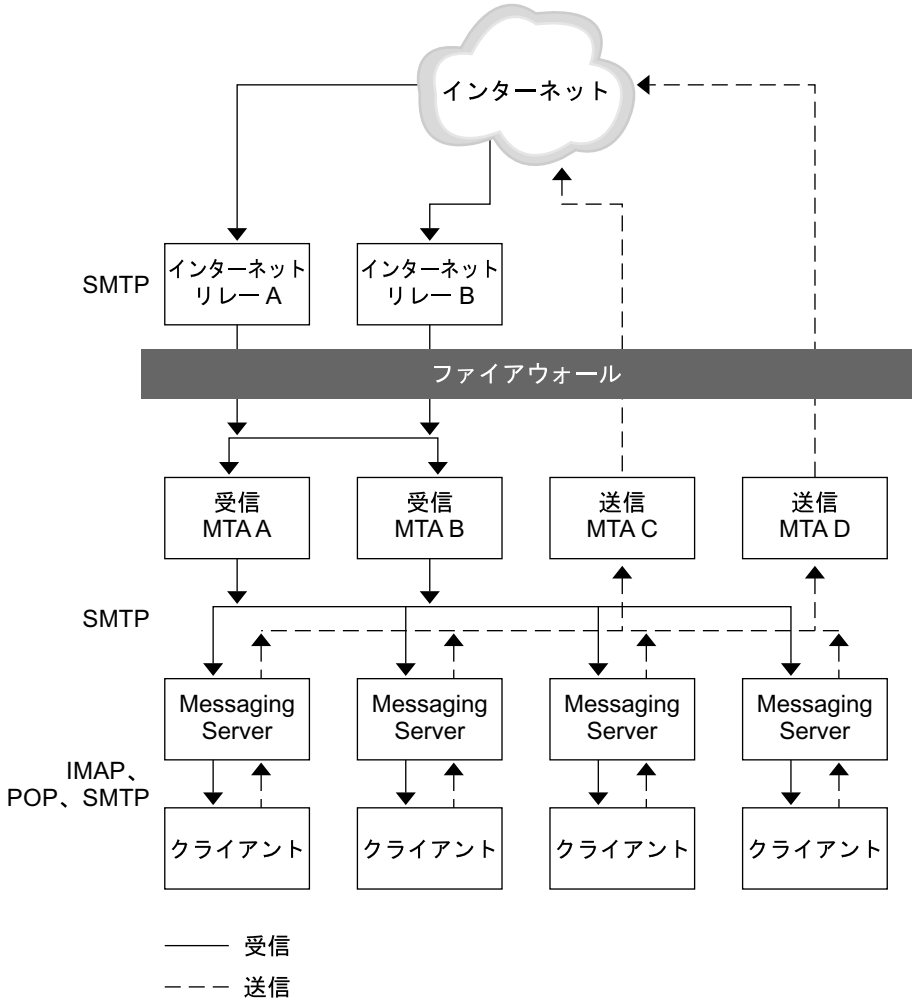
ユーザーをバックエンドサーバーに分散することで、MMP または MEM を使用するかぎり、ユーザーの管理が簡単になります。ユーザーは、メールがある 1 つのバックエンドサーバーに接続するため、すべてのユーザーに対して設定を標準化できます。この設定により、複数のサーバーの管理も容易になります。また、Messaging Server ホストへの要求増加に対応して、ホストをシームレスに追加できます。

冗長コンポーネントへのリソース分散

電子メールが、組織の日々の運営上、重要な地位を占める場合は、メッセージングシステムが常に運用可能な状態であることを確実にするために、ロードバランサ、MX レコードのような冗長コンポーネントとリレーが必要になります。

以下の図は、リソースを冗長 MTA リレーに分散した例です。インターネットリレー、受信 MTA、送信 MTA のような同じコンポーネントのセットが、52 ページの図 3-4 で使用されています。ただしこのケースでは、それぞれ 2 つずつが配備されています。

図 3-5　冗長コンポーネントへのリソース分散



冗長 MTA リレーを使用することで、あるコンポーネントが動作不能に陥っても、別のコンポーネントが常に使用可能な状態にあります。また、リソースを冗長 MTA リレーに分散することで、負荷の分散も行われます。たとえば、以前は 1 つのリレーが管理していた負荷が、2 つのインターネットリレーに分散されます。この冗長性により、Messaging Server システムにフォールトトレランスも提供されます。それぞれの MTA リレーが、他の MTA リレーの機能を受け持ちます。

冗長ネットワーク接続をサーバーとメールリレーにインストールすることで、ネットワークの問題に対するフォールトトレランスが実現します。メッセージング配備が組織にとってより重要なものであるほど、フォールトトレランスと冗長性の検討もより重要になります。

MX レコード、リレー、およびロードバランサの詳細については、以下の節で説明します。

MX レコード

等しい優先度の MX レコードにより、メッセージが冗長化されたインターネットリレーと受信 MTA、送信 MTA にルーティングされます。たとえば、送信 MTA は、`relayA.siroe.com` と `relayB.siroe.com` に対応する `siroe.com` の MX レコードを検索します。優先度が同じためにこれらのリレーの 1 つがランダムに選択され、SMTP 接続が開かれます。最初に選択されたリレーが応答しなかった場合は、メールは別のリレーに送信されます。以下の MX レコードの例を参照してください。

```
siroe.com in MX 10 relayA.siroe.com
siroe.com in MX 10 relayB.siroe.com
```

リレー

Messaging Server ホストがそれぞれ多数のユーザーをサポートしており、SMTP メールの送信負荷が高い場合、メールリレーを使用することで Messaging Server ホストはルーティングタスクから開放されます。異なるリレーを指定して、それぞれにメッセージの送信と受信を処理させることで、さらに負荷の分散を図ることもできます。

受信リレーと送信リレーの両方が、1 つの In/Out SMTP リレーホストとして組み合わされる場合もあります。1 つまたは複数のリレーホストが必要であるかどうかを判断するには、アーキテクチャ全体の受信および送信メッセージのトラフィック特性を確認します。

ロードバランサ

ロードバランサを使用すると、負荷を複数のサーバーに分散して、どれか 1 つのサーバーでも過負荷状態にならないようにします。ロードバランサは、クライアントからの要求を受けて、各サーバーの CPU とメモリの使用率を追跡するようなアルゴリズムを使用して、利用可能なサーバーに要求をリダイレクトします。ロードバランサは、共通サーバーで実行されるソフトウェアとして、純粋に外部のハードウェアソリューションとして、またはハードウェアとソフトウェアを組み合わせたパッケージとして使用可能です。

垂直スケーラビリティの計画

垂直スケーラビリティは、CPU の追加など、個々のサーバーマシンへのリソースの追加に関係があります。それぞれのマシンには、一定の負荷を処理できる能力があります。一般には、リソースに制限があるか、配備の拡大に応じて追加のハードウェアを購入できない場合に、配備における垂直スケーラビリティを検討します。

配備の垂直的スケールを行うには、以下のことが必要です。

- 各メッセージングコンポーネントのサイズを決定する

  第 6 章「サイズ決定戦略の計画」の「アーキテクチャ戦略の構築」を参照
- システムのプロトタイプの負荷をテストする

  第 6 章「サイズ決定戦略の計画」の「負荷シミュレータ」を参照
- システムパフォーマンスを監視し、それに従って配備を調整する

高可用性に向けた計画

高可用性は、計画された停止時間と予期しない停止時間を短時間にとどめるための配備の設計です。通常、高可用性設定は緩やかに結合された 2 つ以上のシステムで構成されたクラスタです。各システムがそれぞれのプロセッサ、メモリ、オペレーティングシステムを持っています。ストレージはシステム間で共有されます。特別なソフトウェアがシステムをバインドし、単一点での障害からシステムが完全に自動的に回復できるようにします。Messaging Server には、Sun™ Cluster サービスと Veritas® クラスタリングソリューションをサポートする高可用性のオプションが用意されています。

高可用性に向けた計画を作成する場合は、可用性とコストとのバランスを検討する必要があります。一般に、より可用性の高い配備では、設計と運用のコストも高くなります。

高可用性は、アプリケーションサービスの中断や停止時間によるデータアクセス機会の損失に対する保険です。アプリケーションサービスが利用不能になった場合、組織は収入、顧客、その他の機会を失うことになります。組織にとっての高可用性の価値は、停止時間のコストに直接関係します。停止時間のコストが高くなるほど、高可用性のための追加コストを正当化するのも容易になります。また、一定のレベルの可用性を保証するサービスレベル契約を組織が結んでいる場合もあります。可用性の目標を達成できない場合、財務的な打撃を直接受ける可能性があります。

詳細は、第 10 章「サービス可用性に向けた計画」を参照してください。

Messaging Server アーキテクチャのパフォーマンスの考慮事項

この節では、Messaging Server コンポーネントのパフォーマンス特性を評価して、正確なアーキテクチャの開発を行う方法について説明します。

この節では以下の内容について説明します。

- メッセージストアのパフォーマンスの考慮事項
- MTA パフォーマンスの考慮事項
- メールメッセージプロキシ (MMP) パフォーマンスの考慮事項
- Messenger Express Multiplexor (MEM) パフォーマンスの考慮事項

メッセージストアのパフォーマンスの考慮事項

メッセージストアのパフォーマンスは、以下のようなさまざまな要素に影響を受けます。

1. ディスク入出力
2. 受信メッセージレート (メッセージ挿入レートとも呼ばれる)
3. メッセージサイズ
4. ログインレート (POP/IMAP/HTTP)
5. IMAP および HTTP のトランザクションレート
6. さまざまなプロトコルの並行接続数
7. ネットワーク入出力

前の要素リストは、メッセージストアに影響を与えるおおよその順序で記載されています。メッセージストレージに関するパフォーマンス問題のほとんどは、ディスクの入出力能力が不十分なことに原因があります。さらに、物理ディスク上のストアのレイアウトもパフォーマンスに影響を与えます。より小規模のスタンドアロンシステムでは、単純なディスクのストライピングでも十分な入出力が得られます。ほとんどの大規模システムでは、ファイルシステムを分離し、ストアのさまざまな部分に入出力を提供します。

メッセージングサーバーのディレクトリ

Messaging Server は 5 つのディレクトリを使用して大量の入出力活動に対応しています。これらのディレクトリは高頻度でアクセスされるため、ディレクトリごとにディスクを用意するか、より理想的には、ディレクトリごとに RAID を用意します。以下の表で、これらのディレクトリについて説明します。

表 3-1 アクセス頻度の高い Messaging Server ディレクトリ

高入出力ディレクトリ	説明とパラメータの定義
MTA キューディレクトリ	このディレクトリでは、MTA チャネルを通る各メッセージについて 1 つずつのファイルが、大量に作成される。ファイルが次の目的地に送信されると、そのファイルは削除される。ディレクトリの場所は、`imta_tailor` ファイルの `IMTA_QUEUE` オプションにより制御される。MTA キューディレクトリを変更する前に、『Sun Java System Messaging Server 管理ガイド』にあるこのオプションについての説明を参照 デフォルトの場所 : *msg_svr_base*`/data/imta/queue`

表 3-1 アクセス頻度の高い Messaging Server ディレクトリ (続き)

高入出力ディレクトリ	説明とパラメータの定義
Messaging Server ログディレクトリ	このディレクトリには、新しいログメッセージが常に追加されるログファイルがある。変更の回数は、ログレベルの設定による。ディレクトリの場所は、`configutil` のパラメータ、`logfile.*.logdir` により制御される。* は admin、default、http、imap、または pop のようなログにより生成されたコンポーネントを示す。MTA ログファイルは `imta_tailor` ファイルの IMTA_LOG オプションを使用して変更できる デフォルトの場所 : *msg_svr_base*/data/log
メールボックスデータベースファイル	これらのファイルはキャッシュの同期と継続的な更新を必要とする。このディレクトリは最も高速なディスクボリュームに配置する。これらのファイルは常に *msg_svr_base*/data/store/mboxlist ディレクトリに存在する
メッセージストアインデックスファイル	これらのファイルにはメールボックス、メッセージ、ユーザーに関するメタ情報が含まれる。デフォルトではこれらのファイルはメッセージファイルと共に格納される。`configutil` パラメータの `store.partition.*.path` はディレクトリの場所を制御する。* はパーティション名を示す。リソースに余裕がある場合は、これらのファイルを 2 番目に高速なディスクボリュームに配置する デフォルトの場所 : *msg_svr_base*/data/store/partition/primary
メッセージファイル	これらのファイルにはメッセージが含まれており、メッセージごとに 1 つのファイルとなっている。ファイルは頻繁に作成され、変更されることはなく、最終的には削除される。デフォルトでは、これらのファイルはメッセージストアインデックスファイルと同じディレクトリに格納される。ディレクトリの場所は、`configutil` のパラメータ `store.partition.*.path`、で制御される。ここで * はパーティション名を示す サイトによっては、`store.partition.primary.path` によって指定される、プライマリと呼ばれる単独のメッセージストアパーティションがある。大規模なサイトでは、`store.partition.*.path` により指定される追加パーティションがある。ここで * はパーティション名を示す デフォルトの場所 : *msg_svr_base*/data/store/partition/primary

以下の節では、Messaging Server の高頻度アクセスディレクトリについてさらに詳しく説明します。

MTA キューディレクトリ

非 LMTP 環境では、MTA キューディレクトリは高頻度で使用されます。LMTP は、受信メッセージが MTA キューに置かれず、ストアに直接挿入されるように機能します。このメッセージの挿入により、メッセージストアマシンの全体的な入出力要件が少なくなり、メッセージストアマシンの MTA キューディレクトリの使用頻度が大きく減少します。システムがスタンドアロンの場合、または Webmail 送信のためのローカル MTA を使用する場合は、送信メールトラフィックのために、まだかなりの入出力が発生します。LMTP を使用した適切な 2 階層環境では、このディレクトリは軽い頻度で使用される程度です。Messaging Server の前のバージョンでは、大規模なシステムではこのディレクトリをそれ自身のストライプまたはボリューム上に設定する必要がありました。

ログファイルディレクトリ

ログファイルディレクトリでは、設定されているログのレベルにより、さまざまな量の入出力が要求されます。メッセージストアのその他の高入出力要求とは異なり、ログディレクトリへの入出力は非同期です。典型的な配備シナリオでは、LUN 全体をログ専用には使用しません。かなり規模の大きなストア配備、または大量のログが必要な環境では、専用の LUN を使用するのが理に適っています。

ほとんどすべての環境で、メッセージストアをデータ喪失から守る必要があります。要求される喪失からの保護と継続的な可用性のレベルは、RAID5 のような単純なディスク保護から、ミラーリング、日常的なバックアップ、リアルタイムのデータ複製、リモートデータセンターまで、さまざまです。データの保護に関しても、Automatic System Recovery (ASR) が可能なマシンから、ローカル HA 機能、自動リモートサイトフェイルオーバーまで、さまざまなものがあります。これらの決定は、ハードウェアの量とサービスの提供に必要なサポート要員の数に影響します。

mboxlist ディレクトリ

`mboxlist` ディレクトリには入出力が非常に集中しますが、特にサイズが大きいというわけではありません。`mboxlist` には、ストアとトランザクションログで使用される Sleepycat (Berkeley) データベースがあります。高頻度の入出力があり、それを分割できないことから、大規模な配備では `mboxlist` ディレクトリをそれ自身のストライプかボリューム上に配置する必要があります。これは、メッセージストアの多くの操作が Sleepycat データベースにアクセスするため、垂直的スケーラビリティの喪失の原因にもつながります。アクセスが激しいシステムでは、これがボトルネックになります。`mboxlist` ディレクトリの入出力パフォーマンスのボトルネックによって、ストアの raw パフォーマンスと応答時間が悪くなるだけでなく、垂直的スケーラビリティも減少します。バックアップから高速に回復することが要求されるシステムでは、このディレクトリを Solid State Disks (SSD) 上に配置するか、パフォーマンスの高いキャッシングアレイを使って、ファイルシステム上でサービスを継続したまま障害回復処理を進行できるような高い書き込みレートを許可します。

複数のストアパーティション

メッセージストアは、複数のストアパーティションをサポートしています。各パーティションを、それ自身のストライプまたはボリューム上に配置します。ストア上に配置するパーティションの数は、さまざまな要素により決定されます。明確な要素としては、サーバーのピーク負荷時の入出力要件があります。追加のストアパーティションとしてファイルシステムを追加することで、メールの配信や取得のためにサーバーで可能な IOPS (1 秒あたりの総入出力) を引き上げます。ほとんどの環境で、大きくて数が少ないストライプあるいは LUNS よりも、多数の小さなストライプあるいは LUNS のほうが、より大きな IOPS が得られます。

いくつかのディスクアレイを使用すると、アレイを 2 つの異なる方法で設定できます。それぞれのアレイを LUN として設定し、それをファイルシステムにマウントします。または、それぞれのアレイを LUN として設定し、それをサーバー上でストライプします。どちらも有効な設定です。ただし、複数のストアパーティション (小さいアレイでは 1 つのパーティション、または LUN のストライプセットをサーバーボリュームにした大きなアレイ上の複数のパーティション) は最適化と管理が容易です。

ただし、通常は raw パフォーマンスは、ストアパーティションの数を決定する場合の優先事項とはなりません。企業環境では、IOPS よりも容量のほうが重要となる場合が多いでしょう。また、LUN をソフトウェアストライプで設定し、1 つの大きなストアパーティションとすることも可能です。ただし、複数の小さなパーティションのほうが、一般に管理は容易です。ストアパーティションの数を決定する際に適切な最優先事項は、一般的には回復時間です。

ストアパーティションの回復時間は、いくつかのカテゴリに分類されます。

- 最初に、電源、ハードウェア、またはオペレーティングシステムの障害によるクラッシュからの回復と並行して、`fsck` が複数のファイルシステム上で動作する。HA プラットフォームで強く推奨され、要求されるジャーナリングファイルシステムを使用している場合は、この要素は小さなものとなる
- 2 番目に、バックアップおよび回復プロシージャが複数のストアパーティション上で並行して実行される。メッセージストアではすべてのストアパーティションで単独のデータベースが使用されているため、この並行動作は `mboxlist` ディレクトリの垂直的スケーラビリティにより制限される。ストアパーティションあたりの 1 つのスレッドの実行と並行して、ストアクリーンアッププロシージャ (`expire` および `purge`) が実行される
- 最後に、再ミラーリングまたは RAID 再同期プロシージャが、小さな LUN で高速に実行される。ここでは厳密なルールはないが、多くのケースでの推奨事項として、ストアパーティションは 10 個以上のディスクで構成すべきではない

ストレージアレイで使用されるドライブのサイズは、容量要件に対する IOPS 要件という問題になります。ほとんどの住居用 ISP POP 環境では、「より小さなドライブ」を使用します。大規模な割り当てによる企業配備では、「より大きな」ドライブを使用します (比較として、Sun ディスクアレイにおける小さなドライブは 36G バイト、大きなドライブは 73G バイト以上)。繰り返しになりますが、すべての配備は異なっており、一連の要件を個別に検討する必要があります。

メッセージストアのスケーラビリティ

マルチプロセスとマルチスレッドにより、メッセージストアは良好なスケール化がなされています。実際には、メッセージストアは 1 つのプロセッサから 4 つのプロセッサまで、一次直線形の比率を上回るスケール化が行われています。これは、4 つのプロセッサシステムは、1 つのプロセッサシステムを 4 つ合わせたものよりも大きな負荷を処理できることを意味します。メッセージストアは 4 から 12 のプロセッサ数についてもかなり直線形でスケール化されます。12 から 16 のプロセッサ数では、能力は増強されますが、直線形ではなくなります。LMTP を使用すると、同じサイズのストアシステムでサポートされるユーザー数は大きく増加しますが、メッセージストアの垂直的スケーラビリティはより制限されます。

MTA パフォーマンスの考慮事項

MTA のパフォーマンスは多くの要素に影響されます。影響を及ぼす要素には以下の項目が含まれますが、これらに限定されません。

- ディスクパフォーマンス
- SSL の使用
- 送受信のメッセージ数および接続数
- メッセージのサイズ
- 対象宛先数およびメッセージ数
- MTA との接続スピードと接続待ち時間
- スパムフィルタリングまたはウィルスフィルタリングの必要性
- SIEVE ルールとその他のメッセージ解析 (変換チャネルの使用など) の使用

MTA ルーターは CPU と入出力を集中的に使用します。MTA では、キューディレクトリ用とログディレクトリ用の 2 つの異なるファイルシステムが使用されます。小規模なホスト (4 プロセッサ以下) では、MTA ルーターとして機能し、これらのディレクトリを別のファイルシステムに分ける必要はありません。キューディレクトリでは、かなり大きい量で同期書き込みが行われます。ログディレクトリでは、小さな量の非同期書き込みが連続的に行われます。

ほとんどのケースで、ディスクサブシステムの MTA で冗長性を導入して、ディスクの障害時にメールデータが永久に失われることを回避したいと考えるでしょう。ディスクの障害は、ハードウェアの障害で最も起こる可能性の高いものです。これは、多くの内部ディスクを持つ外部ディスクアレイやシステムが最適だということを意味します。

MTA RAID のトレードオフ

外部 RAID コントローラデバイスとソフトウェアミラーによる JBOD アレイの使用との間にはトレードオフの関係があります。JBOD によるアプローチは、ハードウェアの購入という点では安価な場合がありますが、より多くのラックスペースと電力を必要とします。JBOD アプローチは、ソフトウェアによるミラーリングを行うことでサーバーのパフォーマンスを少し低下させ、一般的には保守コストも高くなります。ソフトウェア RAID5 は、パフォーマンスへの影響が非常に大きいため、代わりに使うことができません。そのため、RAID5 を使用する場合は、RAID5 キャッシングコントローラアレイを使用します。

MTA のスケーラビリティ

MTA ルーターでは、8 つを超えるプロセッサを使用できません。また、1 つから 4 つまでのプロセッサ数では、メッセージストアのように直線形以上の比率でスケール化されます。

MTA と高可用性

MTA ルーターを HA の制御のもとに置くのはあまりお勧めできません。しかし、それが保証されている環境では例外です。ハードウェアの障害時にも、メールの配信を短時間で指定した時間枠内で実行しなければならないという要件がある場合は、MTA を HA のソフトウェア制御のもとに配置します。ほとんどの環境では、ピーク負荷要件に対応できるように MTA の数を単純にいくつか増やします。これにより、1 つの MTA で障害が発生した場合でも、または大規模な配備環境で何らかの理由で複数の MTA ルーターの接続が遮断された場合でも、適切なトラフィックフローが生み出されます。

さらに、MTA の配置に関しては、MTA を常にファイアウォールに配置するよう配慮します。

メールメッセージプロキシ (MMP) パフォーマンスの考慮事項

MMP では、ログ以外の目的でディスク入出力が使用されることはありません。MMP は、CPU とネットワークに完全に結合されています。その他の Messaging Server コンポーネントとは異なり、MMP はマルチプロセスマルチスレッドの機能を備えていません。主要な実行コードは、シングルプロセスマルチスレッドです。したがって、MMP は十分にマルチプロセス化されていないため、その他のコンポーネントのようなスケール化はできません。

MMP では、5 つ以上のプロセッサは使用できません。また、2 つから 4 つのプロセッサ数でも直線形以下のスケール化となります。MMP には、2 つのプロセッサを備えたラックマウントのマシンが適しています。

その他のコンポーネントソフトウェア (MEM、Calendar Server フロントエンド、Communications Express Web Client、LDAP プロキシなど) を MMP と同じマシンに配置する配備を選択した場合は、4 つのプロセッサによる SPARC マシンによる配備の拡張を検討します。そのような構成を行うことにより、管理、パッチの導入、監視などが必要なマシンの総数を減らすことができます。

MMP のサイズは、接続レートとトランザクションレートにより決まります。POP のサイズ決定は、POP 接続がほとんどアイドル状態にならないため、きわめて明快です。POP 接続では、接続が行われ、作業が行われ、そして接続が遮断されます。IMAP のサイズ決定はより複雑です。IMAP では、ログインレート、並行レート、接続のビジー状態の起こり方について確認する必要があります。MMP も、接続の待ち時間と帯域幅に多少影響を受けます。MMP はメッセージストアからクライアントに送信されるデータのバッファとして機能するため、ダイアルアップ環境では、ブロードバンド環境の場合よりも並行して処理できるユーザーの数が少なくなります。

SSL の使用率が接続のかなりの割合を占める場合は、ハードウェアアクセラレータをインストールします。

MMP と高可用性

MMP は HA の制御のもとに配置してはなりません。個別の MMP には静的データはありません。可用性の高い環境では、1 つ以上の MMP マシンを追加して、1 つ以上の MMP が停止してもピーク負荷に対して十分な能力を確保します。Sun Fire Blade™ Server ハードウェアを使用する場合は、Blade ラックユニット全体が停止する可能性を考慮して、適切な冗長性の配備を計画します。

Messenger Express Multiplexor (MEM) パフォーマンスの考慮事項

MEM では、Webmail クライアントに対して中階層のプロキシが提供されます。このクライアントを使用して、ユーザーはブラウザを通じてメールにアクセスし、メールを作成できます。MEM のメリットは、メールを格納しているのはバックエンドサーバーであるにもかかわらず、エンドユーザーは MEM にだけ接続して、自分の電子メールにアクセスできることです。MEM は、ユーザーの LDAP 情報を通じて HTTP セッション情報とユーザープロファイルを管理することで、この機能を実現しています。2 番目のメリットは、すべての静的ファイルと LDAP 認証の状態が Messaging Server のフロントエンドに存在することです。このメリットにより、メッセージストアバックエンドからの Web ページレンダリングに関連した、CPU の追加要件が相殺されます。

MEM には、MMP と同じ特性が数多くあります。MEM は、5 つ以上のプロセッサでもスケール化を図ることができますが、ほとんどの環境では、そうしてもそれほどメリットはありません。また、将来的には、Webmail コンポーネントがメッセージストアから外され、Web サーバーのもとで Java サーブレットとして XML レンダリングを実行するアクセスレイヤーマシンに移されます。Java サーブレットは、現在 3 つ以上のプロセッサによるスケール化には対応していません。したがって、MEM 用には SPARC または Intel の 2 プロセッサマシンから選択するか、次世代のソリューションが利用可能になった時には、現在の 2 プロセッサによる MEM ハードウェアを別の用途に振り向けて小規模なマシンに交換することを想定します。

MMP と MEM は同じサーバーセット上に配置できます。そうすることのメリットとして、少数の MMP または MEM が必要な場合に、冗長性確保のために必要なハードウェアの追加を最小限に抑えることができます。MMP と MEM を同じサーバーセット上に配置することで生じる唯一のデメリットの可能性は、1 つのプロトコルに対するサービス拒否攻撃が別のプロトコルにも影響を与えることです。

ディスクストライプ幅の設定

ディスクストライピングを設定するときには、システムを通過するメッセージの平均サイズにストライプ幅を合わせます。128 ブロックのストライプ幅は、通常の使用には大きすぎて、パフォーマンスに悪影響を与えます。代わりに、8、16、32 ブロック (それぞれ 4、8、16K バイトのメッセージサイズの場合) の値を使用します。

メールボックスデータベースキャッシュサイズの設定

Messaging Server は、メールボックスデータベースの呼び出しを頻繁に行います。そのため、そのデータができるだけ迅速に返されることが重要です。メールボックスデータベースの部分をキャッシュ化すると、メッセージストアのパフォーマンスが改善されます。最適なキャッシュサイズを設定することで、メッセージストア全体のパフォーマンスを大きく向上させることができます。キャッシュのサイズは、`configutil` のパラメータ `store.dbcachesize` を使用して設定します。

メールボックスデータベースは、データページに格納されます。さまざまなデーモンにより `stored`、`imapd`、`popd` などのデータベースが呼び出されると、指定されたページがキャッシュに格納されているかどうかが、システムによりチェックされます。ページがキャッシュ内に存在する場合は、それがデーモンに渡されます。存在しない場合は、システムは 1 ページをキャッシュからディスクに書き戻し、指定されたページを読み込んでそれをキャッシュに書き込む必要があります。ディスクの書き込みと読み取り回数を減らすことはパフォーマンスの向上につながりますが、それだけに、キャッシュサイズを最適に設定することが重要となります。

キャッシュサイズが小さすぎる場合は、指定されたデータをディスクから必要以上の頻度で読み込む必要があります。キャッシュサイズが大きすぎる場合は、ダイナミックメモリ (RAM) が浪費され、ディスクとキャッシュの同期に余計な時間がかかります。これら 2 つの状況の中では、キャッシュが大きすぎる場合よりも小さすぎる場合の方が、より大きなパフォーマンスの低下を招きます。

キャッシュの効率性は、ヒットレートにより測定されます。ヒットレートは、データベースがキャッシュにより処理される時間の割合のことです。最適化されたサイズのキャッシュでは、ヒットレートは 99 パーセントに達します。すなわち、要求されたデータベースページの 99 パーセントが、ディスクから取得されることなくデーモンに返されます。設定の目標値は、要求されたデータの少なくとも 95 パーセントがキャッシュにより返されるページを、キャッシュが相当数保持できるようにすることです。キャッシュから返されるページが 95 パーセント未満の場合は、キャッシュサイズを大きくする必要があります。

キャッシュのヒットレートは、Sleepycat データベースコマンド `db_stat` を使用して測定できます。

例 :

```
# /opt/SUNWmsgsr/lib/db_stat -m -h /var/opt/SUNWmsgsr/store/mboxlist

2MB 513KB 604B  Total cache size.
1               Number of caches.
2MB 520KB       Pool individual cache size.
0               Requested pages mapped into the process' address
space.
55339             Requested pages found in the cache (99%).
```

この例では、ヒットレートは 99 パーセントです。これは、キャッシュサイズが最適であるか、大きすぎることを示します。キャッシュサイズが大きすぎる場合は常に 99 パーセントとなります。これをテストするには、ヒットレートが 99 パーセント以下になるまでキャッシュサイズを小さくしていきます。ヒットレートが 98 パーセントになったら、データベースキャッシュサイズが最適化されたことを意味します。逆に、`db_stat` が 95 パーセント未満のヒットレートを示した場合は、`store.dbcachesize` を使用してキャッシュサイズを大きくします。

注 ユーザーベースが変化すると、ヒットレートも変化します。このパラメータを定期的にチェックして、必要に応じて調整します。このパラメータの上限は Sleepycat データベースの制約による 2G バイトです。

第 4 章

ネットワークインフラストラクチャに対するニーズの決定

ネットワークインフラストラクチャは、システム構成の基盤となるもので、ネットワークの動作を生み出すサービスを構成します。Messaging Server の配備で、プロジェクトの目標を基準にネットワークインフラストラクチャを決定することで、基準化され、拡張可能なアーキテクチャを確保できます。

この章には、以下の節があります。

既存ネットワークの理解

既存のネットワークインフラストラクチャを理解して、それが配備の目標をどの程度満たすものであるかを判断する必要があります。既存のインフラストラクチャを調査することで、既存のネットワークコンポーネントをアップグレードしたり、新規のコンポーネントを購入したりする必要があるかどうかがわかります。以下の領域を調査して、既存のネットワークの完全な全体像を構築する必要があります。

1. ケーブルの長さ、グレードなどの物理的な通信リンク
2. アナログ、ISDN、VPN、T3 のような通信リンクと、サイト間で利用可能な帯域幅と待ち時間
3. 以下のサーバーの基本情報
    - ホスト名
    - IP アドレス
    - ドメインメンバー用のドメインネームシステム (DNS) サーバー

4. 以下に挙げるデバイスのネットワーク上の場所
    - ハブ
    - スイッチ
    - モデム
    - ルーターとブリッジ
    - プロキシサーバー
5. モバイルユーザーを含むそれぞれのサイトのユーザー数

この調査結果一覧を完成させた後で、プロジェクトの目標に照らしてその情報を再検討し、配備を成功させるにはどのような変更が必要であるかを判断します。

ネットワークインフラストラクチャの理解

以下の共通ネットワークインフラストラクチャコンポーネントは、配備の成否に直接影響します。

- ルーターとスイッチ
- ファイアウォール
- ロードバランサ
- ストレージエリアネットワーク (SAN)
- DNS

ルーターとスイッチ

ルーターはインフラストラクチャのネットワークを接続して、システム間の通信を可能にします。配備後のルーターの能力には余力を持たせて、プロジェクトの拡大とそれに伴う処理の増加に備える必要があります。

スイッチは、血管のようにネットワーク内のシステムを接続します。

フル稼働状態のルーターやスイッチはボトルネックとなる可能性があり、クライアントが別のネットワーク上にあるサーバーにメッセージを送信するのにかなりの時間がかかる結果となります。そのような場合には、先見性の欠如や、ルーターやスイッチをアップグレードする資金の欠乏から、そのコスト以上に個人の生産性が低下してしまうこともあります。

ファイアウォール

ファイアウォールは、ルーターとアプリケーションサーバーの間に位置し、アクセス制御を行います。ファイアウォールは本来、信頼されていないネットワーク (インターネット) から信頼済み (内部) ネットワークを保護するものです。現在ではより一般的に、外部ネットワークやインターネットなどの信頼されていないネットワークから、信頼済みまたは隔離された自己のネットワーク上のアプリケーションサーバーを保護する目的で使われています。

ルーターの設定を行うことで、ファイアウォールを通過するデータのスクリーニングを行い、ファイアウォール全体の機能が強化されます。ルーターの設定により、NFS や NIS のような好ましくないサービスをブロックし、パケットレベルのフィルタリングを使用して信頼されていないホストやネットワークからの通信をブロックできます。

さらに、インターネットまたは信頼されていないネットワークに開放されている環境に Sun サーバーをインストールするときに、アプリケーションをホストするのに必要な最小限の数まで、Solaris のインストールパッケージを減らすことができます。サービス、ライブラリ、およびアプリケーションの数を最小化することにより、保守が必要なサブシステムの数が減少し、セキュリティの向上につながります。Solaris™ Security Toolkit は、Solaris Operating Environment を最小化し、強化し、セキュアなシステムにするための、柔軟性と拡張性に富んだメカニズムを提供します。

サイトのセキュリティポリシーで、以下の問題に対する対策を考慮する必要があります。

ロードバランサ

ロードバランサを使用して、Web サーバーまたはアプリケーションサーバー全体の負荷を分散し、実行するタスクの種類に基づいて要求を分散します。さまざまな専用アプリケーションを異なるアプリケーションサーバーで使用しているような場合は、ユーザーが要求するアプリケーションの種類に応じてロードバランサを使用します。

データセンターが複数ある場合は、ロードバランサの地理的な分散も考慮する必要があります。地理的なロードバランシングにより、要求やサイトの能力、ユーザーとの距離に基づいて負荷の分散が行われます。1 つのセンターがダウンした場合は、地理的なロードバランサによりフェイルオーバー機能が提供されます。

Web ファーム上のロードバランサでは、サーバーの前とルータの後ろにロードバランサを配置して、トラフィックを適切なサーバーにルーティングします。ソフトウェアロードバランシングソリューションは、Web サーバーにインストールします。ソフトウェアによるソリューションでは、サーバーの 1 つが通常はトラフィックスケジューラとして機能します。

ロードバランシングソリューションでは、受信したパケットのヘッダと内容を読み取ることができます。これにより、ユーザーや要求の種類を含むパケット内の情報の種類別にロードバランスを行うことができます。パケットヘッダを読み取るロードバランシングソリューションにより、権限のあるユーザーを識別し、特定のタスクを処理するサーバーに要求を送ることができます。

サービスを提供しているすべてのサーバーとの間で、ロードバランサが動的な通信をどの程度行っているかを調査する必要があります。スケジューラはそれぞれのサーバーに ping を実行するか「ライブ」なエージェントをサーバー上で作成してロードデータを確認していますか。ロードバランサが TCP パケットをどのように解析しているかも調査する必要があります。そして、ロードバランサがパケットを処理するスピードにも注目します。ロードバランサの中には、他のロードバランサより効率性の高いものもあります。ロードバランサの効率性は、通常スループットで測定されます。

ストレージエリアネットワーク (SAN)

配備を成功に導くためには、ストレージシステムのデータ要件を理解することが必要です。SAN のシステムでは、ストレージをそれが使用されているサーバーから独立した形で配備されることが多くなってきています。SAN のシステムを配備することで、ストレージデバイスを再配置することなくマシンを交換することができるため、機能しなくなったサーバーの回復に要する時間を短縮することができます。

以下の質問を参考にして、SAN の導入により配備するストレージの要件が適切に達成されているかどうかを評価します。

- 読み取りと書き込みは効果的に行われていますか。
- より高速な I/O ストレージが必要ですか。ストライピングの採用は選択として最適ですか。
- 高い稼働時間率を必要としていますか。ミラーリングの採用は選択として最適ですか。
- データのバックアップはどのような方法で行いますか。バックアップはどのタイミングで行いますか。

DNS

DNS クエリの使用頻度が高いサーバーにはローカルキャッシング DNS サーバーを用意して、ルックアップによる待ち時間を短縮し、ネットワークトラフィックを減らします。

要件を決定する際には、メールストア、メールリレーイン、メールリレーアウトなどの機能別にホスト名を割り当てるようにします。すべてのホスト名が現在 1 台のマシン上でホストされている場合でも、このポリシーを考慮する必要があります。サービスをそのように構成しておくと、そのサービスを別のハードウェアに移すときに、変更に伴う影響をかなり小さくすることができます。

ネットワークインフラストラクチャレイアウトの計画

インフラストラクチャのトポロジを考えるときに、以下の視点から検討を行う必要があります。

- DMZ
- イントラネット
- 内部ネットワーク
- プロキシ

非武装地帯 (DMZ)

今日、ほとんどの企業ネットワークで DMZ が取り入れられています。DMZ により、企業ネットワークがインターネットから分離されます。DMZ は厳重に保護された領域で、Web サーバーのようなインターネットサービスと機能を提供するサーバーが配置されます。これらのマシンは、直面する攻撃に耐えられるように強化されています。そのような攻撃によりセキュリティが破られた場合のリスクを制限するために、通常これらのサーバーには内部ネットワークに関する情報が含まれていません。たとえば、ネームサーバー機能には、インターネットに接続されたサーバーとルーターしか含まれていません。

さらに進んだ DMZ では、ファイアウォールのセキュリティと機能がより強固になったことから、DMZ がファイアウォールの後ろのセグメントに移動されています。しかし、DMZ は依然として内部ネットワークからは分離されています。Web サーバー、FTP サーバー、メールサーバー、および外部 DNS をホストするすべてのマシンは、必ず DMZ セグメントに配置する必要があります。

単純なネットワーク設計では、インターネットサービス、VPN アクセス、およびリモートアクセスのための個別の DMZ セグメントだけを定義します。ただし、VPN アクセスとリモートアクセスのトラフィックにはセキュリティ上の問題が存在します。したがって、これらのタイプのトラフィックについては、それ以外のネットワークから分離された適切な接続が必要となります。

DMZ セグメントを提供するファイアウォールは、対応するサービスポートと DMZ 内でそのサービスを提供しているホストに宛てられた受信パケットだけを許可するものでなければなりません。また、DNS やメールのようなサービスを提供するマシンは、そのサービスのためにインターネットにアクセスする必要がありますが、これらのマシンに対するインターネットへの送信トラフィックを制限します。要求された接続のタイプにより、DMZ を受信専用と送信専用に分けることも 1 つの方法です。しかし、サービス拒否攻撃により DNS や電子メールサービスが妨害される可能性を考えると、受信と送信専用のサーバーに分けてこれらのサービスを提供することには検討の余地があります。電子メールベースのトロイの馬やワームにより、送信メールサーバーが制御不能に陥り、オーバーランが発生した場合でも、受信メールは受け取ることができます。DNS サーバーと同じアプローチを適用します。

イントラネット

DMZ は、インターネットへのサービスを提供するホストのためのネットワークセグメントを提供します。この設計により、内部ホストは外部からの攻撃にさらされるホストとは別のセグメントに置かれるため、保護されます。内部的には、内部ユーザーに限定された同様のサービス (Web、ファイルサーバー、内部 DNS など) を提供しています。インターネットサービスをセグメント化するのと同様に、内部サービスもセグメント化します。このような方法によるサービスの分離により、ルーターのフィルタリングでより緊密な制御を行うことができます。

インターネットに向けたサービスを DMZ で分離してセキュリティを確保したように、プライベート内部サービスも独自の内部 DMZ 内に配置する必要があります。

ネットワークのサービスとサイズによっては複数の DMZ が有用なように、複数のイントラネットも同様に有用です。

セグメントを提供するファイアウォールのルールは、DMZ のファイアウォールに使用されるものと同様に構成する必要があります。受信トラフィックは、内部メールサーバーに渡される受信メールのような DMZ からの情報をリレーするマシンと、内部ネットワーク内にあるマシンだけから送られてくるものでなければなりません。

内部ネットワーク

内部ネットワークセグメントを構成しているセグメントです。これらのセグメントには、ユーザーのマシンや部署で使用するワークステーションが含まれます。これらのマシンは、イントラネット内のホストからの情報を要求します。開発、ラボ、およびテストネットワークセグメントもこれに含まれます。各内部ネットワークセグメント間のファイアウォールを使用してトラフィックのフィルタリングを行い、部門間のセキュリティをさらに強化します。これらのセグメント上で使用される内部ネットワークトラフィックとサービスのタイプを識別して、内部ファイアウォールが有効であるかどうかを判断します。

これらのマシンに、インターネット上のマシンと直接通信することを許可すべきではありません。これらのマシンでは、DMZ 内のマシンとの直接通信を避けた方が賢明です。これらのマシンが要求するサービスがイントラネット上のホストにあれば理想的です。一方で、イントラネット上のホストは DMZ 内のホストと通信を行って、電子メールの送信や DNS などのサービスを完了することができます。このような間接的な通信であれば問題はありません。

プロキシ

DMZ 内には、インターネット上のマシンと直接通信を行うマシンだけを配置する必要があります。ユーザーがインターネットへのアクセスを要求した場合は、以前のトポロジに基づいた問題が発生します。このような場合は、プロキシが有効です。内部ネットワークか、またはさらに望ましいのはイントラネットセグメント内にプロキシを配置します。インターネットにアクセスする必要のあるマシンは、要求をプロキシに渡し、プロキシがそのマシンに代わって要求を実行します。インターネットへのこのリレーにより、マシンが直面する可能性のある危険を防ぐことができます。

プロキシはインターネット上のマシンと直接通信を行うため、DMZ 内に配置する必要があります。ただしこれは、内部のマシンが直接 DMZ 内のマシンと通信を行うのを防ぐという意図と矛盾します。この通信を間接的なものにするために、二重のプロキシシステムを使用します。イントラネット内の二次プロキシは、内部マシンの接続要求を DMZ 内のプロキシに渡し、そこでインターネットへの直接接続が行われます。

ファイアウォールの設定

通常のパケットフィルタリング機能のほかに、ファイアウォールには IP スプーフィングを防ぐ機能もあります。可能な限り IP スプーフィング保護機能を使用してください。

たとえば、インターネットから内部ネットワークへのエントリポイントが 1 つだけで、インターネットからのパケットに内部マシンの発信元アドレスがある場合、それはおそらくスプーフされたものです。ネットワークのトポロジに基づいて、内部マシンの発信元アドレスを持つパケットは、インターネットからではなく内部ネットワークから発信されたものでなければなりません。IP スプーフィングを防ぐことでこのような可能性はほとんどなくなり、IP アドレスベースの認証をすり抜けることも困難になるため、他のファイアウォールのルールを減らすことができます。内部ファイアウォールにも同様の IP スプーフィング対策を行います。

モバイルユーザー

リモートユーザーまたはモバイルユーザーに対しては、どのようなアクセス手段を提供するかを検討する必要があります。そのようなユーザーがアクセスできない手段があるでしょうか。どのようなタイプのセキュリティポリシーを必要としていますか。SSL による認証が必要ですか。また、モバイルユーザーの数にほとんど変化がないか、今後増加するのかについても検討します。

第 5 章

メッセージングトポロジの設計

アーキテクチャ設計により、ハードウェアリソースとソフトウェアリソースに Messaging Server コンポーネントをどのように配置するかが決定されます。そして、これが配備環境設計の要件決定の基礎となります。

この章では、メッセージングトポロジの設計方法について説明します。メッセージングトポロジは、ネットワーク化されたメッセージングシステムの物理的および論理的なレイアウトを示すものです。とくに、トポロジは、ネットワーク上でデバイスがどのように配置され、互いにどのようにやり取りするかを示します。さらに、ネットワークを経由してデータを配信する方法も示します。トポロジは、データフローを規定するネットワークプロトコルに結びつけられています。

この章には、以下の節があります。

地理的ニーズの理解

メッセージングトポロジ設計の最初のステップは、地理的ニーズを確認することです。特に、組織内のそれぞれの場所に必要なメッセージングサービスを決定する必要があります。

1. 配備の目標を確認したら、次に配備内のそれぞれの場所に必要な機能を決定する
2. 組織の物理的な制約、特に以下の項目について理解する
    - 使用可能な帯域幅
    - 組織内の物理的な場所間の距離

- それぞれの物理的な場所におけるメールトランザクションレートとメールストレージの量

トポロジ設計戦略の決定

トポロジを開発する前に、企業内のどこにメッセージングサービスを配置するかを決定する必要があります。目標により、組織に適用可能なトポロジには以下の 4 つがあります。

- 集中トポロジ

  ほとんどまたはすべてのシステムコンポーネントとメッセージングサーバーを 1 つの場所に統合する

- 分散トポロジ

  ほとんどまたはすべてのシステムコンポーネントとメッセージングサーバーを複数のサイトに分散する

- ハイブリッドトポロジ

  いくつかのシステムコンポーネントを統合し、その他のコンポーネントを複数の場所に分散する

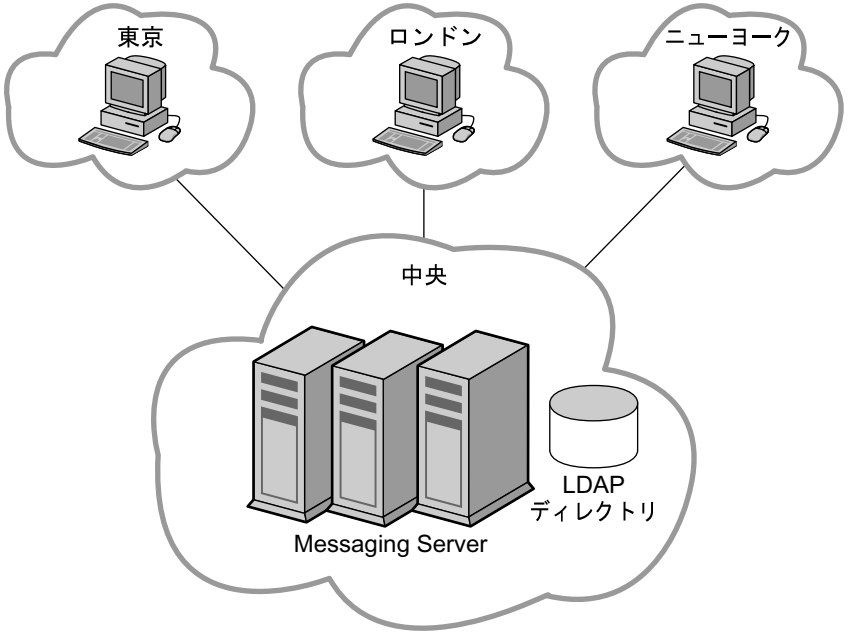
- サービスプロバイダトポロジ

  複数のドメインをホストして、より大きなカスタマベースを処理する。集中トポロジと同様に、ほとんどのシステムコンポーネントを 1 つの場所に統合する

集中トポロジ

集中トポロジでは、ほとんどまたはすべてのシステムコンポーネントとメッセージングプロセスを 1 つのサイトに配置します。リモートサイトのクライアントは、Wide Area Network (WAN) により中央メッセージングサーバーと通信を行います。77 ページの図 5-1 は集中トポロジを示します。

図 5-1　集中トポロジ



以下のような場合に、集中トポロジの導入を検討します。

- リモートサイトでのメッセージングがミッションクリティカルなものではない
- 小さなサイズのテキストメッセージの送受信を行うユーザーが多い
- 組織が 1 つの物理的な場所にあるか、または小人数のユーザーが複数の場所に分散している
- リモートサイトのサポート要員がいない
- リモートサイトと中央サイト間で、少なくとも ISDN 以上の良質な帯域幅が使用可能

集中トポロジの導入にはいくつかのメリットがあります。一般に、集中トポロジでは、ハードウェアとサポートのコストが低くなります。集中トポロジでは、単純なメッセージングアーキテクチャと少数の複製契約によるディレクトリ複製構造のため、管理が容易です。単純なアーキテクチャと地理的に離れたサイト間でインストールを調整する必要がないため、集中トポロジでは迅速な配備が可能です。

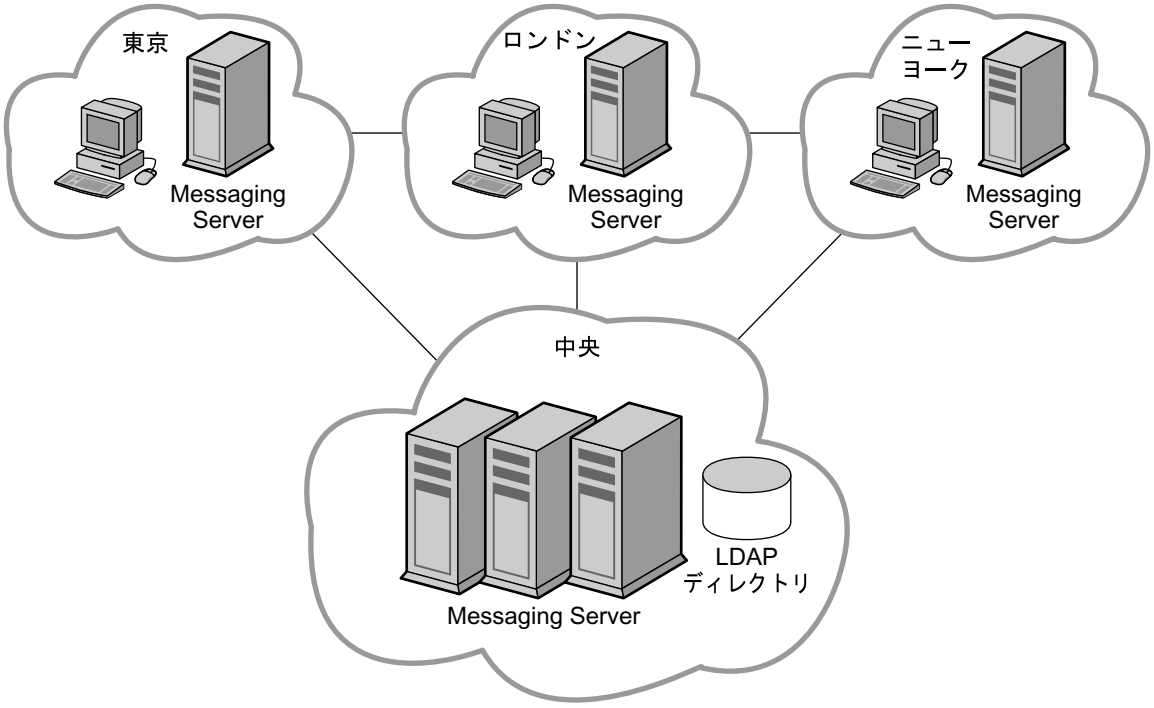
ただし、集中トポロジの実施にはメリットと等しくデメリットもあります。集中化アプローチは WAN に大きく依存しています。ネットワークが正しく機能しなくなると、同じサイトのユーザーもリモートサイトのユーザーも、共に電子メールの送信ができなくなります。ネットワークの帯域幅とトラフィックにより、使用率がピークに

達したときはサービスの処理が遅くなる場合があります。同じドメイン内にメッセージを送信するユーザーにとって、集中トポロジは非効率的となります。たとえば、77 ページの図 5-1 に示されるように、東京サイトのあるユーザーが送信したメッセージは、同じ東京サイトの別のユーザーに配信される前にまず中央サイトに送られます。

分散トポロジ

分散トポロジでは、ほとんどまたはすべてのシステムコンポーネントとメッセージングプロセスを、通常はリモートサイトとなる複数のサイトに分散配置します。以下の図は分散トポロジを示します。

図 5-2　分散トポロジ



以下のような場合に、分散トポロジの導入を検討します。

- リモートサイトでのメッセージングがミッションクリティカルなものである
- ユーザーが大量のメッセージの送受信を行う
- リモートサイトに大量のユーザーを抱えている

- リモートサイトにサポート要員がいる
- リモートサイトへの帯域幅が貧弱

  帯域幅がトポロジ戦略に大きな影響を及ぼす場合は、帯域幅のアップグレードを検討します。一般に、帯域幅は比較的安価です。Virtual Private Networking (VPN) の導入についても検討します。VPN ではファイアウォールで保護された専用線ではなく、既存の広帯域幅インターネット網を使用します。

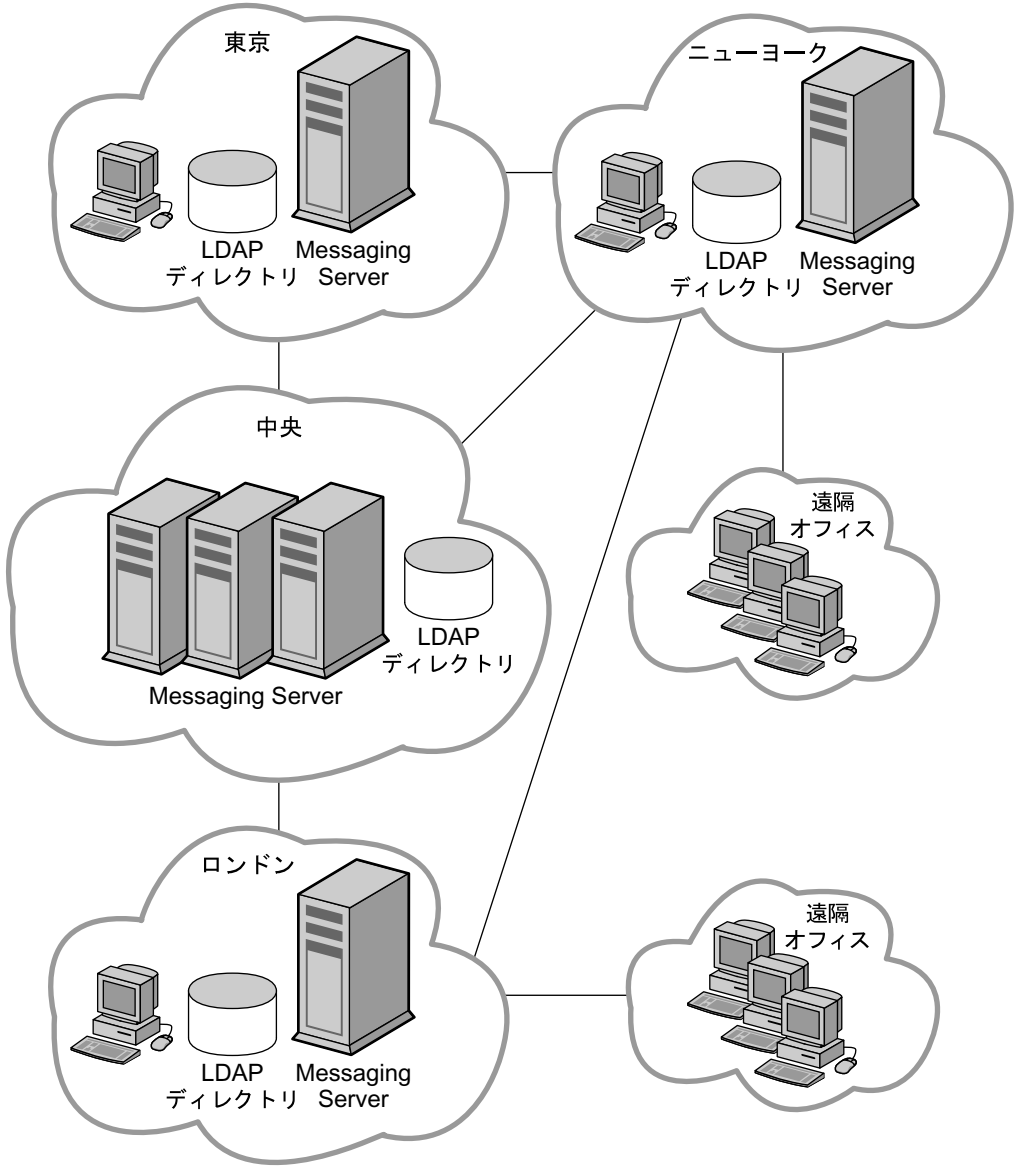
分散トポロジの導入にはいくつかのメリットがあります。メッセージを WAN 経由で取得する必要がないため、地域サイトのユーザーはメッセージに迅速にアクセスできます。さらに、場所内で送信されるメッセージのトラフィックは、集中トポロジの場合よりも少なくなります。ただし、遠隔オフィスは WAN に依存します。したがって、大量のメッセージトラフィックが遠隔オフィスで生成される場合、WAN をアップグレードする必要が出てきます。

分散トポロジを導入することのデメリットは、多くの場所で多くのハードウェアを保守しなければならないため、一般にハードウェアとサポートのコストが高くなることです。分散トポロジは複雑なため、サポートのコストも高くなります。たとえば、分散トポロジにおけるフェイルオーバーは、集中トポロジの場合よりも難しくなります。さらに、複数のサーバーを複数のサイトに分散するため、Messaging Server の初期配備に時間がかかります。

ハイブリッドトポロジ

ハイブリッドトポロジでは、集中トポロジと分散トポロジを組み合わせて、組織のニーズを満たします。80 ページの図 5-3 はハイブリッドトポロジを示します。

図 5-3　ハイブリッドトポロジ

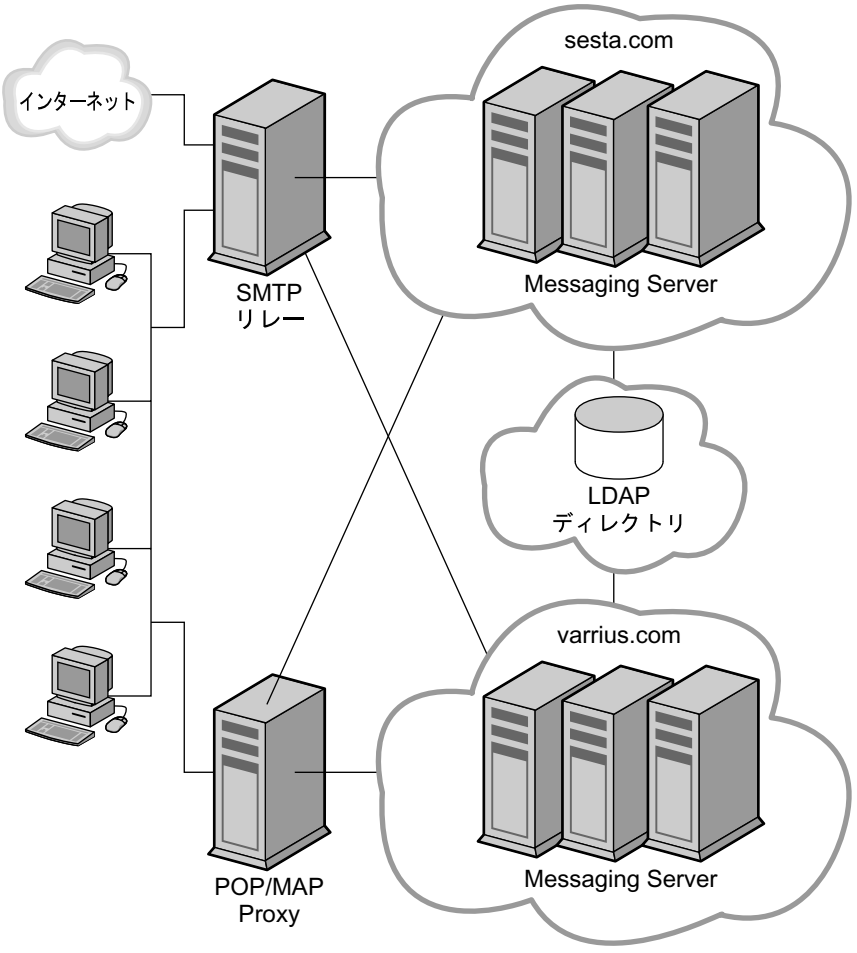


ハイブリッドトポロジからメリットを得られる組織として、大規模なユーザーベースをサポートできるサイトを数多く持つ組織があげられます。大規模なユーザーベースをサポートするサイトは、メッセージングサーバーを独自に保有できます。これらの大規模なサイトには、その近くに小規模な遠隔オフィスを持つ場合もあります。ただし、これらの遠隔オフィスには固有のメッセージングサーバーは必要ありません。代わりに最寄りの主要オフィスが、遠隔オフィスのためのサービスの中央ロケーションとして機能します。

サービスプロバイダトポロジ

サービスプロバイダトポロジは、本質的には大規模な集中トポロジです。通常、サービスプロバイダは複数のドメインをホストしており、企業よりも大規模なカスタマベースを抱えています。システムは集中化されており、ピーク時でも複数のユーザーをサポートする能力があります。82 ページの図 5-4 はサービスプロバイダトポロジを示します。

図 5-4　サービスプロバイダトポロジ



メッセージングトポロジ要素の理解

この節では、メッセージングトポロジにおける最も一般的な要素について説明します。基本的な要素について理解を深めることで、独自のトポロジの設計が容易になります。

以下のトピックについて説明しています。

メッセージングトポロジのコンポーネント

76 ページの「トポロジ設計戦略の決定」で、メッセージングトポロジの 3 つのコンポーネントとして、Messaging Server、Directory Server、およびクライアントについて簡単に説明しました。この節では、基本的なメッセージングトポロジにおけるその他のコンポーネントについて説明します。

Messaging Server : ユーザーのメールボックスを収容して管理し、インターネットリレーと MTA リレーで説明されているように、Messaging Server の MTA としても機能する

クライアント : 多くの場合 Messaging Multiplexor を通じて、Messaging Server からメッセージングサービスにアクセスする

Directory Server : Messaging Server により名前とエイリアスの検索に使用される。ダイレクト LDAP 検索によりメッセージがどこにルーティングされるかが決められる

Messaging Multiplexor : メッセージ取得のために適切なメッセージングサービスにクライアントを接続する

インターネットリレー : インターネットからファイアウォールを越えてメッセージをリレーする。通常、Messaging Server はこの機能を実行するように設定される

MTA リレー : 受信 MTA は、受信したメッセージを適切な Messaging Server 内の有効なアドレスにルーティングする。送信 MTA はクライアントから送信されたメッセージを受け取り、LDAP にクエリを行って送信先を検索し、メッセージを適切なサーバーに送信するか、ファイアウォールを越えてインターネットに向けて送信する。通常、Messaging Server はこの機能を実行するように設定される

DNS Server : サーバー名を IP アドレスに解決し、ネットワーク内の適切なアドレスにメッセージが届くようにする

ファイアウォール：内部サイトのインターネットアクセスを制限する。組織内の部門間にもファイアウォールを設置することが考えられる

メールリレー

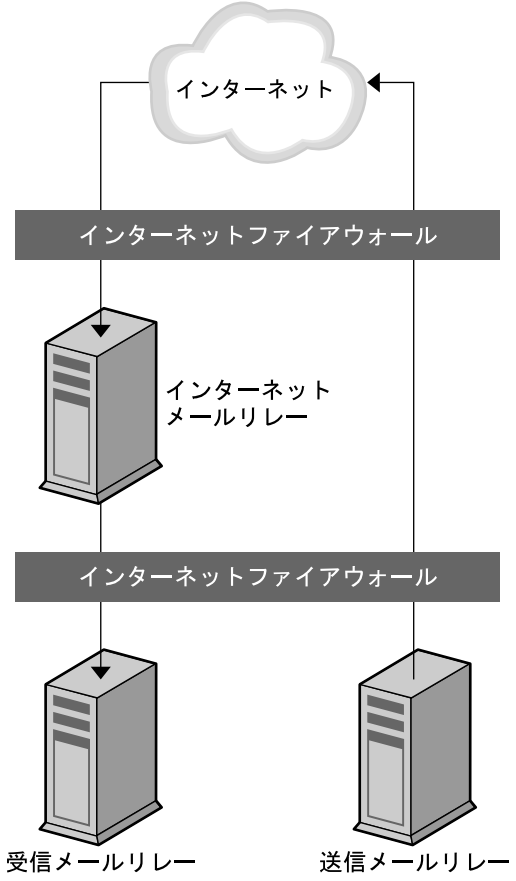
この節では、メールリレーを使用してメッセージングシステムを保護し、サイトの送受信メッセージトラフィックのフローを制御する方法について説明します。

インターネットリレーは単一点での接続で、組織外のサイトからのメッセージを受け取ります。インターネットリレーは、ファイアウォールを越えて受信 MTA に、通常は別の Messaging Server に受信メッセージを送ります。

次に、受信 MTA はディレクトリのクエリを行って、組織内のメッセージの送信先を判断します。インターネットリレーは、ファイアウォールの外部ウォールと内部ウォールの間に位置するファイアウォールの非武装地帯 (DMZ) に配置され、受信 MTA に関する情報にだけアクセスし、それ以外の情報にはアクセスしません。

送信 MTA は、クライアントから送信されたメッセージを受け取ります。送信 MTA は LDAP のクエリを行って送信先を検索し、メッセージを適切なサーバーに送信するか、ファイアウォールを越えてインターネットに向けて送信します。これにより、ユーザーのためにメッセージを取得するというメッセージングサーバーとしての機能から MTA が解放されます。85 ページの図 5-5 にこの概念を示します。

図 5-5　メッセージトポロジにおけるメールリレー

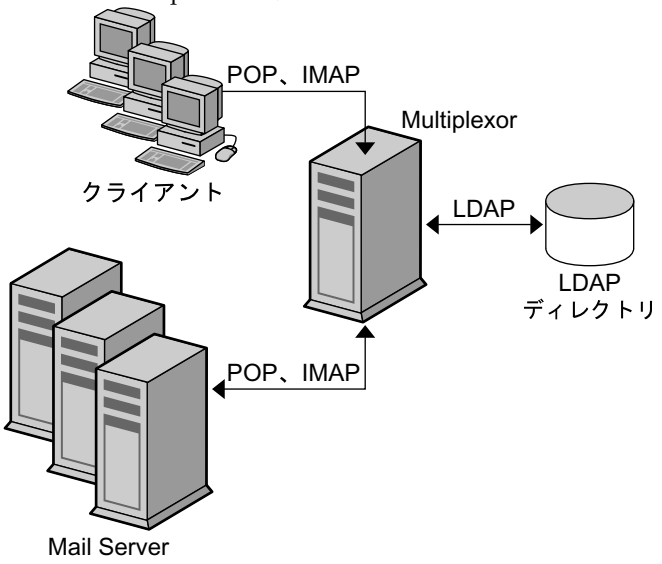


Messaging Multiplexor (MMP) および Messenger Express Multiplexor (MEM)

MMP により、Messaging Server のレイアウトをエンドユーザーから隠すことができます。その結果、メールボックスが配置されているサーバーを特定されることなく、ユーザーに汎用 MMP を割り当てることができます。メッセージアクセスクライアントは、受信メッセージを取得するときに MMP を指定します。

そのようなクライアント接続と認証の際に、MMP はディレクトリ内のユーザー情報の検索を行い、ユーザーのメッセージがどこにあるかを判断します。次に、MMP はクライアントを特定のサーバーに接続します。以下の図は、Messaging Server に対する IMAP4 と POP3 接続のプロキシとして MMP が機能する仕組みを示します。MEM 機能を使用することで、Messenger Express のような 複合 HTTP サービスを利用できます。以下の図は、Messaging Server 環境における Multiplexor の機能を示します。

図 5-6 Multiplexor の概要



ゲートウェイ

組織には、旧バージョンのメッセージングシステムがメッセージング処理の専用メソッドとして存在する場合があります。ユーザーを移行させるまで、両方のメッセージング戦略を残しておかなくてはなりません。これらの旧バージョンのシステムにアクセスする場合には、SMTP ゲートウェイを使用できます。これは、新規のシステムと旧バージョンのシステム間で SMTP 接続を有効にするものです。

メッセージングトポロジ例の作成

トポロジ上のニーズ、戦略、トポロジ要素について基本的な部分を理解すれば、メッセージングトポロジを作成できます。メッセージングトポロジの作成を容易にするために、この節では Siroe Corporation の例を使用します。

Siroe Corporation は、ニューヨークに本社を置くマルチメディア企業です。ロサンジェルスとシカゴに小さなオフィスを持ち、サンディエゴとミネアポリスに遠隔オフィスがあります。

ステップ 1: メッセージング目標の確認

トポロジ作成の最初のステップは、組織の目標を確認することです。第 2 章「要件の分析」で行ったように、Siroe のメッセージング目標を、ビジネス目標、技術的および財務的制約に分類します。

Siroe のビジネス目標

財務、マーケティング、法務、IT、エンジニアリングの各グループがニューヨークにあります。クリエイティブグループはロサンジェルスとサンディエゴにあります。テクニカルサポートグループはシカゴとミネアポリスにあります。メッセージのほとんどは、シカゴ、ロサンジェルス、ニューヨーク間で送信されています。

Siroe Corporation の従業員は、通信の主要手段を電子メールに依存しています。平均すると、従業員は 1 日に約 15 件のメッセージを送信しており、スプレッドシート、プレゼンテーション、またはアニメーション形式の添付ファイルを送信しています。

配備の計画者は、メッセージングサーバーをシカゴ、ロサンジェルス、ニューヨークに配置することを決定しました。サンディエゴとミネアポリスの電子メールトラフィックは比較的少ないため、遠隔オフィスでは、シカゴとロサンジェルスのサーバーに接続する電子メールクライアント接続だけとなります。

Siroe の財務的および技術的制約

予算上の制約により、Siroe は稼働中の既存のインフラストラクチャとハードウェアを使用し、サーバーをクリティカルなニーズのある場所に移動する予定です。24 時間年中無休のサポートは、ニューヨーク、シカゴ、ロサンジェルスのオフィスでのみ実施します。すべてのオフィスは T3 回線で接続されます。

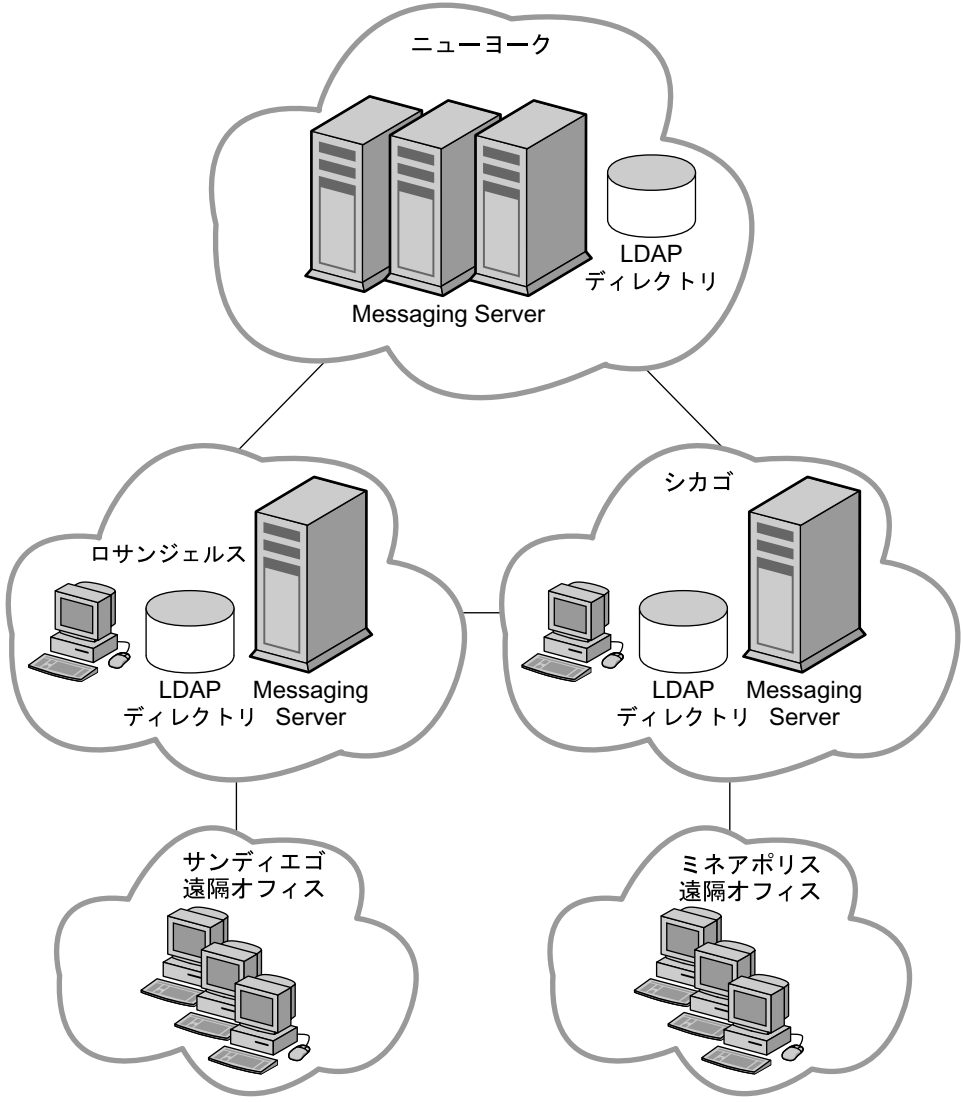
ステップ 2: トポロジ戦略の選択

メッセージングトポロジ作成の 2 番目のステップは、76 ページの「トポロジ設計戦略の決定」で説明されているトポロジ戦略の選択です。Siroe Corporation は、ビジネス目標と財務的および技術的制約の評価を行いました。その結果、以下の判断を下しました。

- メッセージングサーバーを遠隔オフィスに配置する必要はなく、メールクライアントだけとする
- 遠隔オフィスには、T3 回線による高品質の帯域幅が存在する
- 場所にかかわらず、メールユーザーは会社全体に対して大量のメッセージの送受信を行う
- ニューヨーク、ロサンジェルス、シカゴのユーザー数が多く、ミネアポリスとサンディエゴのユーザー数は少ない
- ニューヨーク、ロサンジェルス、シカゴにはサポート要員が存在する

次に、Siroe Corporation は目標と制約を一般的な設計戦略にマップしました。89 ページの図 5-7 は Siroe Corporation がハイブリッドトポロジを選択したことを示します。

図 5-7　Siroe Corporation のハイブリッドトポロジ

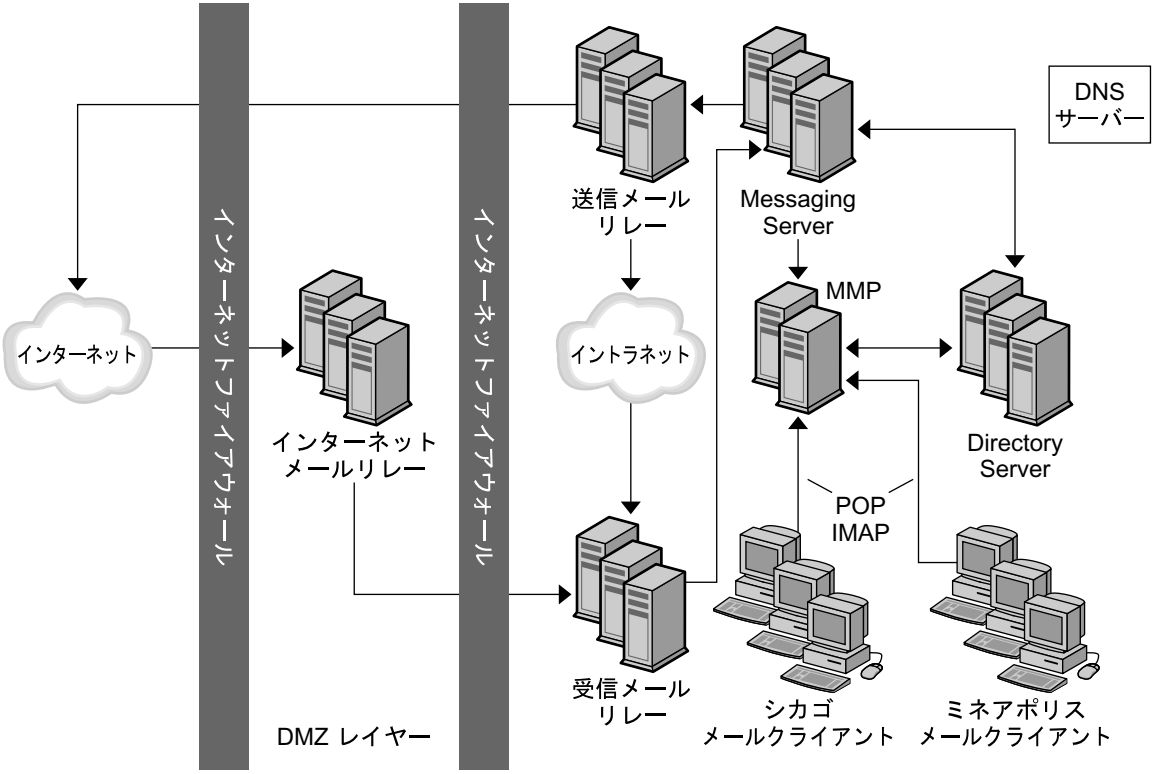


システムに対して送受信されるメッセージトランザクションのレートはニューヨークが最も高いため、Messaging Server を最も多く配置します。ニューヨークより小規模のロサンジェルスとシカゴは、サンディエゴとミネアポリスもサポートします。ただし、これらの遠隔オフィスには固有のメッセージングサーバーは必要ありません。代わりに、シカゴとロサンジェルスが遠隔オフィスのためのサービスの中央ロケーションとして機能します。

ステップ 3: トポロジ要素の計画

メッセージングトポロジ作成の最後のステップは、83 ページの「メッセージングトポロジ要素の理解」で説明されているように、実際の配備におけるトポロジ要素の計画を行うことです。以下の図は、シカゴとミネアポリスオフィスのトポロジ要素を示します。

図 5-8 シカゴとミネアポリスオフィスのための Siroe のメッセージング配備におけるトポロジ要素



作業負荷の 30 パーセントがサードパーティーのベンダと請負業者で構成されるため、トポロジ内で外部ファイアウォールに内部ファイアウォールを追加して、社内の場所へのアクセスを制限します。インターネットリレーをトポロジ内に配置し、インターネットからのメッセージをルーティングし、ファイアウォールを越えてリレーします。MTA リレーが追加され、受信メッセージと送信メッセージがルーティングされます。受信メッセージと送信メッセージを分離することにより、大量のメッセージトラ

フィックに対応できます。MMP は、従業員の POP および IMAP メールクライアントを Messaging Server 内のそれぞれのメールボックスに接続します。MMP を使用することで、従業員はログイン時に特定のメールホストを知る必要がなく、管理者は従業員のメールボックスを別のメールサーバーにシームレスに移動できます。

メッセージングトポロジを作成することで、配備におけるすべての要素の物理的および論理的配置を考慮できます。また、導入のやり直しを最小限にとどめることが可能になります。

第 6 章

サイズ決定戦略の計画

配備を計画する場合には、Messaging Server の設定方法を検討して、パフォーマンス、スケーラビリティー、および信頼性を最適化する必要があります。

サイズ決定はそのための重要な要素の 1 つです。サイズ決定のプロセスを実行することで、Messaging Server ユーザーへの作業負荷の見積もりを踏まえた、希望するレベルのサービスまたは応答時間を実現するために必要となるハードウェアとソフトウェアを確認できます。サイズ決定はインタラクティブな作業です。

この章は、メッセージング配備のサイズ決定の基礎について説明し、正しいサイズ決定データを得て配備上の判断ができるようにすることを目的としています。また、Messaging Server のサイズ決定プロセスの背景と理論的根拠についても説明します。

注	配備にはそれぞれに固有の特徴があるため、この章では特定のサイトに関するサイズ決定情報の詳細な説明はしていません。代わりにここでは、サイズ決定計画を構築する場合には何を考慮しなければならないのかを説明します。配備のハードウェアとソフトウェアのニーズを決定する場合には、ご購入先のテクニカルサポート担当者と共に作業を行ってください。

この章には、以下の節があります。

サイズ決定データの収集

この節の説明を読んで、Messaging Server のサイズ決定に必要なデータを確認してください。この節には、以下の項目があります。

ピークボリュームの判断

ピークボリュームは、1 日の特定の時間帯でメッセージングシステムにトランザクションが最も集中したときのトランザクション数です。このボリュームは、サイト間やユーザークラスの違いにより大きく異なります。たとえば、ある中規模企業のマネージャクラスでは、朝の午前 9 時から 10 時の間、昼の午後 12 時から 1 時の間、夕方の午後 5 時から 6 時の間にピークボリュームが発生します。

ピークボリュームを分析する場合には、以下のポイントを考慮します。

1. ピークがいつ発生し、どのくらい継続するかを判断する
2. ピークボリューム負荷を前提として配備のサイズを決定する

   パターンの分析が終了すれば、システムの負荷を処理しやすくし、ユーザーの求めるサービスを提供するための選択を行える
3. システムのピークボリュームを決定するときには、使用する Messaging Server がその負荷をサポートできることを確認する

使用率プロファイルの作成

正確なサイズ決定には、負荷の測定が不可欠です。使用率プロファイルにより、Messaging Server ホスト上のプログラムとプロセスが実行する要素が決定されます。

この節では、使用率プロファイルを作成して、配備で発生する負荷の量を測定する方法について説明します。

使用率プロファイルを作成するには、以下の質問に答えてください。

1. システムのユーザー数は何人ですか。

   システムのユーザー数を数える時には、メールアカウントを持ちメールシステムにログインできるユーザーだけでなく、メールアカウントを持っているが、現在システムにはログインしないユーザーも含めます。特に、以下の表に示す、アクティブなユーザーと非アクティブなユーザーとの相違点に注意します。

表 6-1 アクティブなユーザーと非アクティブなユーザー

ユーザー	説明
アクティブなユーザー	POP、IMAP、または HTTP のようなメールアクセスプロトコルを使用してメールシステムにログインしているユーザー。アクセスプロトコルの種類により、アクティブなユーザーはメールサーバーに接続していたり、接続していなかったりする たとえば、POP ユーザーはメールアカウントを開くが、メールクライアントからメールサーバーに対して確立される POP 接続は短時間で、断続的 アクティブなユーザーは、`mailuserstatus` または `inetuserstatus` のようなアクティブステータスを持ったメール属性ではありません。メール属性の詳細については、『Sun Java System Communications Services Schema Reference』を参照
非アクティブなユーザー	メールアカウントを持っているが、現在はメールシステムを使用していないユーザー

ユーザー数が 300 以下のきわめて小規模な配備の場合は、サイズ決定戦略の計画でこのプロセスを実行する必要はありません。専門のサービス担当者と作業を行い、個別のニーズについて判断します。

2. POP、IMAP、および Messenger Express クライアントがサービスにアクセスするピークボリューム時に、システムへの接続数はどのくらいになりますか。

   特に、サポートするそれぞれのクライアントアクセスサービスの並行接続、アイドル接続、ビジー接続の数に注意します。96 ページの表 6-2 でこれらの用語が定義されています。

表 6-2　クライアントアクセスサービスへの接続

接続	説明
並行接続	メールシステム上で確立される、固有の TCP 接続またはセッション (HTTP、POP、または IMAP) の数 アクティブなユーザーは複数の並行 IMAP セッションを行うことができる。一方 POP クライアントまたは Messenger Express クライアントを使用するユーザーは、クライアントごとに 1 つしか接続できない。さらに、POP 接続と Messenger Express 接続は、サーバーに接続してデータを取得し、サーバーへの接続を切断して、データの表示、ユーザー入力の受け入れを行い、そしてメールサーバーへの再接続を行うため、POP および Messenger Express クライアントアクセスサービスのアクティブなユーザーは、ある時点においてはアクティブな接続を行わずにサービスにアクセスすることも可能
アイドル接続	確立された IMAP 接続で、時々送信される `check` または `noop` コマンドを除き、メールクライアントと Messaging Server との間で情報送信を行わないもの
ビジー接続	進行中の接続。ビジー接続の例としては、メールクライアントが送信したばかりのコマンドを処理中、つまり、メールクライアントに応答を送り返している状態のメールサーバーがある

配備における並行接続の数は、以下のいずれかの方法で決定します。

a. UNIX プラットフォームで `netstat` を使用して、確立された TCP 接続数をカウントする

b. Messenger Express または IMAP クライアントアクセスサービスのユーザーの、最後のログインとログアウトの時間を取得する 詳細は、『Sun Java System Messaging Server 管理ガイド』を参照してください。

3. 大規模な配備を行う場合には、ユーザーをどのように組織化しますか。

以下の選択肢が考えられますが、これに限られません。

- アクティブなユーザーと非アクティブなユーザーをそれぞれのマシンから集めて、アクティブユーザーを集めたマシンと非アクティブユーザーを集めたマシンとに分ける

  非アクティブなユーザーがアクティブなユーザーになる場合は、そのユーザーをアクティブなユーザーのマシンに移動します。このアプローチを採用すると、アクティブなユーザーと非アクティブなユーザーを同じマシンに置いた場合よりも、必要なハードウェアを減らすことができます。

- ユーザーをサービスのクラス別に分ける

  コントリビュータ、マネージャ、エグゼクティブのユーザーを、それぞれのサービスのクラス、権限、専門サービスに応じたメールストレージ容量の割り当てを提供するマシンに分けます。

4. それぞれのメールボックスで使用されるストレージの量はどのくらいですか。

   メールボックスあたりのストレージの容量を測定するときには、指定した割り当てではなくメールボックスの実際の使用率で見積もります。

5. インターネットからどれぐらいの数のメッセージがメッセージングシステムに送信されますか。

   メッセージの数は、ピークボリューム時の 1 秒あたりのメッセージ数で測定します。

6. ユーザー別ではどれぐらいの数のメッセージが送信されますか。

   - メールシステムのエンドユーザーに対して送信される数
   - インターネットに対して送信される数

   このメッセージの数も、ピークボリューム時の 1 秒あたりのメッセージ数で測定します。

7. 異なるサイズ範囲では、配信分布状態はどのようになっていますか。

   例 :

   - 5K バイト未満
   - 5K バイト以上 10K バイト未満
   - 10K バイト以上 100K バイト未満
   - 100K バイト以上 100K バイト未満
   - 500K バイト以上 10M バイト未満
   - 10M バイト以上

   配信されるメッセージのサイズがわからない場合は、メールシステムの平均のメッセージサイズを使用しますが、これはサイズの範囲がわかる場合ほど有効ではありません。

   メッセージのサイズは、MTA の配信レート、メッセージストアへの配信レート、およびメッセージ取得のレートに影響を与えるため、特に重要なものです。

8. Secure Sockets Layer (SSL) を使用しますか。使用する場合は、ユーザーの何パーセントが、またどのようなタイプのユーザーが使用しますか。

   たとえば、ある組織では、ピーク時間中に IMAP 接続の 20 パーセントで SSL が使用されます。

9. ウィルススキャニングまたはその他の専用のメッセージ処理を使用し、その処理をすべてのユーザーに適用しますか。

   Messaging Server の設定により、MTA は専用の処理で指定された基準に一致するすべてのメッセージをスキャンする必要があり、その結果システムの負荷が増大します。

これらの質問に答えることで、配備のための、準備段階としての使用率プロファイルが完成します。Messaging Server のニーズの変更に応じて、この使用率プロファイルにも修正を加えます。

その他の質問

以下の質問は使用率プロファイルの作成に使用できるものではありませんが、配備のサイズ決定戦略には重要なものです。これらの質問にどのように答えるかによって、ハードウェアの追加を検討しなければならなくなる場合もあります。

1. 配備にどの程度の冗長性を持たせますか。

   たとえば、高可用性の実現を考えている場合です。

2. どのようなバックアップ戦略と回復戦略 (障害回復、メールボックスの回復、サイトのフェイルオーバーなど) を実行しますか。回復タスクが完了するまでにどのくらいの時間を予想しますか。

ユーザーベースの定義

ユーザープロファイルの作成が完了したら、次にそれをこの節で説明されているユーザーベースの例と比較してみます。ユーザーベースは、ユーザーが送受信するメッセージサイズの範囲と、ユーザーが実行するメッセージング操作のタイプで構成されます。メッセージングユーザーは、5 つのユーザーベースに分類されます。

- 軽量級の POP ユーザー
- 重量級の POP ユーザー
- 軽量級の IMAP ユーザー
- 標準的な IMAP ユーザー
- 標準的な Messenger Express ユーザー

この節のユーザーベースの例では、ユーザーの行動を幅広く一般化しています。実際のユーザープロファイルは、このユーザーベースとは多少異なるかもしれません。これらの差異は、負荷シミュレータを実行する時 (100 ページの「負荷シミュレータ」を参照) に調整できます。

軽量級の POP ユーザー

軽量級の POP ユーザーベースは、一般に、簡単なメッセージング要件を持つ家庭のダイアルアップユーザーで構成されます。それぞれの並行クライアント接続は、1 時間あたり約 4 件のメッセージを送信します。これらのユーザーは、1 回のログインセッション中にすべてのメッセージの読み取りと削除を行います。さらに、これらのユーザーは 1 回の受信では、自分のメッセージの作成と送信をほとんど行いません。メッセージの約 80 パーセントが 5K バイト以下のサイズで、約 20 パーセントが 10K バイト以上です。

重量級の POP ユーザー

重量級の POP ユーザーベースは、高速ブロードバンドのユーザーか小規模な企業のアカウントであるのが一般的で、軽量級の POP ユーザーベースより高度な要件を持っています。このグループは、ケーブルモデムか DSL を使用してサービスプロバイダに接続します。それぞれの並行クライアント接続は、1 時間あたり約 6 件のメッセージを送信します。メッセージ受信者数の平均は 1 メッセージあたり約 2 人です。メッセージの 65 パーセントが、5K バイト以下のサイズです。このユーザーベースのメッセージの 30 パーセントが、5K バイトから 10K バイトの間のサイズです。5 パーセントのメッセージが 1M バイトを超えるサイズです。ユーザーのうち、85 パーセントが読んだ後ですべてのメッセージを削除しています。ただし、15 パーセントのユーザーは、メッセージをサーバー上に残したまま数回のログインを行ってから、メッセージを削除しています。メールは、これらのメールボックスのわずかな割合を占めるだけです。同じメッセージがサーバーから数回取得される場合もあります。

軽量級の IMAP ユーザー

軽量級の IMAP ユーザーベースは、高速なブロードバンドインターネットサービスを利用するユーザーで代表されます。このユーザーが利用するサービスには、メッセージ検索やクライアントフィルタのような高度なメッセージングシステム機能のほとんどが含まれます。このユーザーベースは、メッセージのサイズ、受信者の数、それぞれの並行接続別の送受信メッセージ数に関して、重量級の POP ユーザーベースに類似しています。軽量級の IMAP ユーザーは一般的に、一度のログインでセッションを数時間継続し、ログアウトする前にほとんどまたはすべてのメールを削除します。その結果、ログインセッション中にメールが蓄積されますが、通常はメールボックスに 20 から 30 件以上のメッセージが蓄積されることはありません。ほとんどの受信ボックスで、残っているメッセージの数は 10 件以下です。

標準的な IMAP ユーザー

標準的な IMAP ユーザーベースは、高度な企業ユーザーに代表され、営業日にはログインセッションがほぼ 1 日継続します。これらのユーザーは、大量のメールの送受信と保管を行います。さらに、これらのユーザーの場合、メッセージの割り当ては無制限か、またはかなり大きなものとなります。受信ボックスには大量のメールが 1 日中

蓄積されていき、溢れそうになったときにはすべて、または一部が消去されます。メッセージは定期的にフォルダに整理され、1 時間に何度かの割合で検索されます。それぞれの並行クライアント接続は、1 時間あたり約 8 件のメッセージを送信します。このカテゴリのユーザーの場合、送信する 1 件のメッセージの平均受信者数は 4 人で、メッセージのサイズは重量級の POP ユーザーおよび軽量級の IMAP ユーザーのベースとほぼ同じです。

標準的な Messenger Express ユーザー

標準的な Messenger Express ユーザーベースは、標準的な IMAP ユーザーに似ています。このユーザーベースのメッセージのサイズは、標準的な IMAP、軽量級の IMAP、および 重量級の POP ユーザーと同じです。メッセージの配信頻度も標準的な IMAP ユーザーと同じです。

組織内で、特に複数のクライアントアクセス手段を提供する場合は、おそらく複数のユーザーベースを持つことになります。ユーザーベースをこれらのカテゴリの中から決定したら、使用率プロファイルと「負荷シミュレータ」で説明されている負荷シミュレータを使用して、そのユーザーベースのテストを行います。

負荷シミュレータ

Messaging Server のパフォーマンスを測定するには、ユーザーベース (98 ページの「ユーザーベースの定義」を参照) とユーザープロファイル (94 ページの「使用率プロファイルの作成」を参照) を負荷シミュレータに入力します。

負荷シミュレータは、ピークボリューム環境を作り出し、サーバーにかかる負荷の量を調整します。これにより、システムに過負荷をかけることなく希望する応答時間を実現するには、ハードウェア、スループット、または配備のアーキテクチャを変更する必要があるかどうかを判断できます。

➤ 負荷シミュレータを使用するには

1. テストするユーザーベース (軽量級の IMAP など) を定義します。

   必要に応じて、使用率プロファイルに最適化するように個別のパラメータを調整します。

2. テストするハードウェアを定義します。
3. 負荷シミュレータを実行し、ユーザーベースを使用してテストされたハードウェアの最大並行接続数を測定します。
4. 結果を記録して、稼働中の配備の結果と比較します。
5. ピーク負荷状態の応答時間が組織で容認されるレベルになるまで、さまざまなユーザーベースとハードウェアを使用してこのプロセスを繰り返します。

注	推奨負荷シミュレータとサポートについては、ご購入先の専門サービス担当者に連絡してください。

システムパフォーマンスの評価

負荷シミュレータを使用してハードウェアとユーザーベースの評価を行うと、システムパフォーマンスを測定する必要があります。以下のトピックで、システムの全体的なパフォーマンスを向上させる方法について説明します。

- メモリの使用率
- ディスクのスループット
- ディスク容量
- ネットワークスループット
- CPU リソース

メモリの使用率

配備で使用するそれぞれのマシンに、適切な量の物理メモリが搭載されていることを確認してください。物理メモリを追加するとパフォーマンスが向上し、ピークボリューム時でもサーバーが適切に動作するようになります。メモリが不足していると、Messaging Server で過剰なスワッピングが発生し、効率的な動作が行われません。

1 つの CPU について、少なくとも 1G バイトのメモリを搭載してください。ほとんどの配備で、UltraSPARC® III システムの CPU 1 つについて 2G バイトのメモリが必要になります。

ディスクのスループット

ディスクのスループットとは、システムでメモリからディスクに、またはディスクからメモリに転送されるデータ量のことです。このデータ転送レートは、Messaging Server のパフォーマンスに重大な影響を及ぼします。システムのディスクスループットを向上させるには、以下のことを考慮します。

- 保守作業を検討し、バックアップのための十分な帯域幅があることを確認します。特にリモートバックアップの場合は、ネットワーク帯域幅にも影響を与えます。プライベートバックアップネットワークは、より効率的な代替バックアップ手段となります。
- ストアのパーティションと、`tmp` や `db` のようなストアデータ項目 の分割を慎重に行って、スループットを向上させます。
- 大規模な配備では、ユーザーベースが必ず RAID (Redundant Array of Independent Disks) 環境全体に分散されるようにします。
- ディスクからデータを取得する操作のスピードを向上させるために、データを複数のディスクでストライピングします。
- RAID がハードウェア上に存在しない場合は、RAID のサポートに十分な CPU リソースを割り当てます。

ディスク I/O を帯域幅ではなく、IOPS (1 秒あたりの I/O の合計) で測定したい場合もあります。システムがきわめて短い応答時間 (10 ミリ秒未満) で処理できる、個別のディスクトランザクションの数を測定する必要があります。

ディスク容量

サーバーシステムのディスク容量を計画する場合は、環境ソフトウェア、Messaging Server ソフトウェア、メッセージの内容、およびトラッキングの運用も含めて容量を確保する必要があります。可用性が要求される場合には、必ず外部ディスクアレイを使用します。ほとんどのシステムで、内部システムディスクでは 4 台までのディスクしかサポートされないため、パフォーマンスを向上させるには外部ディスクが必要となります。

さらに、ユーザーディスク容量を割り当てます。この容量は、通常、サイトのポリシーに従って決定されます。

ネットワークスループット

ネットワークスループットは、一定時間内にクライアントアプリケーションとサーバー間のネットワークで転送可能なデータ量のことです。ネットワークに接続されたサーバーがクライアントからの要求に応答できない場合、通常クライアントは要求の再送信を何度も行います。再送信のたびに、システムにはオーバーヘッドと余分なネットワークトラフィックが生じます。

データの完全性とシステムのパフォーマンスを向上させて、ネットワークの混雑を解消することで、再送信の数を減らすことができます。

- ボトルネックを解消するには、ネットワークインフラストラクチャが負荷を処理する能力を確保します。
- ネットワークを分割します。たとえば、Siroe Company では、クライアントアクセスに 100Mbps のイーサネットを、バックボーンに 1G バイトイーサネットを使用していました。
- 将来の拡張に備えて十分な容量を確保するには、ネットワークを構築するときに理論最大値を使用してはなりません。
- トラフィックのフローを異なるネットワークパーティションに分割して衝突を減らし、帯域幅の使用を最適化します。

CPU リソース

メッセージストア、MTA、および複合サービス (MMP および Messenger Express Multiplexor) だけを実行しているシステムに対して、十分な CPU を用意します。さらに、使用を計画している RAID システムにも十分な CPU を用意します。

アーキテクチャ戦略の構築

システムパフォーマンスのニーズを確認したあと、Messaging Server 配備のサイズ決定で次のステップは、アーキテクチャの決定に基づいて特定のコンポーネントのサイズを決定することです。

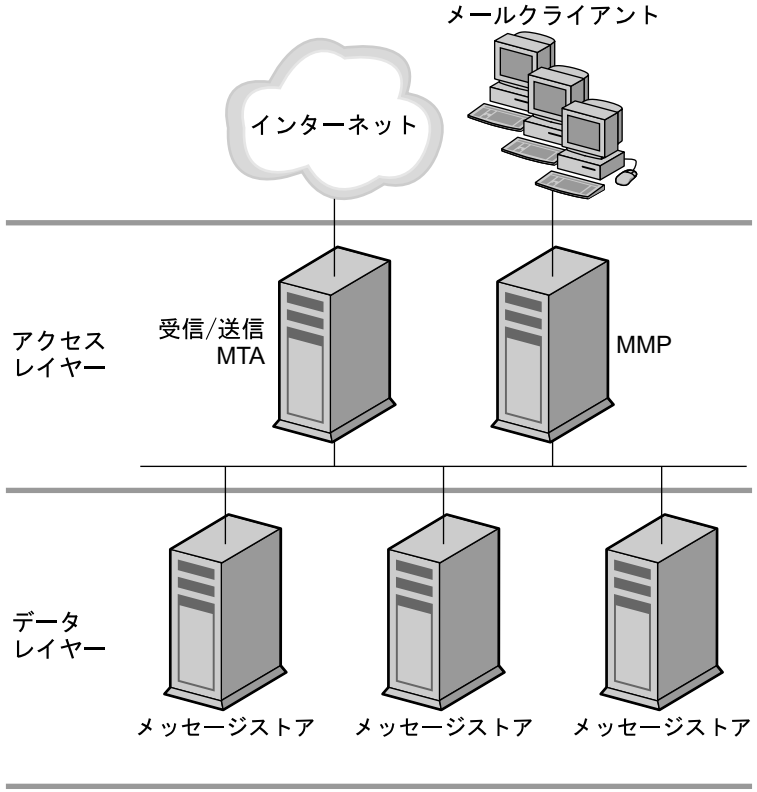
以下の節で、2 階層と 1 階層のアーキテクチャを配備する場合のサイズ決定で考慮しなければならないことについて説明します。

注 アーキテクチャ計画の詳細については、第 3 章「メッセージングアーキテクチャの開発」を参照してください。

2 階層アーキテクチャ

2 階層アーキテクチャでは、Messaging Server 配備を アクセスレイヤーとデータレイヤーの 2 つのレイヤーに分割します。簡略化した 2 階層アーキテクチャでは、MMP と MTA をアクセスレイヤーに追加します。MMP が POP と IMAP メールリーダーのプロキシとして機能し、MTA が送信されたメールのリレーを行います。データレイヤーには、メッセージストアと Directory Server を配置します。以下の図は、簡略化した 2 階層アーキテクチャの説明です。

図 6-1　簡略化した 2 階層アーキテクチャ



1 階層アーキテクチャに対して、2 階層アーキテクチャには、サイズの決定に影響を与えるいくつかのメリットがあります。2 階層アーキテクチャでは以下のことが可能です。

- 1 階層アーキテクチャよりも保守が簡単
- SSL、ウィルススキャニング、メッセージ再処理、サービス拒否攻撃のような負荷の高いプロセスをはずすことができる
- 限られた停止時間内で、拡張管理とシステムのアップグレードが容易

次のいくつかの節で、2 階層アーキテクチャにおける特定のコンポーネントのサイズ決定について説明します。

➤ メッセージストアのサイズ決定

Message Store のサイズ決定の目的は、ストアが処理可能な最大並行接続数を確認し、1 秒間にストアに配信されるメッセージの数を決定することです。

1. 負荷シミュレータを使って集めた数値をもとに、マシン 1 台あたりのストアマシン数と並行接続数を決定します。サイズ決定ツールの詳細については、100 ページの「負荷シミュレータ」を参照してください。
2. それぞれのストアマシンに必要なストレージの容量を決定します。
3. バックアップとファイルシステム回復時の復元で必要な場合は、複数のストアパーティションまたはストアマシンを使用します。

ご購入先の専門サービス担当者に、メッセージストアのユーザーの推奨最大数を尋ねてください 推奨数値を得るには、以下の点を理解しておく必要があります。

- 使用率のパターン (100 ページの「負荷シミュレータ」を参照)
- 配備内のハードウェアすべてのアクティブなユーザーの最大数
- バックアップ、復元、回復に要する時間。これらの時間は、メッセージストアのサイズが大きくなるにつれて長くなる

➤ 受信および送信 MTA ルーターのサイズを決定するには

一般的には、MTA サービスは受信サービスと送信サービスとに分けます。次に、同じ方法でそれぞれのサイズを決定できます。ルータのサイズ決定の目的は、1 秒間にリレーできるメッセージの最大数を決定することです。

受信ルーターのサイズを決定するには、実稼働環境での MTA 受信ルーターの raw パフォーマンスを知る必要があります。

1. 受信ルーターの raw パフォーマンスをもとに、SSL、ウィルススキャニングプロセス、その他の臨時のメッセージ処理を追加します。
2. 1 日のピークボリューム時のサービス拒否攻撃についても考慮します。
3. 十分な量のルーターを追加して、必要に応じた負荷の分散と冗長性を確保します。

   冗長性を持たせることで、1 つ以上のタイプのマシンで、スループットや応答時間に実質的な影響を与えることなくピーク負荷を処理できます。

さらに、一時的なメッセージの量に対して十分なディスク容量を計算して、ネットワーク上の問題やリモート MTA の機能停止に備えます。

➤ 複合サービスのサイズを決定するには

MMP と MEM のサイズを決定する場合には、システム負荷、特に MMP に対する POP と IMAP の並行接続数と、MEM に対する HTTP 接続数に基づいて計算を行います。

注	ここでの手順は、MEM と MMP が同じマシンにインストールされていることを前提にしています。

さらに、以下のことを実行する必要があります。

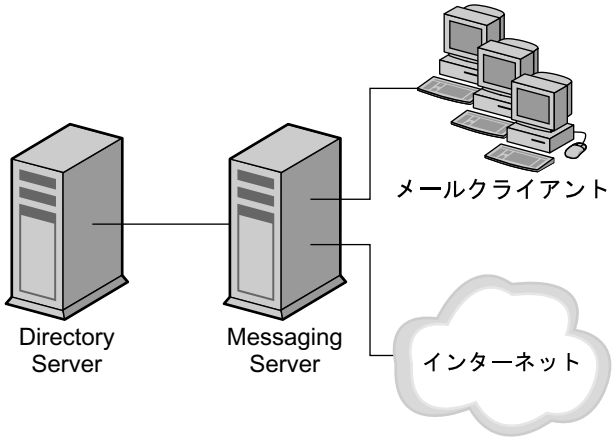
1. 必要に応じて、MMP と MMP の SSL 用に CPU またはハードウェアアクセラレータを追加します。
2. マシンに MEM を設定している場合は、メモリを追加ます。
3. MMP の SMTP プロキシにディスクを追加します。
4. サービス拒否攻撃について考慮します。
5. 必要に応じて、負荷分散と冗長性の能力を追加します。

   受信 MTA ルーターの場合と同様に、配備に冗長性を持たせることで、1 つ以上のタイプのマシンで、スループットや応答時間に実質的な影響を与えることなくピーク負荷を処理できます。

1 階層アーキテクチャ

1 階層アーキテクチャでは、アクセスレイヤーとデータレイヤーの分割がありません。MTA、メッセージストア、場合によっては Directory Server が 1 つのレイヤーにインストールされます。107 ページの図 6-2 は 1 階層アーキテクチャを示します。

図 6-2　簡略化した 1 階層アーキテクチャ



1 階層アーキテクチャでは、2 階層アーキテクチャよりもハードウェアの初期コストが低くなります。しかし、1 階層アーキテクチャを選択した場合は、保守のためにかなりの停止時間を見込んでおく必要があります。

➤ 1 階層アーキテクチャのサイズを決定するには

1. 104 ページの「2 階層アーキテクチャ」でのサイズ決定と同様に、メッセージストアのサイズを決定します。
2. 必要に応じて SSL 用の CPU を追加します。
3. サービス拒否攻撃について考慮します。
4. SMTP 接続数の増加に対応してディスクを追加します。
5. 送信 MTA ルーター用のディスクを追加します。

注　1 階層アーキテクチャおよび 2 階層アーキテクチャにおけるメッセージングコンポーネントのサイズ決定に関する特別な手順については、専門サービス担当者に連絡してください。

第 7 章

Messaging Server スキーマとプロビジョニングオプションの理解

この章では、Messaging Server のスキーマとプロビジョニングオプションについて説明します。Messaging Server のプロビジョニングは複雑であるため、製品をインストールする前に、オプションについて理解しておく必要があります。

この章には、以下の節があります。

メッセージングスキーマの選択

この節では、Messaging Server でサポートされている 2 つのスキーマと、使用する場合の選択方法について説明します。

注

Sun Java System LDAP Schema バージョン 1 から Sun Java System LDAP Schema バージョン 2 への移行方法については、『Sun Java System Communications Services Schema Migration Guide』の「`commdirmig` command」を参照してください。

スキーマ 1 のインストールとプロビジョニングのサポートは、以降のリリースから廃止されます。ただし、独自のプロビジョニングツールを持つ顧客は、LDAP スキーマ 1 を引き続き使用できます。

使用するスキーマの選択

プロビジョニングのニーズにより、インストールに最適なスキーマを選択します。

- Sun Java System Portal Server または Sun Java System Identity Server のような、シングルサインオン機能を持つその他の Java Enterprise System コンポーネント製品と Messaging Server を統合していますか。

  答えが「はい」の場合は、スキーマ 2 を使用します。

- Messaging Server を初めてインストールしますか、それとも古いバージョンからのアップグレードですか。

  Messaging Server を初めてインストールする場合は、スキーマ 2 を使用します。

  Messaging Server の古いバージョンからのアップグレードの場合は、スキーマ 1 または 2 のどちらも使用できます。

- プロビジョニングのニーズとして、インタフェースの使用がありますか。インタフェースを使用する場合、グラフィカルインタフェースですか、コマンド行インタフェースですか。

  グラフィカルユーザーインタフェースを使用する必要がある場合、またはエンドユーザーにグラフィカルインタフェースユーザーを使用したプロファイルの変更を許可する場合は、スキーマ 1 を使用します。ただし、このオプションは、新しくインストールした Messaging Server では使用できないことに注意してください。このオプションは、Messaging Server 6 がインストールされた環境での既存の Messaging Server 5.x でのみ使用できます。

  コマンド行インタフェースを使用する場合は、既存の Messaging Server でスキーマ 1、既存または新しくインストールした Messaging Server でスキーマ 2 を使用できます。

LDAP スキーマ 1

LDAP スキーマ 1 は、組織ツリーと DC ツリーで構成されるプロビジョニングスキーマです。このスキーマ (ここでは単に「スキーマ」と呼ぶ) は、以前の Messaging Server 5.x バージョンでサポートされていました。

Messaging Server がユーザーエントリまたはグループエントリを検索するときは、DC ツリーのユーザーまたはグループのドメインノードを見て、`inetDomainBaseDN` 属性の値を抽出します。この属性には、実際のユーザーまたはグループエントリの組織サブツリーへの DN 参照があります。

以前のバージョンの Messaging Server がインストールされているサイトでのみ、スキーマ 1 を使用します。

注	将来 Messaging Server にその他の Sun Java System 製品を統合してインストールする場合は、スキーマ 2 への移行が必須となります。

LDAP スキーマ 1 がサポートするプロビジョニングツール

スキーマ 1 は、Sun™ ONE Delegated Administrator for Messaging と LDAP プロビジョニングツールをサポートしています。詳細は、112 ページの「Messaging Server プロビジョニングツールの理解」を参照してください。

スキーマ 2 (ネイティブモード)

スキーマ 2 は新しく定義された一連のプロビジョニング定義で、Directory Server LDAP を使用してエントリとして格納できる情報のタイプを定義しています。

ネイティブモードでは、検索テンプレートを使用して LDAP Directory サーバーを検索します。ドメイン検索テンプレートによりドメインが検索されると、次にユーザーまたはグループ検索テンプレートにより、特定のユーザーまたはグループが検索されます。

Messaging Server を初めてインストールし、2 つのツリープロビジョニングモデルに依存したその他のアプリケーションをマシンにインストールしていない場合は、ネイティブモードを使用します。Java Enterprise System 製品群にその他の製品をインストールする場合も、このモードを使用する必要があります。

スキーマ 1 を使用する既存の Messaging Server 5.x があり、Messaging Server とその他の Java Enterprise Server と統合したい場合は、Messaging Server 6 に移行した後で、ディレクトリをスキーマ 2 に移行させる必要があります。LDAP スキーマバージョン 1 から LDAP スキーマバージョン 2 への移行方法の詳細については、『Sun Java System Communications Services Schema Migration Guide』を参照してください。

注	Java Enterprise System 製品群のすべての Sun Java System products で、スキーマ 2 ネイティブモードをプロビジョニングモデルとして使用するようお勧めします。

LDAP スキーマ 2 がサポートするプロビジョニングツール

スキーマ 2 は、Sun Java System Communications Services ユーザー管理ユーティリティをサポートしています。詳細は、112 ページの「Messaging Server プロビジョニングツールの理解」を参照してください。

スキーマ 2 互換モード

スキーマ 2 互換モードは、スキーマ 1 とスキーマ 2 ネイティブモードとの中間のモードです。スキーマ 2 互換モードは両方のスキーマをサポートしており、すでに保有している既存の 2 つのツリー設計を維持できます。スキーマ 2 互換モードは、Messaging Server をインストールする前に、Identity Server をインストールしていることが前提となっています。

スキーマ 1 を必要とする既存のアプリケーションがある場合には、スキーマ 2 互換モードを使用します。ただし、Identity Server やシングルサインオン機能などのように、スキーマ 2 を要求する機能も必要です。

注 スキーマ 2 互換モードは、スキーマ 2 ネイティブモードへの移行の便宜を提供するためのものです。最終的なスキーマ選択では、スキーマ 2 互換モードを使用しないでください。スキーマ 1 からスキーマ 2 互換モードへ移行してから最終的にスキーマ 2 ネイティブモードへと移行するプロセスは、スキーマ 1 からスキーマ 2 ネイティブモードへの単純な移行よりも複雑です。詳細は、『Sun Java System Communications Services Schema Migration Guide』を参照してください。

Messaging Server プロビジョニングツールの理解

サポートされている Messaging Server プロビジョニングツールを使用して、LDAP ディレクトリでユーザー、グループ、ドメインエントリ情報のクエリ、変更、追加、または削除を行うことができます。この節では、これらの Messaging Server プロビジョニングツールを検証します。

110 ページの「使用するスキーマの選択」にある質問の他に、114 ページの表 7-1 を使用して、スキーマとプロビジョニングツールオプションを評価します。

注 Messaging Server のインストールと設定を行う前に、Messaging Server エントリのプロビジョニングのためのスキーマモデルとツールを決定する必要があります。

以下の節で、サポートされているプロビジョニングツールに関する高度な情報について説明します。

- Sun ONE Delegated Administrator for Messaging
- LDAP プロビジョニングツール
- ユーザー管理ユーティリティ

Sun ONE Delegated Administrator for Messaging

Sun ONE Delegated Administrator for Messaging には、プロビジョンユーザーとグループのためのコマンド行ユーザーインタフェースとグラフィカルユーザーインタフェースがあります。Delegated Administrator は、プロビジョニング定義の Messaging Server 5.x バージョンである Sun LDAP スキーマ 1 を使用します。

LDAP プロビジョニングツール

スキーマ 1 ユーザーとグループは、LDAP Directory ツール (スキーマ 2 はサポートされていない) を使用してプロビジョニングを行います。Delegated Administrator のグラフィカルおよびコマンド行インタフェースとは異なり、ユーザーインタフェースを使用せずに、LDAP を通じて LDIF レコードの追加、削除、変更を行うことで、ダイレクトにユーザーとグループのプロビジョニングを行います。

ユーザー管理ユーティリティ

Sun Java System Identity Server はスキーマ 2 を使用します。Java Enterprise System 製品群の Sun Java System コンポーネント製品がスキーマ 2 を使用するため、Communications Services 6 ユーザー管理ユーティリティを使用します。これは、複数の Java Enterprise System 製品を使用している場合や、Messaging Server を新しくインストールする場合に、特に該当します。

注	Identity Server をインストールしても、Messaging Server とのグラフィカルユーザーインタフェースの互換性はありません。したがって、インタフェースを使用してユーザーとグループのプロビジョニングを行う場合には、ユーザー管理ユーティリティのみを使用できます。

インストール方法については、『Sun Java System Communications Services User Management Utility Administration Guide』を参照してください。

プロビジョニングツールオプションの比較

114 ページの表 7-1 に、サポートされているさまざまなスキーマ、プロビジョニングツール、プロビジョニングの制限、および詳細についての参考マニュアルがあります。

表 7-1　Messaging Server のプロビジョニングメカニズム

サポートされているプロビジョニングツール	プロビジョニングツールの機能	プロビジョニングツールの制限	詳細情報
Sun ONE Delegated Administrator for Messaging グラフィカルユーザーインタフェース 使用スキーマ : スキーマ 1	ユーザー、グループ、ドメイン、およびメーリングリストの管理者のためのグラフィカルユーザーインタフェースを提供。エンドユーザーは休暇メッセージとフィルタを管理	• Messaging Server 6 にアップグレードしている既存の Messaging Server 5.x の顧客だけが使用可能 • Sun ONE Web Server 6.0 (Messaging Server 5.2 バンドルとしてのみ入手可能) でのみ使用可能。Sun ONE Web Server 6.1 では使用不可 • Sun Schema 2 およびその他の Java Enterprise System 製品との互換性なし • Sun Java System Messenger Express によるメールフィルタの使用は不可。Delegated Administrator によるフィルタの使用が必要 • Messaging Server 5.2 製品でのみ使用可能なオートリレーチャネルの使用が必要	Sun ONE Delegated Administrator for Messaging 1.3 マニュアルを参照 Sun ONE Delegated Administrator インタフェースのインストールと管理方法を説明
Sun ONE Delegated Administrator for Messaging コマンド行インタフェース 使用スキーマ : スキーマ 1	ユーザー、グループ、ドメイン、およびメーリングリストの管理者のためのコマンド行インタフェースを提供	• Sun Schema 2 およびその他の Java Enterprise System 製品との互換性なし	Sun ONE Delegated Administrator for Messaging 1.3 マニュアルを参照 Sun ONE Delegated Administrator コマンド行ユーティリティの構文と使用方法を提供

表 7-1 Messaging Server のプロビジョニングメカニズム (続き)

サポートされているプロビジョニングツール	プロビジョニングツールの機能	プロビジョニングツールの制限	詳細情報
LDAP プロビジョニングツール 使用スキーマ : スキーマ 1	LDAP エントリを直接変更するツールまたはカスタムプロビジョニングツールを作成するツールを提供	• Sun Schema 2 およびその他の Java Enterprise System 製品との互換性なし	『Sun ONE Messaging Server 5.2 Provisioning Guide』および『Sun ONE Messaging and Collaboration Schema Reference Manual』を参照 Sun LDAP スキーマ 1 プロビジョニングモデルの説明 LDAP プロビジョニングツールと特定の属性およびオブジェクトクラスの使用法も説明
Sun Java™ System Console 使用スキーマ : スキーマ 1	Sun Java System Console にプロビジョニング機能があるが、Messaging ユーザーとグループのプロビジョニングには非推奨。代わりに、割り当て、ログファイル、その他の関連するメッセージストア項目などのサーバー設定の管理に Sun Java System Console を使用	• Sun Schema 2 およびその他の Java Enterprise System 製品との互換性なし • Console ではユーザーとグループを適切に追加したり変更したりできないため、プロビジョニングツールとしては非推奨	『Sun Java System Messaging Server 管理ガイド』および対応する Sun Java System Console Online Help を参照
User Management Utility 使用スキーマ : スキーマ 2	ユーザー、グループ、ドメイン、およびメーリングリストの管理者のためのコマンド行インタフェースを提供 その他の Java Enterprise System と互換性あり	• Sun スキーマ 1 との下位互換性なし • Sun Java System Identity Server では GUI プロビジョニングツールの使用不可 • Sun Java System Identity Server のインストールとこのコマンド行インタフェースの有効化が必要	『Sun Java System Communications Services User Management Utility Administration Guide』を参照 コマンド行ユーティリティの構文と使用法を提供

第 8 章

スパム防止およびウィルス対策の計画

Messaging Server は、一方的に送信されてくるバルク電子メール (UBE、または「スパム」) とウィルスに対処するためのツールを数多く提供しています。この章では、利用可能なさまざまなツールと対策について説明します。

この章には、以下の節があります。

スパム防止およびウィルス対策ツールの概要

インターネットに接続されるコンピュータの数が増加し、オンラインでのビジネスが容易となるにつれて、スパムやウィルスなどを含めたセキュリティに関する問題もいっそう増加しつつあります。そのため、これらの問題に対処するための Messaging Server 配備を計画する必要があります。

Messaging Server を経由して送受信されるメールトラフィックは、さまざまな基準に基づいて異なるチャネル別に分類できます。この基準には、発信元および宛先電子メールアドレスや発信元 IP アドレスまたはサブネットが含まれます。これらのさまざまな電子メールフローやチャネルに異なる処理特性を適用できます。その結果、これらのチャネル上で、さまざまなアクセス制御、メールフィルタ、処理の優先順位、およびツールをさまざまな方法と組み合わせで使用できます。たとえば、ドメイン内から発信されたメールを配備の外部から発信されたメールと区別して処理できます。

チャネルベースのメッセージフロー以外の便利な分類方法として、メーリングリストトラフィックがあります。Messaging Server に送信されてくる特定のメーリングリストのトラフィックは、数多くのチャネルを経由して受信し、また数多くの異なるチャネルに分けて送信できます。メーリングリストを使用すると、チャネルではなく、リスト自体を基準に考えるのが有用であることがわかります。Messaging Server はこの分類を認識し、数多くのチャネル固有のスパム対策ツールをメーリングリスト固有の方法で適用できます。

Messaging Server で使用できるスパム防止およびウィルス対策ツールの概要を以下で説明します。

- **アクセス制御** : 既知のスパム発信元からのメールを排除し、組織内でメールを送受信可能なユーザーを制御できるようにする
- **メールボックスフィルタリング** : ユーザーが Web インタフェースを通じて独自のスパムフィルタを管理し、メールボックスに配信されるメールの特性を制御できるようにする
- **アドレス検証** : 不正な発信者アドレスを持つメールを拒否する
- Real-time Blackhole List : 既知のスパム発信元のリストを責任を持って管理し、常に更新を行う Mail Abuse Protection System の Real-time Blackhole List (MAPS RBL) に基づき、スパムの発信元として認識された発信者からのメールを拒否する
- **リレーブロッキング** : メールシステムの悪用者が、メールシステムをリレーとして使用して大量の受信者にスパムを送信しようとするのを防ぐ
- **認証サービス** : Simple Authentication and Security Layer (SASL) プロトコルを使用して、SMTP サーバー内でのパスワード認証を有効にする
- **サイドライニング** : スパムの可能性のあるメッセージを確認なしで保留するか、場合によっては削除する
- **総合追跡** : 信頼性の高いメカニズムを使用してメッセージの発信元を特定する
- **変換チャネル** : サードパーティーのウィルス対策およびスパム防止製品を統合する

これらのツールは個別に使用したり、組み合わせて使用したりできます。単独ですべてのスパムを防ぐことのできるツールはありません。しかし、組み合わせて使用することで、これらのツールはメールシステムの不正使用を防ぐ効果的な手段となります。以下の節で、これらのツールについてより詳しく説明します。

アクセス制御

Messaging Server には、さまざまな検証基準に従ってメールを拒否できる汎用の機能が備わっています。この基準には、メッセージの発信元および宛先電子メールアドレスや発信元 IP アドレスが含まれます。たとえば、このメカニズムを使用して、特定の発信者やドメイン全体 (spam@public.com からのメールというような) からのメールを拒否できます。スクリーニング情報のために大量のリストが必要な場合は、アクセス基準を格納したデータベースでリストを拡張することもできます。UBE 関連以外でも、これと同じアクセス制御のメカニズムは、特定のチャネルからのメール送信を許可または禁止された内部ユーザーのデータベースを管理するのに適しています。たとえば、インターネットメールの送受信を許可するか禁止するかを、ユーザー別に制限できます。

詳細は、132 ページの「アクセス制御」および『Sun Java System Messaging Server 管理ガイド』を参照してください。

メールボックスフィルタリング

Messaging Server には、ユーザー別、チャネル別、およびシステム全体で使用できるメールフィルタがあります。ユーザー別チャネルは、Messenger Express のどの Web ブラウザからでも管理が可能です。これらのフィルタを使用して、ユーザーは自分のメールボックスに配信されるメールを制御できます。たとえば、「簡単に儲かる」式の UBE をユーザーが受け取りたくない場合、そのような件名のメールを拒否するよう指定できます。Messaging Server のメールフィルタリング機能は、Internet Engineering Task Force (IETF) により開発された Sieve フィルタリング言語 (RFCs 3028 および 3685) に基づいています。

詳細は、134 ページの「メールボックスフィルタの使用」および『Sun Java System Messaging Server 管理ガイド』を参照してください。

Brightmail や SpamAssassin のようなサードパーティーのコンテンツフィルタリングソフトウェアを使用して、コンテンツベースなウィルススキャニングのフィルタリングを実装することも可能です。詳細は、123 ページの「スパム防止およびウィルス対策の考察」を参照してください。

アドレス検証

UBE メッセージは、しばしば不正発信者のアドレスを使用します。Messaging Server SMTP サーバーは、メッセージを不正な発信者のアドレスと照合させることで、これを利用できます。発信者のアドレスが DNS サーバーに対するクエリにより有効なホストネームに対応していないと判断された場合、そのメッセージは拒否されます。ただし、DNS をこのように使用する場合は、パフォーマンスが低下する可能性があります。

『Sun Java System Messaging Server 管理ガイド』で説明されているチャネルキーワード `mailfromdnsverify` を使用して、チャネル別ベースのアドレス検証を行うことができます。

Real-time Blackhole List

Mail Abuse Protection System の Real-time Blackhole List (MAPS RBL) は、発信元 IP アドレスによって識別された既知の UBE 発信元のリストを動的に更新します。Messaging Server SMTP サーバーは MAPS RBL をサポートしており、MAPS RBL が UBE の発信元として認識した IP アドレスからのメッセージ受信を拒否できます。MAPS RBL は、インターネット DNS を使用した無料サービスです。

詳細は、以下を参照してください。

`http://mail-abuse.org/rbl`

MTA Dispatcher の `ENABLE_RBL` オプションを使用すると、Messaging Server SMTP サーバーで RBL を有効にできます。詳細は、『Sun Java System Messaging Server 管理ガイド』を参照してください。

リレーブロッキング

総合的な UBE 対策としては、アクセス制御、メールボックスフィルタリング、アドレス検証、RBL を使用して UBE を受け取らないようにする対策と、システムが不正に利用されてメールを他のシステムにリレーしてしまうことを防ぐ対策が必要です。後者は、リレーブロッキングと呼ばれます。リレーブロッキングの最も単純な方法は、非ローカルシステムからのリレーを拒否しながら、ローカルユーザーとシステムにはメールのリレーを許可することです。IP アドレスを選別の基準として使用すると、ローカルと非ローカルを簡単かつ安全に判断できます。デフォルトでは、Messaging Server はインストール時にリレーブロッキングを行うように設定されます。詳細は、134 ページの「マッピングテーブルによるリレー防止設定」および『Sun Java System Messaging Server 管理ガイド』を参照してください。

認証サービス

Messaging Server SMTP サーバーには Simple Authentication and Security Layer (SASL、RFC2222) が実装されています。SASL は POP クライアントと IMAP クライアントで使用され、SMTP サーバーにパスワードベースのアクセスを行うことができます。SASL の一般的な使用方法は、認証を受けた外部ユーザーにメールのリレーを許可することです。これにより、自宅からまたは出張中に ISP を使用するローカルユーザーに共通の問題が解決されます。そのようなユーザーは、メールシステムに接続するときに、ローカルとは異なる IP アドレスを使用します。発信元 IP アドレスのみを考慮するリレーブロックでは、これらのユーザーのメールはリレーされません。この問題は、SASL を使用してこれらのユーザーの認証を可能にすることで解決できます。一度認証を受けたユーザーは、メールのリレーが許可されます。

サイドライニング

前述したアクセス制御のメカニズムでは、疑わしいメッセージの処理を保留しておき、あとで手動で検査することもできます。あるいは保留する代わりに、宛先アドレスを変更して疑わしいメールを特定のメールボックスに配信したり、警告なしで削除したりすることもできます。この対策は、UBE が既知の固定された発信元から送られてきたものである場合に有効で、これを完全に受信拒否してしまうと、悪用者が発信元を変更してしまうだけの結果となってしまいます。Messaging Server のメーリングリストでも同様の機能を使用できます。警告なしでメールを削除する場合には、正当な送信者が影響を受けないよう慎重に行う必要があります。

詳細は、『Sun Java System Messaging Server 管理ガイド』を参照してください。

総合追跡

Messaging Server の SMTP サーバーは、すべての受信メールメッセージに関する重要な発信元情報を検出し、記録します。この情報には、発信元 IP アドレスとそれに対応するホスト名が含まれます。検出されたすべての情報は、設定によりログファイルと共にメッセージの追跡フィールド (たとえば、受信ヘッダ行) に記録されます。そのような信頼性の高い情報を利用できることは、ヘッダが詐称されることの多い UBE の発信元を突き止めるのに重要なことです。各サイトでは任意のレポートツールを使用して、プレーンテキストで保存されているこの情報にアクセスできます。

変換チャネル

変換チャネルは非常に汎用的な目的で使われるインタフェースです。チャネル上でスクリプトやプログラムを呼び出して、電子メールメッセージの本文を任意に処理できます。変換プログラムは、それぞれの MIME のメッセージ全体ではなく本文をプログラムまたはスクリプトに渡し、その本文をプログラムまたはスクリプトの出力に置き換えます。変換チャネルは、テキスト形式から PostScript 形式へというようにファイル形式を変換したり、ある言語を別の言語に変換したり、会社の機密情報のためにコンテンツフィルタリングを実行したり、ウィルスを検索したり、メッセージを別のものに置き換えたりするのに使用できます。

サードパーティー製品との統合

Messaging Server の変換チャネルを使用すると、サードパーティーの供給元が提供するコンテンツフィルタリングソフトウェアを配備に統合できます。チャネルキーワードは、Brightmail または SpamAssassin のようなスパム防止およびウィルス対策製品を使用したメールフィルタリングを行うのに使用されます。MTA を設定して、すべてのメッセージまたは特定のチャネルを経由するメッセージのフィルタリングを行ったり、ユーザー別のレベルでフィルタの精度を設定したりできます。スパム防止とウィルス対策のいずれか、または両方の使用を選択できます (SpamAssassin はスパムのフィルタリングのみを行う)。

Sieve の広範囲なサポートにより、スパムやウィルスであると判定されたメッセージの処理設定に大きな柔軟性を持たせることが可能となりました。ウィルスとスパムの削除をデフォルトの動作とするか、スパムを特定のフォルダに集めることができます。ただし、Sieve を使用する場合は、メッセージのコピーを特別なアカウントに転送するか、カスタムヘッダーを追加するか、spamtest sieve 拡張を使用して、SpamAssassin から返されるレイティングに基づいて異なる動作を行うことができます。

スパム防止およびウィルス対策の考察

この節では、スパム防止とウィルス対策の技術を使用した配備を計画する場合の留意事項について、説明します。

スパム防止およびウィルス対策を配備する場合のアーキテクチャ上の問題

Messaging Server MTA は Brightmail または SpamAssassin のようなメールフィルタリングシステムと同じシステムで使用することも、別のシステムで使用することもできます。MTA をメールフィルタリングサーバーから分離することのメリットは、ハードウェアを追加してサーバーのクローンを使用すれば、簡単にフィルタリングの処理能力を上げられることです。システムの能力に余裕があり、過負荷状態になっていない場合は、メールフィルタリングサーバーソフトウェアを MTA と同じサーバーに置くことができます。

一般には、MTA がフィルタを使用するサーバーには、Brightmail の「ファーム」を配備することを検討します。MTA が Brightmail サーバー名のリストを使用するように設定すると、MTA の負荷を分散できます (このロードバランシング機能は、Brightmail SDK により可能になる)。Brightmail サーバーのファームを導入するメリットは、より多くの処理パワーが必要な場合に、Brightmail サーバーを追加するだけで対応できることです。

メールフィルタリング製品は、一般に高い CPU 占有率を要求します。MTA とメールフィルタリング製品をそれぞれ専用のマシンに分けるアーキテクチャを構築することで、メッセージング配備の全体的なパフォーマンスを向上させることができます。

注 メールフィルタリングサーバーの CPU 占有率が高い傾向にあるため、フィルタリングの対象となる MTA ホスト以上の数のメールフィルタリングシステムを使用するアーキテクチャに行き着く場合もあります。

大規模な配備では、それぞれの受信メールと送信メールの MTA プールに対応する、サーバーの受信および送信フィルタリングプールを構築することも検討します。また、「スイング」プールを構築して、必要とされる状況に応じて受信送信のいずれかのプールとして機能させることもできます。

その他の配備全般と同様に、メールフィルタリング層を常時監視する必要があります。経験上、CPU 占有率 50% をしきい値とするのが良い指針です。このしきい値に達したら、メールフィルタリング層の能力増強を検討する必要があります。

RBL の実装

一般的には、RBL を実装するとすぐにスパムを減らすことができます。MTA によって RBL が実装されれば、スパムを少なくとも 10% 以上、すぐに減らすことができます。この数字が 50% にまで達する場合もあります。

RML と Brightmail とは併用できます。Brightmail が一定の時間内に特定の IP アドレスの電子メール 100 件のうち 95 件を処理した場合、その IP アドレスを RBL に追加する必要があります。Brightmail の分析を行う際、Brightmail のメール判定基準のために RBL を調整できます。この調整により、RBL はスパムが集中する場合の処理に十分に備えられます。

スパム防止およびウィルス対策配備の一般的なシナリオ

この節では、Brightmail および SpamAssassin で一般的な配備例について説明します。詳細は、『Sun Java System Messaging Server 管理ガイド』を参照してください。

http://docs.sun.com/db/prod/entsys?l=ja

Brightmail を使用する

Brightmail には、以下の一般的な配備シナリオがあります。

- ローカルメッセージストア (`ims-ms` チャネル) に届く受信メッセージの処理
- インターネット (`tcp-local` チャネル) に送られるメッセージの処理
- インターネット (`tcp-local` チャネル) から届くメッセージの処理
- 特定のドメインに送られるメッセージの処理 (`per-domain` オプション)
- 特定のユーザーに送られるメッセージの処理 (`per-user` オプション)
- Class-of-Service オプションとしての Brightmail 処理の設定

Brightmail がスパムとウィルスの両方のチェックを実行する場合、MTA のメッセージスループットは 50% ほど低下する可能性があります。MTA のスループットを維持するには、各 MTA につき 2 台の Brightmail サーバーが必要です。

SpamAssassin を使用する

Messaging Server では、SpamAssassin の使用がサポートされています。SpamAssassin はフリーウェアのメールフィルタで、スパムの特定に使用されます。SpamAssassin は Perl で記述されたライブラリ、アプリケーションのセット、および SpamAssassin のメッセージングシステムへの統合に使用するユーティリティで構成されています。

SpamAssassin では、すべてのメッセージのスコアが計算されます。スコアは、メッセージヘッダーや本文の情報に対して一連のテストを実行することによって計算されます。各テストに成功するか失敗するかによってスコアは調整されます。スコアは正または負の実数です。スコアが一定のしきい値 (通常 5.0) を超えると、スパムであるとみなされます。

SpamAssassin には高い設定性があります。テストはいつでも追加したり削除したりでき、既存テストのスコアは調整できます。これらはすべてさまざまな設定ファイルを通じて実行されます。SpamAssassin の詳細については、SpamAssassin の Web サイトを参照してください。

http://www.spamassassin.org

Brightmail のスパムおよびウィルススキャンライブラリを呼び出す場合と同じ方法で SpamAssassin `spamd` サーバーに接続できます。

スパム防止およびウィルス対策のためのサイトポリシーの開発

スパムとスパムのリレーを防止するためのポリシーを開発する場合には、スパムの防止機能と電子メールがサイトにタイムリーに配信されることのバランスを取る必要があります。したがってベストのポリシーは、処理時間があまり長くない判定基準をコアとして最初に配置して、スパムの大半を捕捉することです。最終アーキテクチャでストレステストを行ったあとに、この判定基準のコアセットを定義できます。最初の判定基準は以下の内容で始めます。システムを配備したら、捕捉されたスパムと捕捉されなかったスパムを分析してシステムの微調整を行い、必要に応じて機能を入れ替えたり、新機能を追加したりします。

サイトのスパム防止およびウィルス対策ポリシーの開始点として、以下の判定基準を使用します。

- リレー防止は、ORIG_SEND_ACCESS の設定により行う。この構造により、登録者とパートナーのユーザーだけがアクセスを許可され、SMTP 経由でメールを外部に送信できる
- 認証サービスを使用して、ローミングユーザーを検証する。これらのユーザーは、識別情報が確認された後で SMTP 経由の外部への送信を許可される
- システム全体のメールボックスフィルタを使用して、件名行をチェックしてスパムに共通の言い回しをチェックする機能を実装する
- holdlimit キーワードを使用して、メール受信者の最大数を設定する。これにより、スパムの可能性のあるトラフィックを保留にできる。受信者数の初期値を 50 に設定しておいて、ある程度の期間監視を続けてから、必要に応じて増減する
- ポストマスターがマニュアルで利用する、専用のダミーアカウントを設定して、そのアカウントに送られてくるスパムを監視して新しいスパムサイトをつきとめる
- ウィルスが検出されたメッセージは送信者には返送せず、内部の受信者にも転送すべきではない。このようなメッセージでは、ウィルスがメールを生成し、送信者アドレスを偽造しているため、無意味であるため。ウィルスに感染しているメッセージが重要なものであることはきわめて稀
- 感染しているメッセージは、ウィルスに関する情報を収集してリスト化するウィルス対策エンジンに送る。そのような情報を利用して、システム管理者に新しいウィルスやワームの発生を通知するレポートを作成できる

第 9 章

セキュリティで保護された Messaging Server の設計

この章では、セキュリティの手法の概要、一般的なセキュリティ脅威、およびセキュリティニーズ分析の手順の概要について説明します。

この章には、以下の節があります。

セキュリティ戦略の作成

セキュリティ戦略の作成は、配備計画のなかで最も重要なステップの 1 つです。セキュリティ戦略は、組織のセキュリティに対するニーズを満たし、ユーザーに不便を強いることなくセキュリティが確保されたメッセージ環境を提供するものでなければなりません。

さらに、セキュリティ戦略は単純なものにして、管理を容易に行えるようにしておく必要があります。複雑なセキュリティ戦略を用いると、ユーザーがメールにアクセスできなかったり、ユーザーや権限のない侵入者によってアクセスされては困る情報が変更されたり、収集されたりする問題が生じます。

RFC 2196、『Site Security Handbook』に、セキュリティ戦略を構築するための 5 つのステップが記されています。

1. 何を保護するのかをはっきりさせる

たとえば、保護対象のリストにはハードウェア、ソフトウェア、データ、従業員、文書、ネットワークインフラストラクチャ、または組織の評判などが含まれます。

2. 何から保護するのかを判断する

   例 : 権限のないユーザー、スパマー、またはサービス拒否攻撃

3. システムに対する脅威の可能性を推測する

   大規模なサービスプロバイダの場合、セキュリティが脅威に晒される可能性は小規模な組織よりもはるかに高いといえます。さらに、組織の性格がセキュリティに対する脅威を誘発することも考えられます。

4. 費用対効果の高い方法で資産を守る対策を導入する

   たとえば、SSL 接続を設定する際のオーバーヘッドによって、メッセージング配備のパフォーマンスに対する負荷が発生する可能性があります。セキュリティ戦略を設計するうえで、セキュリティニーズとサーバーの能力のバランスを取る必要があります。

5. 戦略を常時見直し、弱点が発見されるたびに戦略を練り直して、よりすぐれたものに改善する

   定期的な監査を行い、セキュリティポリシーの全体的な有効性を検証します。監査は、ログファイルと SNMP エージェントが記録した情報を調査することで行います。SNMP の詳細については、『Sun Java System Messaging Server 管理ガイド』を参照してください。

セキュリティ戦略では、以下の項目についても計画する必要があります。

- 物理的なセキュリティ
- サーバーセキュリティ
- ネットワークセキュリティ
- メッセージングセキュリティ

物理的なセキュリティ

インフラストラクチャの重要部分への物理的なアクセスを制限します。たとえば、ルーター、サーバー、配線クローゼット、サーバールーム、データセンターを、窃盗、改竄、その他の悪用から保護するために、物理的な制限を設けます。権限を持たない人物にサーバールームへの侵入を許し、ルーターの配線を抜かれることがあるようでは、ネットワークとサーバーのセキュリティも無意味なものとなります。

サーバーセキュリティ

重要なオペレーティングシステムアカウントとデータへのアクセスを制限することも、セキュリティ戦略の一部となります。この保護は、オペレーティングシステムで利用できる認証とアクセス制御のメカニズムにより行われます。

さらに、最新のオペレーティング環境のセキュリティパッチをインストールし、数ヶ月ごとに、またベンダーからのセキュリティ警告に対応して、パッチを更新する必要があります。

ネットワークセキュリティ

ネットワークへのアクセスを制限することは、セキュリティ戦略の重要なポイントとなります。通常は、ファイアウォールを使用してネットワークへの全般的なアクセスを制限します。ただし、電子メールはサイト外から使用できるようにしておく必要があります。SMTP がそのサービスの 1 つに該当します。

ネットワークのセキュリティを確保するには、以下の条件が必要となります。

- 使用しないポート上で待機している、オペレーティングシステムが提供するすべてのサービスを停止します。
- 可能な場合は、`telnet` を `sshd` に置き換えます。
- パケットフィルタで内部の発信元 IP アドレスを持つ外部パケットを拒否し、その背後に Messaging Server を配置します。パケットフィルタは、明示的に指定したポート以外に向けたすべての外部接続を遮断します。

メッセージングセキュリティ

Messaging Server には、以下のセキュリティ機能があります。

- 配備におけるメッセージングコンポーネントの保護

  このオプションセットにより、MTA リレー、メッセージストア、Messenger Express メールクライアント、および多重化サービスのセキュリティが確保されます。さらに、サードパーティーのスパムフィルタオプションについてもわかります。

- ユーザー認証の計画

  これらのオプションを使用して、メールサーバーでユーザーが認証される仕組みを決定し、権限を持たないユーザーがシステムにアクセスするのを防ぐことができます。

- メッセージ暗号化戦略の計画

  このオプションセットを使用すると、認証された SMTP とデジタル署名の証明書、暗号、SSL (Secure Sockets Layer) によるユーザー認証とメッセージの保護を行うことができます。

この章の後半では、メッセージングシステムを保護するセキュリティ手法について説明します。

配備におけるメッセージングコンポーネントの保護

この節では、メッセージング配備でコンポーネントをセキュリティで保護する方法について説明します。

注	それぞれのコンポーネントで、`chroot` 機能を使用して、マシンで使用できるコマンドの数を制限します。

MTA リレーの保護

セキュリティで保護された MTA リレーにより、リソースの処理とサーバーを保護します。権限を持たないユーザーからメッセージがリレーされた場合、または大量のスパムが配信された場合には、応答速度が遅くなり、ディスク容量が圧迫され、エンドユーザーのための処理リソースが消費されます。スパムはサーバーのリソースを浪費するだけでなく、エンドユーザーを煩わせるものでもあります。

注	権限を持たない外部のユーザーから配備を保護するだけでなく、内部ユーザーからシステムを保護する必要もあります。

以下の表で、MTA リレーに対する最も一般的な脅威について説明します。

表 9-1 MTA リレーに対する一般的なセキュリティ脅威

脅威	説明
UBE (Unsolicited bulk email) またはスパム	電子ジャンクメールを多数のユーザーに送信する方法を参照
権限に基づかないリレー	別の会社の SMTP サーバーを使用してメールをリレーする。スパムの送信者は、自分の軌跡を消すためにこのテクニックを多用。エンドユーザーは、スパムの送信者にではなくリレーを行ったところに苦情を出す
メール爆弾	同じメッセージを特定のアドレスに繰り返し送るような行為。大量のメッセージにより、メールボックスの容量を超過させるのが狙い
電子メールスプーフィング	別の発信元からの電子メールを、ある発信元からのものにみせかける
サービス拒否攻撃	あるサービスの正規ユーザーがそのサービスを利用できないようにする。たとえば、アタッカーがネットワークを占有し、正規のユーザーのトラフィックを妨害する

MTA リレーに関するこの節では、配備で使用できるセキュリティオプションについて説明します。

- アクセス制御
- 変換チャネルとサードパーティーのフィルタリングツール
- RBL チェック
- クライアントアクセスの制御

- セキュリティ戦略の監視

アクセス制御

アクセス制御を使用して、特定のユーザーから (へ) のメッセージをシステムレベルで拒否できます。また、特定のユーザー間でより複雑なメッセージトラフィックの制限を構成することもできます。さらに、ユーザーに独自の受信メッセージのフィルタ設定を許可し、メッセージヘッダの内容に基づいてメッセージを拒否することなどができます。

エンベロープレベルでアクセス制御を行いたい場合は、マッピングテーブルを使用してメールのフィルタリングを行います。ヘッダベースでアクセス制御を行いたい場合、またはユーザー独自の制御を行いたい場合は、サーバー側のルールとともに一般的なメールボックスフィルタを使用します。

マッピングテーブルの概要

メールサービスへのアクセスを制御するには、一定のマッピングテーブルを使用します。以下の表で、マッピングテーブルの使用により、だれがメールを送信または受信できるのか、あるいは送受信ともにできるのかを制御する方法を説明します。詳細は、『Sun Java System Messaging Server 管理ガイド』を参照してください。

表 9-2 アクセス制御マッピングテーブル

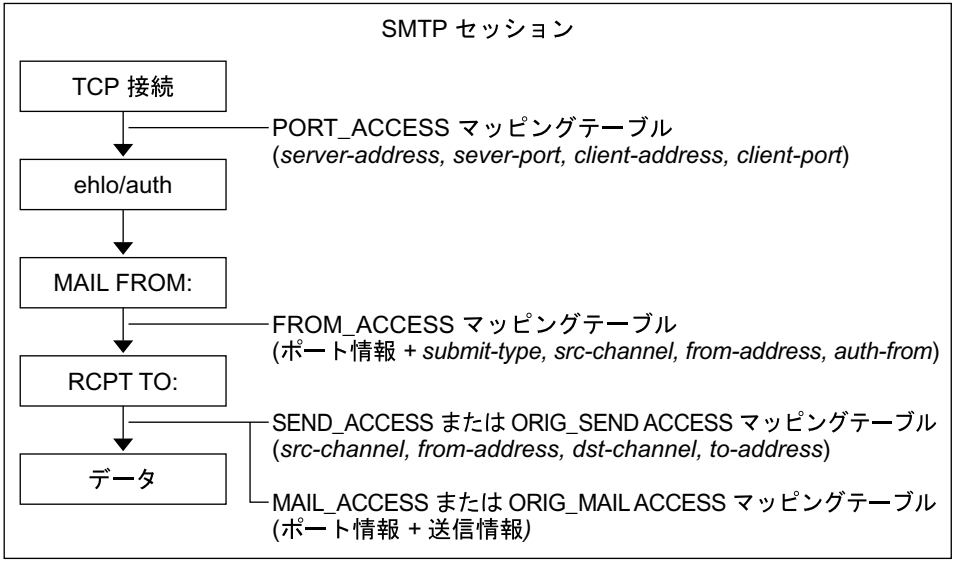
マッピングテーブル	説明
SEND_ACCESS	エンベロープ From: アドレス、エンベロープ To: アドレス、ソースおよび宛先チャネルに基づいて、受信接続をブロックする場合に使用する。書き換えやエイリアス展開などの処理が行われてから、To: アドレスが調べられる
ORIG_SEND_ACCESS	エンベロープ From: アドレス、エンベロープ To: アドレス、ソースおよび宛先チャネルに基づいて、受信接続をブロックする場合に使用する。書き換え後、エイリアス展開の前に To: アドレスが調べられる
MAIL_ACCESS	SEND_ACCESS テーブルと PORT_ACCESS テーブルを組み合わせた情報に基づいて受信接続をブロックする場合に使用する。SEND_ACCESS のチャネルとアドレス、および PORT_ACCESS の IP アドレスとポート番号に関する情報が基準となる
ORIG_MAIL_ACCESS	ORIG_SEND_ACCESS テーブルと PORT_ACCESS テーブルを組み合わせた情報に基づいて受信接続をブロックする場合に使用する。ORIG_SEND_ACCESS のチャネルとアドレス、および PORT_ACCESS の IP アドレスとポート番号に関する情報が基準となる

表 9-2　アクセス制御マッピングテーブル (続き)

マッピングテーブル	説明
FROM_ACCESS	エンベロープ From アドレスに基づいてメールをフィルタリングする場合に使用する。このテーブルは、To: アドレスが不適切な場合に使用する
PORT_ACCESS	IP 番号に基づいて受信接続をブロックする場合に使用する

以下の図は、メール受信プロセスの中でマッピングテーブルが使用される場所を示したものです。

図 9-1　マッピングテーブルとメール受信プロセス



MTA サービスディスパッチャーが管理するすべてのネットワークポートで、PORT_ACCESS 拒否応答が保証されている場合は、リモートホストから最初の接続が行われた時点で、それが実行されます。FROM_ACCESS による拒否は、送信側が受信者情報またはメッセージデータを送信する前に、MAIL FROM: コマンドへの応答として行われます。SEND_ACCESS または MAIL_ACCESS による拒否は、送信側がメッセージデータを送信する前に、RCPT TO:: コマンドへの応答として行われます。SMTP メッセージが拒否された場合は、Messaging Server がメッセージデータを受信せずメッセージデータを確認しないため、そのような拒否を処理するためのオーバーヘッドが最小になります。複数のアクセス制御マッピングテーブルが存在する場合、Messaging Server はそれらをすべて調べます。

注	メッセージが受け入れられた場合は、さらに変換チャネルとユーザー定義のフィルタによりフィルタリングされます。

マッピングテーブルによるリレー防止設定

アクセス制御マップを使うことによって、Messaging Server システムが SMTP メールのリレーに利用されるのを防ぐことができます。たとえば、メールシステムを使ってジャンクメールを多数のメールボックスに送信しようとする者がいるような場合です。

Messaging Server のデフォルトでは、ローカルの POP メールクライアントおよび IMAP メールクライアントによるリレーを含むすべての SMTP リレー操作が防止されます。143 ページの「Authenticated SMTP を有効にする」で説明しているように、IMAP クライアントまたは POP クライアントが SMTP AUTH により認証されなかった場合で、Messaging Server の SMTP サーバーを使って外部アドレスにメッセージを送信しようとした場合、その送信は拒否されます。このため、内部システムとリレーを許可するサブネットを認識するように設定を変更した方がよいでしょう。

➤ 外部ホストからのリレーを防止するには

ドメイン外にあるホストからドメイン外の別のホストにメッセージがリレーされるのを防ぐには、以下の方法を取ります。

1. 受信メールをいくつかのチャネルに分けます。
   例 :
   - ドメイン内の IP アドレスは `tcp_internal` チャネルに送られます。
   - 認証されたセッションは `tcp_auth` チャネルに送られます。
   - その他のすべてのメールは、`tcp_local` チャネルに送られます。
2. 『Sun Java System Messaging Server 管理ガイド』の「メールフィルタリングとアクセス制御」の章で詳しく説明されているように、POP クライアントと IMAP クライアントからのメールは、`INTERNAL_IP` マッピングテーブルを使用して識別され、処理が許可されます。

メールボックスフィルタの使用

フィルタは、メッセージに適用される 1 つ以上の条件付きアクションで構成されています。Messaging Server フィルタはサーバーに保存され、サーバーによって評価されます。そのため、それらは SSR (サーバー側ルール) と呼ばれることもあります。

チャネルレベルのフィルタと MTA 全体のフィルタを作成し、不正メールの配信を防止できます。また、フィルタテンプレートを作成し、Messenger Express を使用してそれをエンドユーザーに使わせることもできます。エンドユーザーはテンプレートを使用して個人のメールボックスフィルタを構築し、不要なメールメッセージが自分のメールボックスに配送されないようにすることができます。サーバーは、次の優先順位に従ってフィルタを適用します。詳細は、『Sun Java System Messaging Server 管理ガイド』を参照してください。

1. ユーザー単位のフィルタ

   ユーザー単位のフィルタは、特定ユーザーのメールボックスに送信されるメッセージに適用されます。フィルタテンプレートを作成し、Messenger Express クライアントを使用してそれをエンドユーザーに使わせることができます。エンドユーザーはテンプレートを使用して個人のサーバーフィルタを構築し、自分のメールボックスへのメールメッセージの配送を管理できます。フィルタにより、不要なメッセージ、リダイレクトメールなどの拒否や、メールボックスフォルダに配信されるメッセージのフィルタリングなどが行われます。

   個人用メールボックスフィルタにメッセージの許可あるいは拒否が定義されている場合は、メッセージに対してそのフィルタ処理が行われます。

   フィルタテンプレートは、Sieve スクリプトの「ハードコード」された要素をプロンプトと入力フィールドに置き換えます。Java サーブレットは、sieve テンプレートを解析し、ブラウザ内でユーザーインタフェースを生成するのに使用されます。エンドユーザーが入力フィールドに値を入力すると、サーブレットがその値を取得して、ユーザーのディレクトリにあるプロファイルエントリ内の sieve スクリプトに保存します。Messenger Express インタフェースを通じて、プロンプトと入力フィールドがエンドユーザーに提示されます。

   しかし、受取人がメールボックスフィルタを設定していない場合、またはユーザーのメールボックスフィルタが明示的に適用されないメッセージの場合、Messaging Server によってチャネルレベルのフィルタが適用されます。

2. チャネルレベルのフィルタ

   チャネルレベルのフィルタは、チャネルのキューに入った各メッセージに適用されます。この種のフィルタの一般的な用途は、特定のチャネルから入ってくるメッセージをブロックすることです。

   チャネルレベルのフィルタを作成するには、SIEVE を使用してフィルタを書く必要があります。SIEVE を使用してフィルタを作成する場合の指示の詳細は、『Sun Java System Messaging Server 管理ガイド』の「メールフィルタリングとアクセス制御」の章を参照してください。

   チャネルレベルのフィルタにメッセージの許可あるいは拒否が定義されている場合は、メッセージに対してそのフィルタ処理が行われます。それ以外の場合は、Messaging Server によって MTA 全体のフィルタが適用されます (該当する場合)。

3. MTA 全体のフィルタ

MTA 全体のフィルタは、MTA のキューに入るすべてのメッセージに適用されます。この種のフィルタの一般的な用途は、メッセージの宛先とは関係なく、ダイレクトメールや受信したくないメッセージをブロックすることです。

MTA 全体のフィルタを作成するには、SIEVE を使用してフィルタを書く必要があります。SIEVE を使用してフィルタを作成する場合の指示の詳細は、『Sun Java System Messaging Server 管理ガイド』の「メールフィルタリングとアクセス制御」の章を参照してください。

デフォルト設定を使用した場合、それぞれのユーザーはメールボックスフィルタを所有していません。ユーザーが Messenger Express インタフェースにアクセスして 1 つまたは複数のフィルタを作成すると、そのフィルタが LDAP ディレクトリに保存されます。

変換チャネルとサードパーティーのフィルタリングツール

変換チャネルは、MTA を通じて配信されるメッセージを本文部分ごとに変換します。この処理は、サイトで提供されるプログラムかコマンドにより行われます。変換チャネルは、テキストや画像のフォーマット変換、ウィルスのスキャン、言語の変換などを行うことができます。MTA で通信するさまざまなメッセージ形式を変換することができ、特定の処理やプログラムをメッセージの本文部分に指定することができます。変換チャネルをウィルススキャニングプログラムと併用する場合は、ウィルスの除去、メッセージの保留または拒否を選択できます。特別な変換チャネル設定を使用すると、それぞれのメッセージ本文に対する適切な変換を選択できます。詳細については、『Sun Java System Messaging Server 管理ガイド』の「事前定義のチャネルの使用」の章を参照してください。

注 変換チャネルのような特別な処理を行うと、システムに余分の負荷がかかります。戦略のサイズを検討する場合には、この点を考慮してください。

変換チャネルを使用すると、サードパーティーのスパム防止およびウィルス対策ソフトウェアソリューションを利用できます。また、MTA API を使用してチャネルを作成し、リモートスキャニングエンジンを起動することもできます。MTA API の詳細については、『Sun Java System Messaging Server Developer's Reference』を参照してください。

一般に、サードパーティーのソリューションは外部サイトから保護して、バックエンドまたは中間のリレーのみで使用するのが最も適した使い方です。

Brightmail ソリューションは、Brightmail サーバーと、リアルタイムのスパム防止およびウィルス対策 (サービスプロバイダ向けのみ) ルールアップデートで構成されており、ルールはメッセージングサーバーにダウンロードされます。Brightmail Logistics and Operations Center (BLOC) が電子メールプローブからスパムを受信すると、オペレータがただちに適切なスパム防止ルールを作成します 次に、これらのルー

ルが Brightmail カスタママシンにダウンロードされます。同様に、Symantec Security Response のリアルタイムのウィルスルールが Brightmail から送信されます。これらのルールは顧客の Brightmail サーバーでスパムやウィルスを検出するために使用されます。

Messaging Server では、SpamAssassin の使用もサポートされています。SpamAssassin はフリーウェアのメールフィルタで、スパムの特定に使用されます。SpamAssassin では、すべてのメッセージのスコアが計算されます。スコアは、メッセージヘッダーや本文の情報に対して一連のテストを実行することによって計算されます。各テストに成功するか失敗するかによってスコアは調整されます。スコアは正または負の実数です。スコアが一定のしきい値を超えると、スパムであるとみなされます。

Brightmail および SpamAssassin for Messaging Server の設定の詳細については、『Sun Java System Messaging Server 管理ガイド』を参照してください。

RBL チェック

Mail Abuse Protection System の Real-time Blackhole List (MAPS RBL) は、スパムの発信やリレーを行ったり、スパムのサポートサービスを提供したりしてホストやネットワークを悪用している者に好意的、あるいは中立的な立場を取っていると判断されたホストとネットワークのリストです。

外部からの MAPS RBL に対する接続の比較を行うように、MTA リレーを設定することができます。また、DNS ベースのデータベースを使用して、不特定多数宛のメールを送る可能性のある受信 SMTP 接続を判別できます。

詳細については、『Sun Java System Messaging Server 管理ガイド』の「メールフィルタリングとアクセス制御」の章を参照してください。

クライアントアクセスの制御

Messaging Server は、POP、IMAP、および HTTP について、サービスごとの高度なアクセス制御機能をサポートしています。Messaging Server のアクセス制御機能は、TCP デーモンと同じポートで待機するプログラムです。アクセス制御機能では、アクセスフィルタによるクライアントの識別情報の検証が行われ、そのクライアントがフィルタリング処理を通過した場合は、デーモンへのアクセスが許可されます。

大企業やサービスプロバイダのメッセージングサービスを管理する場合、これらの機能を使用して、スパム (大量メール送信) や DNS スプーフィングを行うユーザーをシステムから除外したり、ネットワークの全般的なセキュリティを強化したりできます。

Messaging Server の TCP クライアントアクセス制御システムは、必要な場合、その処理の一部として、次のようなソケットの終端アドレスの分析を行います。

- 両方の終端の逆引き DNS 検索 (名前に基づくアクセス制御を行うため)

- 両方の終端の正引き DNS 検索 (DNS スプーフィングを検出するため)
- `Identd` コールバック (クライアントエンドのユーザーがクライアントホストに認識されていることを調べるため)

システムは、この情報をフィルタと呼ばれるアクセス制御文と比較して、アクセスの許可または拒否を決定します。サービスごとに、個別の許可フィルタと拒否フィルタのセットを使用して、アクセスを制御します。許可フィルタは明示的にアクセスを許可し、拒否フィルタは明示的にアクセスを禁止します。

クライアントがサービスへのアクセスを要求すると、アクセス制御システムは、そのクライアントのアドレスまたは名前情報を、以下の条件を使用して順番に対象のサービスのフィルタと比較します。

1. 検索は、最初の一致項目が見つかった時点で終了する。許可フィルタは、拒否フィルタより先に処理されるため、許可フィルタが優先される
2. クライアント情報が対象のサービスの許可フィルタに一致した場合は、アクセスが許可される
3. クライアント情報がそのサービスの拒否フィルタに一致した場合は、アクセスが拒否される
4. どの許可または拒否フィルタにも一致しなかった場合、アクセスが許可される。例外は、許可フィルタは存在しているが拒否フィルタが存在しない場合で、その場合にはフィルタに一致しなかったアクセスは拒否される

ここで説明するフィルタの構文は柔軟性に富んでいるため、わかりやすい簡単な方法で、さまざまなアクセス制御ポリシーを実装できます。許可フィルタと拒否フィルタは自由に組み合わせて使用できますが、大半のアクセスを許可するフィルタまたは大半のアクセスを拒否するフィルタを使用すると、ほとんどのポリシーを実装できます。

クライアントアクセスフィルタは、問題のあるドメインの数が把握できる場合に特に有効です。UBE の場合、Messaging Server はすべてのスパムメッセージを格納し処理しなければなりませんが、クライアントアクセスフィルタの場合はスパムメッセージを処理する必要がありません。クライアントアクセスフィルタはドメイン全体からのメールをブロックするため、この機能は慎重に使用する必要があります。

クライアントアクセスフィルタには、以下の制限があります。

- メッセージをリレーする前に、SMTP クライアントがログインする必要がある
- クライアントアクセスフィルタは、大規模は配備には向いていない

クライアントアクセスフィルタの詳細については、『Sun Java System Messaging Server 管理ガイド』の「セキュリティ設定とアクセス制御」の章を参照してください。

セキュリティ戦略の監視

サーバーの監視は、セキュリティ戦略で重要な位置を占めます。システムに対する攻撃を識別するには、メッセージキューのサイズ、CPU の使用率、ディスクの空き容量、ネットワークの使用率を監視します。メッセージキューのサイズが異常に大きくなったり、サーバーの応答時間が長くなったりするのは、MTA リレーへの攻撃の可能性があります。また、通常とは異なるシステムの負荷パターンや接続についても調査します。ログを毎日チェックして、異常な活動がないか調べます。

メッセージストアの保護

メッセージングサーバーで最も重要なデータは、メッセージストア内のユーザーのメールです。メールメッセージは、暗号化されない個別のファイルとして格納されることに留意してください。したがって、物理的なアクセスや root アクセスからメッセージストアを保護する必要があります。

メッセージストアをセキュリティで保護するには、ストアがインストールされているマシンへのアクセスを制限します。暗号化されないプレーンテキストのパスワードの代わりに、CRAM-MD5 パスワードまたは DIGEST-MD5 パスワードを使用できます。パスワードの詳細については、141 ページの「ユーザー認証の計画」を参照してください。

ストアマシンの認証にパスワードを作成するだけでなく、VPN アクセス、ssh、または pam のようなツールを使用して、マシンへのログインが許可されている有効なユーザーのリストを作成することもできます。

また、1 階層のアーキテクチャよりも 2 階層のアーキテクチャをお勧めします。メッセージストアは、メッセージングシステムのコンポーネント中で最もディスクに負担をかける作業を行うため、フィルタリング、ウィルススキャニング、およびその他のディスクに負担をかけるセキュリティ処理を同じマシンで行わないようにします。2 階層のアーキテクチャでは、システムに余分な負荷がかかるメッセージストアと同じマシンで UBE フィルタ、リレー防止機能、およびクライアントアクセスフィルタを使用せずにすみます。代わりに、MTA リレーがその処理を行います。さらに、ストアへのユーザーアクセスが 2 階層配備の MMP または MEM (Messenger Express Multiplexor) により制限され、実質的にメッセージストアにセキュリティ層を追加したことになります。

1 階層のアーキテクチャで配備を行う場合は、セキュリティ処理の追加と、SSL やウィルススキャニングなどに必要となる負荷を考慮してください。詳細は、第 6 章「サイズ決定戦略の計画」を参照してください。

メッセージストアにセキュリティ処理を追加する場合は、ユーザーごとにディスク割り当てを行って、ディスクの使用率を制限します。また、空き容量が制限に近づいたときには管理アラームを出すようにします。さらに、MTA の場合と同様に、サーバーの状態、ディスク容量、サービスの応答時間を監視します。詳細については、『Sun Java System Messaging Server 管理ガイド』の「メッセージストアの管理」の章を参照してください。

MMP および Messenger Express Multiplexor (MEM) の保護

MMP はメッセージストアのプロキシとして機能するため、エンドユーザーデータへのアクセスを防ぎ、権限のないアクセスから保護する必要があります。ユーザー ID とパスワードは、基本的な認証機能となります。さらに、クライアントアクセスフィルタを使用すれば、ユーザーが特定のドメインや特定の IP アドレスの範囲にアクセスするのを制限できます。SMTP リレーサーバーのセキュリティを提供する方法としては、SMTP 認証または SMTP Auth (RFC 2554) をお勧めします。SMTP AUTH は、認証済みのユーザーだけに MTA を介したメール送信を許可します。詳細は、143 ページの「Authenticated SMTP を有効にする」を参照してください。

POP サービスまたは IMAP サービスの前に、MMP を別のマシンまたは別のユーザー ID のもとに配置します。フロントエンドマシンには MMP と MTA のみを配置してから、フロントエンドマシン、メールストア、および LDAP サーバー間で、セキュリティで物理的に保護されたネットワークを構築できます。

ユーザーがインターネットからログインする場合は、Messenger Express からメッセージストアへのアクセスのセキュリティには特に配慮が必要となります。一般的には、ストアはファイアウォールにより外部と分離します。さらに、HTTP アクセスサービスへの単一の接続ポイントとして機能する特別なサーバーとして、Messenger Express Multiplexor (MEM) を使用することも考えられます。MMP と同様に、MEM は、メールクライアントとの間で、暗号化されていない通信と暗号化された (SSL) 通信の両方をサポートしています。MEM は、エンドユーザーデータへのアクセスと権限のないアクセスからの保護も行う必要があります。

ログファイルを定期的に監視することで、権限のないアクセスを防ぐことができます。

ユーザー認証の計画

ユーザー認証を行うことで、ユーザーはメールクライアントへのログインとメールメッセージの取得が可能になります。ユーザー認証の方法には以下のものがあります。

- プレーンテキストと暗号化されたパスワードによるログイン
- Simple Authentication and Security Layer (SASL) による認証
- Authenticated SMTP を有効にする
- Secure Sockets Layer (SSL) による証明書ベースの認証

プレーンテキストと暗号化されたパスワードによるログイン

ユーザー ID とパスワードは、LDAP ディレクトリに保存されます。「最小の長さ」のようなパスワードのセキュリティ基準は、ディレクトリのポリシー要件で決定されます。パスワードのセキュリティ基準は、Messaging Server 管理の一部ではありません。ディレクトリサーバーのパスワードポリシーを理解するには、『Sun Java System Directory Server 配備計画ガイド』を参照してください。管理者は、メッセージング設定パラメータを設定して、プレーンテキストのパスワードを許可するかどうか、パスワードの暗号化を必須とするかどうかを決めることができます。詳細については、`service.`*xxx*`.plaintextminciper` (*xxx* は『Sun Java System Messaging Server Administration Reference』の `http`, `pop` または `imap` パラメータ) を参照してください。

プレーンテキストによるログインと暗号化されたパスワードによるログインは、どちらも POP、IMAP、および Messenger Express ユーザーアクセスプロトコルで使用できます。

Simple Authentication and Security Layer (SASL) による認証

SASL (RFC 2222) は、POP、IMAP、および SMTP ユーザーアクセスプロトコルの追加認証メカニズムとして機能します。Messaging Server は、以下の表に一覧されているユーザーアクセスプロトコルの SASL をサポートしています。

表 9-3　SASL 認証のユーザーアクセスプロトコルのサポートマトリックス

	プレーン	ログイン	CRAM-MD5	Digest-MD5	証明書	APOP
SMTP AUTH	Yes	Yes	Yes	Yes	-	-
POP	Yes	-	Yes	Yes	-	Yes
IMAP	Yes	-	Yes	Yes	-	-
HTTP (Messenger Express)	Yes	-	-	-	Yes	-

注 S

- CRAM-MD5 を使用する場合は、パスワードをプレーンテキストで LDAP ディレクトリサーバーに保存する必要があります。
- Digest-MD5 は MMP ではまだサポートされていませんが、MMP を使用しないようにすればサポートされます。
- POP を使用する場合は、パスワードをプレーンテキストで LDAP ディレクトリサーバーに保存する必要があります。

SASL を使用する場合、セッションで SSL を使用しなければ、ユーザー名とパスワードは暗号化されません (SSL の詳細については、145 ページの「SSL による暗号化」を参照)。SASL メカニズム、CRAM-MD5、DIGEST-MD5、および LOGIN は、認証情報を暗号化しますが、情報が捕捉された場合には容易に解読されてしまいます。この制約にもかかわらず、SASL は SMTP AUTH (143 ページの「Authenticated SMTP を有効にする」を参照) と組み合わせて、システムで認証されたユーザーにだけシステムをリレーしたメール送信を許可できるため、便利なものとして使用されています。たとえば、正当なユーザーが SMTP サーバーへの認証を受けると、SMTP サーバーで別のチャネルへの切り替えを設定できます。このようにすると、認証されたセッションからのメッセージは、認証されていないユーザーとは別の TCP チャネルから送られてくるメッセージとなります。内部ネットワークのユーザーからのメッセージも、受信接続の IP アドレスに基づいて、その他の発信元からのメッセージとは別のチャネルに切り替えできます。

SASL の詳細については、『Sun Java System Messaging Server 管理ガイド』の「セキュリティ設定とアクセス制御」の章を参照してください。

Authenticated SMTP を有効にする

Authenticated SMTP (SMTP AUTH とも呼ばれる) は、SMTP プロトコルを拡張したものです。Authenticated SMTP を使用すると、サーバーへのクライアント認証が可能になります。認証は、メッセージの送受信時に実行されます。Authenticated SMTP の主な用途は、悪用される可能性のあるオープンリレーを作成することなく、オフィス外のローカルユーザーがメールを送信するのを可能にすることです。クライアントは、AUTH コマンドを使用してサーバーに対する認証を行います。

Authenticated SMTP は、SMTP プロトコルによるメッセージの送信をセキュリティで保護します。Authenticated SMTP を使用する場合に、証明書に基づいたインフラストラクチャを用意する必要はありません (証明書による認証については「Secure Sockets Layer (SSL) による証明書ベースの認証」を参照)。

Authenticated SMTP を使用すると、クライアントは認証メカニズムをサーバーに提示し、認証プロトコルの交換を行うことができます。任意で、後続のプロトコル相互対話で使用するセキュリティ層とネゴシエートを行うこともできます。たとえば、メールクライアントが Authenticated SMTP をサポートしている場合は、メッセージの送信前にエンドユーザーにパスワードの入力を要求できます。

メールの送信に SMTP AUTH の使用を要求している場合は、適切なログを記録してメールが悪用されたケースを追跡できます。

Authenticated SMTP の詳細につては、『Sun Java System Messaging Server 管理ガイド』の「MTA」の章を参照してください。

Secure Sockets Layer (SSL) による証明書ベースの認証

Messaging Server は、SSL プロトコルを使用して、暗号化通信とクライアントおよびサーバーの証明書ベースの認証を行います。この節では、証明書ベースの SSL 認証について説明します。SSL 暗号化の詳細については、145 ページの「SSL による暗号化」を参照してください。

SSL は公開鍵暗号法の概念に基づいています。TLS (Transport Layer Security) は SSL のスーパーセットとして機能しますが、名前が混同されて使われています。

SSL をサポートしているサーバーには、証明書、公開鍵、非公開鍵、証明書、鍵、およびセキュリティデータベースが高レベルで必要となります。これにより、メッセージの認証、機密、完全性が確保されます。

144 ページの表 9-4 で、各クライアントアクセスプロトコルによる SSL 認証のサポートについて説明します。

表 9-4　SSL 認証のサポートマトリックス

	MMP による SSL	代替ポートでの MMP による SSL	SSL	代替ポートでの SSL
SMTP	Yes	Yes	Yes	Yes
POP	-	Yes	-	Yes
IMAP	Yes	Yes	-	Yes
Messenger Express (HTTP)	Yes (Messenger Express Multiplexor による)	Yes (Messenger Express Multiplexor による)	Yes	Yes

SMTP、POP、IMAP、および HTTP プロトコルは、クライアントとサーバーが SSL なしで通信を開始したあと、"start TLS" コマンドを使用して SSL 通信に切り替える方法を提供します。クライアントとサーバーが "start TLS" を実装していない場合は、SMTP、POP、IMAP、および HTTP サーバーが SSL のみを代替ポートで使用するように設定することもできます。

SSL による認証を行うには、メールクライアントはサーバーとの SSL セッションを確立し、ユーザーの証明書をサーバーに提出します。その後、サーバーが、提出された証明書の信頼性を評価します。証明書の信頼性が確認されると、そのユーザーは認証済みであるとみなされます。

SSL を認証用途で使う場合、Messaging Server 用のサーバー証明書を入手する必要があります。この証明書は、使用するサーバーの識別情報をクライアントや他のサーバーに提供します。サーバーには複数のサーバー証明書を用意しておき、証明書自身を識別することができます。サーバーには、信頼できる認証局 (CA) の証明書を必要な数だけインストールして、クライアントの認証に使用できます。

SSL の詳細については、『Sun Java System Messaging Server 管理ガイド』の「セキュリティとアクセス制御」の章を参照してください。

メッセージ暗号化戦略の計画

この節では、暗号化とプライバシーソリューションについて説明します。以下のトピックについて説明しています。

- SSL による暗号化
- 署名され暗号化された S/MIME

SSL による暗号化

SSL は、IMAP、HTTP、および SMTP のアプリケーションレイヤの下のプロトコルレイヤとして機能します。Messaging Server とそのクライアント間、および Messaging Server と他のサーバー間におけるメッセージの転送が暗号化される場合は、通信が盗聴される危険性はほとんどありません。また、接続しているクライアントとサーバーが認証済みの場合は、侵入者がそれらのクライアントになりすます (スプーフィングする) 危険性もほとんどありません。

メッセージ送信でエンドツーエンドの暗号化を行うには、SSL を SMTP、IMAP、および HTTP プロトコルとともに使用する必要があります。

注

SSL 接続の設定によりパフォーマンスのオーバーヘッドが生じると、サーバーへの負担となります。メッセージングシステムの設計とパフォーマンスの分析を行う際には、セキュリティ要件とサーバーのパフォーマンスのバランスをとる必要があります。

暗号化の用途で SSL を使用する場合は、ハードウェア暗号化アクセラレータをインストールすることでサーバーのパフォーマンスを向上させることができます。一般的に、暗号化アクセラレータは、サーバーマシンに常設されたハードウェアボードとソフトウェアドライバで構成されます。

HTTP/SSL (HTTPS) を使用したクライアントとサーバー間の SSL 接続プロセスは、以下のようになります。

1. クライアントが HTTPS を使用して接続を開始する。クライアントが、使用する秘密鍵アルゴリズムを指定する
2. サーバーが認証のための証明書を送り、使用する秘密鍵アルゴリズムを指定する。クライアントと共通の最も強力なアルゴリズムが指定される。秘密鍵が一致しない場合 (たとえば、クライアントの鍵が 40 ビットのみで、サーバーが 128 ビットの鍵を要求している場合)、その接続は拒否される
3. サーバーがクライアント認証を要求するように設定されている場合、この時点でクライアントに証明書が要求される
4. クライアントは、サーバーの証明書の正当性をチェックし、以下の内容を確認する
   - 期限が切れていない
   - 既知の署名された認証局
   - 有効な署名
   - 証明書のホスト名が HTTPS 要求のサーバーのホスト名と一致している

SSL 符号化方式

符号化方式とは、暗号化プロセスでデータの暗号化と解読に使用されるアルゴリズムのことです。各符号化方式によって強度が異なります。つまり、強度の高い符号化方式で暗号化したメッセージほど、承認されていないユーザーによる解読が困難になります。

符号化方式では、キーをデータに適用することによってデータを操作します。一般的に、符号化方式で使用するキーが長いほど、適切な解読キーを使わずにデータを解読することが難しくなります。

クライアントは、Messaging Server と SSL 接続を開始するときに、サーバーに対して、希望する暗号化用の符号化方式とキー長を伝えます。暗号化された通信では、両方の通信者が同じ符号化方式を使用する必要があります。一般的に使用される符号化方式とキーの組み合わせは数多くあります。そのため、サーバーが柔軟な暗号化方式をサポートしている必要があります。符号化方式の詳細については、『Sun Java System Messaging Server 管理ガイド』の「セキュリティ設定とアクセス制御」の章を参照してください。

署名され暗号化された S/MIME

署名され、暗号化されたメッセージは、Secure/Multipurpose Internet Mail Extensions (S/MIME) メッセージと呼ばれます。S/MIME は、クライアント間の通信をセキュリティで保護する手段です

S/MIME を使用すると、送信者は送信する前にメッセージを暗号化できます。受信者は、受信した暗号化されたメッセージを保存し、あとで読むときだけそれを解読することができます。S/MIME を使用するのに Messaging Server で特別な設定やタスクは必要ありません。これは完全にクライアントによる動作となります。これは SSL とは異なり、エンドツーエンドの暗号化機能を提供します。S/MIME の設定の詳細については、クライアントのマニュアルを参照してください。

セキュリティに関する誤解

この節では、配備のセキュリティニーズに対して逆効果になる、メッセージングに関する典型的な誤解について説明します。

- **製品名とバージョン名を隠す**

  製品名とバージョン名を隠したとしても、せいぜい通常のアタッカーの邪魔をする程度です。最悪の場合は、管理者にセキュリティに関する誤った感覚を与えることになり、本当のセキュリティ問題の追跡を怠るという結果になりかねません。

  事実、製品情報とバージョン番号が削除されると、ソフトウェアの識別ができなくなるため、ベンダーのサポート部門がソフトウェアの問題を検証するのが困難となります。

  ハッカーが入念な行動を取ることはほとんどありません。特に、SMTP サーバーの既知の脆弱性を攻撃するときには、あらゆる SMTP サーバーにアクセスを試みます。

  製品名やバージョン番号を隠されても、知識があれば、プロトコルの動きに注目して、ベンダー名とバージョンを判断することもできます。

- **内部マシン名を隠す**

  内部 IP アドレスとマシン名を隠すことで、以下の行為が困難になります。

  - 悪用またはスパムの追跡
  - メールシステムの設定エラーの診断
  - DNS 設定エラーの診断

  知識のあるアタッカーであれば、一度ネットワークに侵入する方法を見つければ、マシンのマシン名と IP アドレスを簡単に見つけ出します。

- **SMTP サーバーの EHLO をオフにする**

EHLO がない場合は、以下のことができなくなります。

- NOTARY
- TLS ネゴシエーション
- メッセージサイズのプリエンプティブ制御

EHLO を使用すると、SMTP クライアントは、制限の有無と、この応答を受けるとすぐに制限を超えたメッセージの送信を停止するかどうかを判断します。ただし、EHLO がオフになっているため HELO を使用しなければならない場合は、送信側の SMTP サーバーはメッセージデータ全体を送信し、その後メッセージサイズが制限を超えているため拒否されたことを通知されます。その結果、処理サイクルとディスク容量の無駄が発生します。

- Network Address Translation (NAT)

  NAT を一種のファイアウォールとして使用する場合は、システム間でエンドツーエンドの接続を行うことはできません。その代わり、中間に第三のノードを置くことになります。この NAT システムはミドルマンとして機能し、潜在的なセキュリティホールとなります。

その他のセキュリティリソース

セキュリティで保護されたメッセージング配備の詳細については、Computer Emergency Response Team (CERT) Coordination Center のサイトを参照してください。

http://www.cert.org

第 10 章

サービス可用性に向けた計画

この章では、配備に適切なサービス可用性のレベル決定方法について説明します。サービス可用性のレベルは、採用したハードウェアとソフトウェアインフラストラクチャ、および実際の保守の方法に関係しています。この章では、いくつかの選択肢とそのメリットおよびコストについて説明します。

この章には、以下の節があります。

Automatic System Reconfiguration (ASR) の概要

単に高可用性 (HA) ソリューションを評価するだけでなく、ASR を可能にするハードウェアの配備についても検討する必要があります。

ASR は停止時間に関連するハードウェア障害を最小限にするためのプロセスです。サーバーに ASR 機能がある場合は、ハードウェアの個別のコンポーネントに障害が発生しても、停止時間を最小限にとどめることが可能になります。ASR により、サーバーの自動再起動と、障害の発生したコンポーネントが交換されるまでそれを停止しておくことが可能になります。欠点は、障害の発生したコンポーネントがサービスから排除される結果、システムのパフォーマンスが低下することです。たとえば、CPU に障害が発生すると、マシンは残りの CPU を使用して再起動されます。システムの I/O ボードまたはチップに障害が発生した場合は、システムの I/O ボードが減少するか、代わりの I/O パスが使用されます。

さまざまな Sun SPARC システムが、さまざまなレベルの ASR をサポートしています。高いレベルの ASR まではサポートしていないシステムもあります。当然ながら、高い ASR 機能を持つサーバーはその分コストも高くつきます。ソフトウェアに高可用性がない場合、コストには制約がないものとすれば、データ格納用にはハードウェアに高い冗長性と ASR 機能を持たせたマシンを選択します。

高可用性モデルの理解

さまざまなタイプの高可用性モデルを Messaging Server として使用できます。一般的によく使用されるモデルに、以下の 3 つがあります。

- 非対称型 (ホットスタンバイ)
- 対称型
- N+1 (N プラス 1)

以下の節で、これらのモデルについてそれぞれ詳しく説明します。

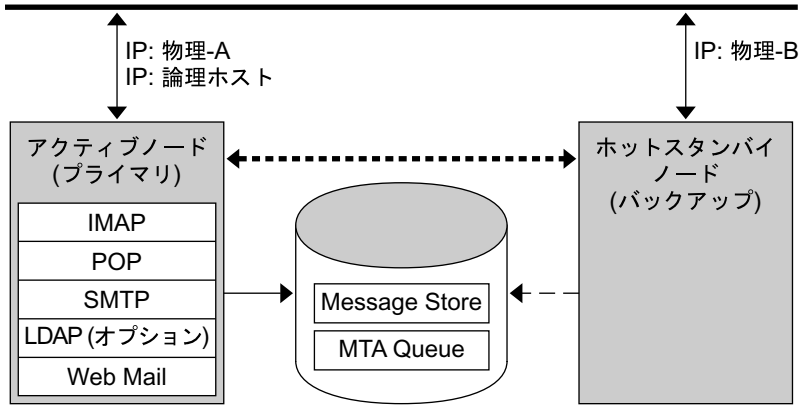
注	異なる高可用性製品が異なるモデルをサポートしている場合もあれば、サポートしていない場合もあります。対応する製品の高可用性に関するマニュアルを参照して、どのモデルがサポートされているかを確認してください。

非対称型

基本的な非対称型または「ホットスタンバイ」型の高可用性モデルは、クラスタ化された 2 つのホストマシンまたは「ノード」で構成されます。1 つの論理 IP アドレスおよび関連するホスト名が両方のノードに指定されます。

このモデルでは、常に 1 つのノードだけがアクティブとなります。バックアップノードまたはスタンバイノードは、大半の時間はアイドル状態のままとなります。両ノード間で単独の共有ディスクアレイが構成され、アクティブまたは「プライマリ」なノードの支配下となります。メッセージストアパーティションおよびメッセージ転送エージェント (MTA) キューは、この共有ボリュームに置かれます。以下の図は非対称型モデルの例です。

図 10-1　非対称型高可用性モデル



前の図には、物理 -A と物理 -B の 2 つのノードがあります。フェイルオーバーの前は、物理 -A がアクティブなノードです。フェイルオーバーが行われると、物理 -B がアクティブなノードとなり、共有ボリュームは物理 -B の支配下に切り替わります。すべてのサービスが物理 -A で停止し、物理 -B で開始されます。

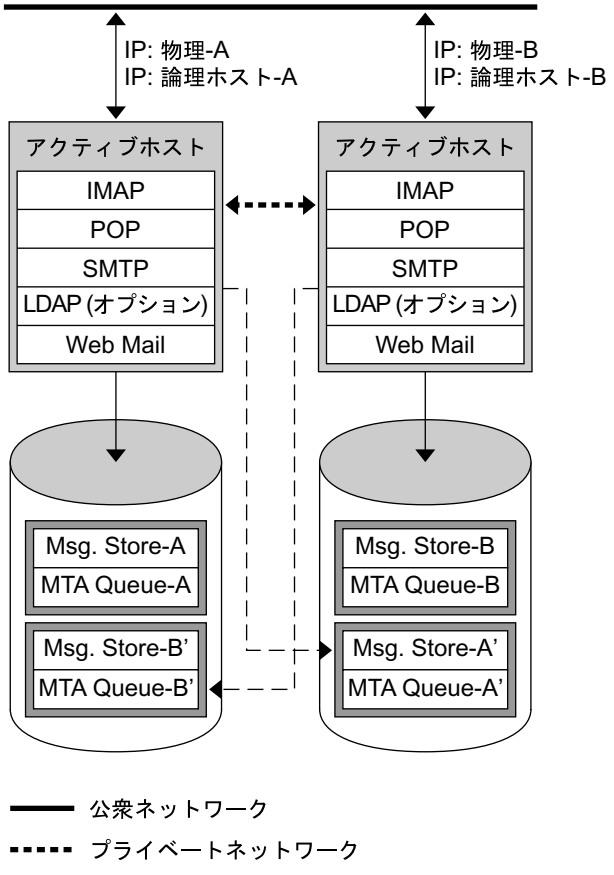
このモデルのメリットは、バックアップノードがプライマリノードの予備としてだけ使用されることです。また、フェイルオーバー時に、バックアップノードでリソースの競合が起こることもありません。しかし、このモデルではバックアップノードがほとんどの時間でアイドル状態にあり、そのリソースがほとんど利用されないことになります。

対称型

基本的な対称型または「デュアルサービス」型の高可用性モデルは、それぞれ固有の論理 IP アドレスを持つ 2 台のホストマシンで構成されます。それぞれの論理ノードは 1 つの物理ノードに関連付けられており、それぞれの物理ノードが、2 つのストレージボリュームを持つ 1 つのディスクアレイを制御します。1 つのボリュームがローカルメッセージ格納パーティションと MTA キューとして使用され、もう 1 つのボリュームは他方のメッセージ格納パーティションと MTA キューのミラーイメージとなります。

以下の図は、対称型高可用性モデルの例です。両方のノードが共にアクティブで、それぞれのノードが他方のバックアップとして機能します。正常な状態では、それぞれのノードが Messaging Server の 1 つのインスタンスだけを実行します。

図 10-2　対称型高可用性モデル



フェイルオーバーが起こったときには、障害の発生したノードのサービスが停止され、バックアップノードで再開されます。この時点で、バックアップノードは両方のノードの Messaging Server を実行し、2 つの個別ボリュームを管理しています。

このモデルのメリットは、両方のノードが同時にアクティブとなり、マシンリソースが完全に利用されることです。ただし、障害が発生している間は、バックアップノードで両方のノードの Messaging Server のサービスを実行するため、リソースの競合が多くなります。したがって、障害の発生したノードをできるだけ素早く修復し、サービスをデュアルサービスの状態に復元する必要があります。

このモデルでは、バックアップストレージアレイも提供されます。ディスクアレイに障害が発生した場合は、バックアップノードのサービスが冗長イメージを取り出します。

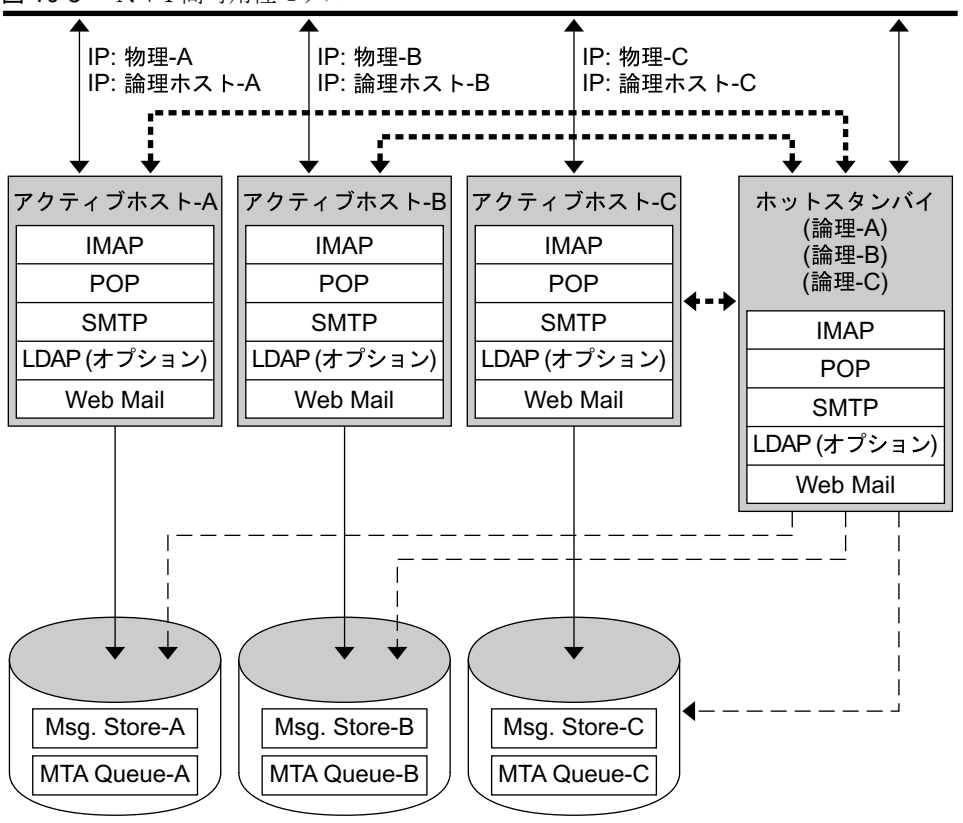
対称型モデルを構成するには、共有ディスクに共有バイナリをインストールする必要があります。ただし、共有バイナリをインストールすることにより、ローリングアップグレード、すなわち、Messaging Server のパッチリリース中にシステムの更新を行う機能が実行できなくなる場合があります (この機能は将来のリリースで対応を検討中)。

N+1 (N プラス 1)

N + 1 または "N プラス 1" モデルは、マルチノードの非対称型構成で運用します。N 個の論理ホスト名と N 個の共有ディスクアレイが必要です。1 つのバックアップノードが、他のすべてのノードに対するホットスタンバイとして確保されます。バックアップノードは、N 個のノードから同時に Messaging Server を実行できます。

154 ページの図 10-3 は、基本的な N + 1 高可用性モデルの例です。

図 10-3　N + 1 高可用性モデル



1 つまたは複数のノードでフェイルオーバーが起こったときは、バックアップノードが障害の発生したノードの責任を引き受けます。

N + 1 モデルのメリットは、サーバーの負荷が複数のノードに分散され、すべてのノードの障害に対して 1 つのバックアップノードだけで対処できることです。したがってマシンのアイドル比率は、単独の非対称型モデルの 1 対 1 に対して、N 対 1 となります。

N + 1 モデルを構成するには、バイナリをローカルディスク (対称型モデルの共有ディスクのことではない) にだけインストールする必要があります。すべての対称型モデル、1 + 1 または N + 1 非対称型モデルあるいは対照型モデルの対称型高可用性ソリューションでは、現在の Messaging Server のインストールとセットアップのプロセスにより、バイナリは強制的に共有ディスクに置かれます。

高可用性モデルの選択

以下の表で、それぞれの高可用性モデルのメリットとデメリットをまとめています。この情報を参考にして、配備にふさわしいモデルを決定してください。

表 10-1 高可用性モデルのメリットとデメリット

モデル	メリット	デメリット	推奨ユーザー
非対称型	• 簡単に構成できる • バックアップノードが 100% 予約される	マシンリソースが完全に利用されない	将来拡張を予定している小規模なサービスプロバイダ
対称型	• システムリソースを有効に利用できる • 可用性が高い	バックアップノードでリソースの競合が起こる 高可用性実現のために完全な冗長ディスクが必要	サーバー障害時のパフォーマンスの低下を許容できる小規模な企業
N + 1	• 負荷分散される • 拡張性が高い	管理と構成が複雑	リソース競合のない分散を必要とする大規模なサービスプロバイダ

システム停止時間の計算

以下の表で、システム障害によりメッセージサービスが利用できなくなる確率について説明します。ここでの計算は、システムクラッシュまたはサーバーのハングアップによるサーバーの停止が平均して 3 か月間に 1 日、ストレージデバイスの停止が 12 か月間に 1 日の割合で発生することを想定しています。両方のノードが同時に停止する可能性はきわめて小さいため、計算では無視しています。

表 10-2 システム停止時間の計算

モデル	サーバーが停止する確率
単独のサーバー (高可用性なし)	Pr (停止) = (システムの停止 4 日 + ストレージの停止 1 日)/365 = 1.37%
非対称型	Pr (停止) = (システムの停止 0 日 + ストレージの停止 1 日)/365 = 0.27%
対称型	Pr (停止) = (システムの停止 0 日 + ストレージの停止 1 日)/365 = 0.27%
N + 1 非対称型	Pr (停止) = (システムの停止 5 時間 + ストレージの停止 1 日)/(365xN) = 0.33%/N

製品の参照情報

Messaging Server がサポートする高可用性モデルの詳細については、以下の製品マニュアルを参照してください。

- Sun Cluster
  - Sun Cluster Concepts Guide for Solaris OS
  - Sun Cluster Data Services Planning and Administration Guide for Solaris OS
  - Sun Cluster Overview for Solaris OS
  - Sun Cluster System Administration Guide for Solaris OS
- Veritas Cluster Server
  - Veritas Cluster Server User's Guide

リモートサイトフェイルオーバーの理解

リモートサイトフェイルオーバーは、プライマリサイトに致命的な障害が発生した場合に、そのプライマリサイトに WAN で接続されているサイトでサービスを開始する機能です。リモートサイトフェイルオーバーにはいくつかの形式があり、それぞれにコストが異なります。

リモートサイトフェイルオーバーでは、すべてのケースでサーバーとストレージを追加して、リモートサイトにインストールおよび設定された、サービスのユーザー負荷のすべてまたは一部を処理する能力を持つようにする必要があります。すべてまたは一部というのは、顧客によっては優先するユーザーとそうでないユーザーがいることを意味します。ISP でも企業でも、そのような状況が起こります。ISP には、この機能のために割増料金を支払うユーザーがいます。企業では、全従業員に電子メールの機能を提供している部門内で、ユーザーによってはそのサポートが高くついている場合があるかもしれません。たとえば、カスタマサポートに直接関わるユーザーのメールに対してリモートサイトフェイルオーバーを選択した場合でも、製造ラインで勤務する従業員には、リモートサイトフェイルオーバーを用意しないというケースが考えられます。リモートハードウェアは、このようにリモートサイトフェイルオーバーメールサーバーにアクセスを許可されたユーザーの負荷を処理できます。

ユーザーベースの使用率だけを制限すると、必要な冗長サーバーとストレージハードウェアの数を減らすことができますが、フェイルバックの設定と管理も複雑になります。そのようなポリシーはまた、長期的には予期しない別の影響をユーザーに与えます。たとえば、ドメインメールルーターが 48 時間にわたって消失した場合、インターネット上の他の MTA ルーターがそのドメイン宛てのメールを保持します。ある時点で、サーバーがオンラインに復帰したときに (うまくいけば DoS による障害を受けることもなく)、そのメールが配信されます。さらに、すべてのユーザーをフェイルオーバーリモートサイトに設定していない場合は、MTA が起動して設定されていな

いユーザーに対して永続的なエラー (バウンス) が返されます。最後に、すべてのユーザーを受け入れるようにメールを設定している場合は、すべてのユーザーをフェイルバックするか、フェイルオーバーがアクティブな間使用できないアカウント宛てのメールを保持し、フェイルバックが起こったらそれを本来の配信の流れに戻すように MTA ルーターを設定する必要があります。

考えられるリモートサイトフェイルオーバーのソリューションには、以下のようなものがあります。

- **単純でコストのかからないシナリオ**：リモートサイトの接続に広帯域幅の大規模ネットワークを使用しません。十分な規模のハードウェアを必ずしも使用する必要はありません。実際のところ、ハードウェアは当面の間は他の目的に使用できます。プライマリサイトからのバックアップがリモートサイトに対して定期的に提供されますが、必ずしも復元の必要はありません。予想される問題点としては、古いデータをオンラインに戻す際に重要なデータが失われたり、かなりの遅れが生じることです。プライマリサイトで障害が発生したときには、手動でネットワークを変更してサービスを開始し、続いて `imrestore` プロセスを開始します。サービスが起動されると、ファイルシステムの復元が開始されます。
- **より複雑で、コストのかかるソリューション**：Veritas および Sun の市販ソフトウェアソリューションでは、ローカル (プライマリ) ボリュームで発生するすべての書き込みがリモートサイトにも書き込まれます。通常の製品では、リモートサイトはプライマリサイトと共にロックステップかそれに近い状態になります。プライマリサイトで障害が発生した場合は、セカンダリサイトがネットワーク設定をリセットし、データをほとんど失うことなくサービスを提供できます。このシナリオでは、テープから復元する意味はありません。プライマリサイトの障害の前に切り替えられなかったデータは、少なくともフェイルバックが起きるか、MTA キューデータの場合には手動による介入が行われるまで、失われることになります。Veritas Site HA ソフトウェアは、プライマリサイトの障害を検出し、ネットワークをリセットしてサービスを起動する用途でよく使われますが、これはより高いレベルのデータ保管には必要ありません。サーバーがデータをコピーするのに負荷と待ち時間が大きく増加するため、このソリューションにはプライマリサイトで大量のハードウェアを必要とします。
- **最も実現性の高いソリューション**：このソリューションでは、データのコピーがメッセージストアサーバーで行われることを除いて、ソフトウェアによるリアルタイムデータコピーソリューションと本質的には同じものです。日立データシステムズ (HDS) のアレイがこの機能を持っています。日立のアレイには、格納サーバーにほとんど、あるいはまったく影響を与えずに、アレイ間でこのデータコピーを実行する機能があります。HDS のアレイはサイズが大きく、このソリューションを導入するための基本コストも、Sun StorEdge™ T3 や 3000 のアレイよりも高くつきます。さらに、HDS をフルに利用したとしても、M バイトあたりのコストも高くなります。大容量のストレージが必要な場合は、サーバーハードウェアを節約するために HDS がコピー処理を行うことを許可すれば、ストレージの追加コストを、少なくともある程度までは調整できます。

ハードウェアやソフトウェアをはじめ、管理コスト、電力費、光熱費、ネットワークコストまで、これらのソリューションでは、さまざまなコストが発生します。これらのコストはすべてそのまま計算に入れて、数字をはじき出します。そうしなければ、めったに起こらない処理を実行するときの思いがけない費用や、停止時間による直接のコスト、データ損失によるコストなど、いくつかのコストを算出するのが困難になります。このような種類のコストを正確に算定するのは不可能です。顧客によっては、停止時間とデータの損失は代償が高くつくか、まったく受け入れられません。その他の顧客にとっては不愉快以上のものではないでしょう。

リモートサイトフェイルオーバーを行う場合には、リモートディレクトリが少なくとも最新のもので、メッセージデータの復元が可能な状態にあることも必要です。リモートサイトにリストアメソッドを使用する場合は、ディレクトリが完全に復元されてからメッセージを復元する必要があります。また、ユーザーをシステムから削除した場合、ディレクトリ内でそのユーザーに無効のタグがつけられるだけなのはやむをえません。ユーザーのデータがあるメッセージバックアップテープが使用される限り、そのユーザーをディレクトリから削除してはなりません。

リモートサイトフェイルオーバーについての質問

以下の質問を参考にして、リモートサイトフェイルオーバーの計画を立ててください。

- サイトで必要とする応答性のレベルはどの程度か

  組織によっては、プライマリサイトで障害が発生したときに、手動処理のスクリプトセットで十分対応可能な場合もあります。短時間 (数分間) のうちにリモートサイトがアクティブになる必要がある組織もあります。そのような組織では、Veritas リモートサイトフェイルオーバーソフトウェアかそれに相当する機能を持ったその他のソフトウェアが必要です。

注

ローカル HA 用の Sun Cluster とリモートサイトフェイルオーバー用の Veritas ソフトウェアを併用しないでください。Sun Cluster は現時点ではリモートサイトフェイルオーバーをサポートしていません。

また、プライマリサイトからバックアップサイトへの自動フェイルオーバーをソフトウェアに許可しないでください。その場合、セカンダリサイトからプライマリサイトの障害が誤って検出される可能性がかなり高くなります。このようなケースでは、ソフトウェアにプライマリサイトを監視させ、障害を検出したときに警告を出させるように設定します。次に、バックアップサイトへの自動フェイルオーバープロセスを開始する前に、障害が実際に発生していることを確認します。

- どれぐらいのデータを保存し、どの程度の速さで利用可能にする必要があるか

これは単純な質問のようですが、細分化されて回答の幅は広くなります。シナリオには、簡単なものからほとんど完全なものまであり、ハードウェア、ネットワークデータインフラストラクチャ、保守のコストの面でも大きな違いがあります。

第 11 章

インストール前の考慮事項と手順

この章では、Messaging Server をインストールする前に検討しなければならない考慮事項と、実行しなければならない手順について説明します。Java Enterprise System インストーラに関する指示と実行については、『Sun Java Enterprise System 2004Q2 インストールガイド』を参照してください。

この章には、以下の節があります。

インストール時の考慮事項

この節では、Messaging Server のインストールの準備のための考慮事項について説明します。

- **リソースの競合** : サーバー間のリソースの競合を回避するには、Messaging Server をインストールしたホストとは別のホストに Directory Server をインストールしてください。
- **インストール権限** : Messaging Server をインストールするには、スーパーユーザーとしてログオンする必要があります。
- Messaging Server Base Directory: Messaging Server は、*msg_svr_base* (たとえば `/opt/SUNWmsgsr`) と呼ばれるディレクトリにインストールされます。このディレクトリは、既知のファイル配置構造 (ファイルディレクトリパス) を持っています。

- **サーバーのアップグレード** : Messaging Server ホストにその他のコンポーネント (Web Server、Directory Server、Identity Server、および管理サーバー) をインストールしない場合は、これらのコンポーネントのアップグレードは不要で、Messaging Server は問題なく動作します。同じマシンにその他のコンポーネントがインストールされている場合は、Messaging Server と共にそのコンポーネントをアップグレードする必要があります。
- **ポート番号の競合** : 同じマシンに特定の製品をインストールすると、ポート番号の競合が起こる場合があります。以下のテーブルで、ポート番号が競合する可能性をまとめています。

表 11-1 可能性のあるポート番号の競合

ポート番号の競合	コンポーネント	コンポーネント
143	IMAP サーバー	MMP IMAP プロキシ
110	POP3 サーバー	MMP POP3 プロキシ
993	SSL を使用した IMAP	SSL を使用した MMP IMAP プロキシ
80	Identity Server (Web サーバーポート)	Messenger Express

可能であれば、ポート番号が競合する製品は別のホストにインストールします。それができない場合は、競合する製品のいずれかでポート番号を変更する必要があります。ポート番号を変更するには、`configutil` ユーティリティを使用します (使い方については『Sun Java System Messaging Server Administration Reference』を参照)。

以下の例では、`service.http.port configutil` パラメータを使用して、Messenger Express HTTP ポート番号を `8080` に変更しています。

```
configutil -o service.http.port -v 8080
```

インストールワークシート

Messaging Server をインストールするときに、以下のインストールワークシートを使用して記録をつけておくと、インストールプロセスで役立ちます。これらのインストールワークシートは、Messaging Server を何度もインストールしたり、アンインストールしたり、アップグレードのためにアップグレードしたりする際に再使用できます。

ヒント インストール中に指定したすべてのポート番号と、そのポート番号を使用する特定のコンポーネントを記録しておきます。

ワークシートには以下のものがあります。

- Directory Server インストール用ワークシート
- 管理サーバー初期実行時設定用ワークシート

Directory Server インストール用ワークシート

Java Enterprise System インストーラ、または以前の Directory Server インストールにより、Directory Server をインストールできます。Directory Server のインストール情報と設定パラメータを以下の表に記録します。管理サーバーと Messaging Server のインストールと設定を行うときや、Messaging Server の初期設定を行うときに、これらのパラメータが必要となります。その他の情報については、『Sun Java System Messaging Server 管理ガイド』を参照してください。

表 11-2 Directory Server インストールパラメータ

パラメータ	説明	例	設定対象	実際の設定値
Directory Installation Root	Directory Server ホストにあるディレクトリで、サーバープログラム、設定、保守、および情報のファイルの格納専用に使用される	`/var/opt/mps/serverroot`	`comm_dssetup.pl` Perl スクリプト	
Host	完全指定ドメイン名 完全指定ドメイン名は、ホスト名とドメイン名の 2 つの部分から構成される	`svr1.west.sesta.com`	管理サーバー設定	

表 11-2　Directory Server インストールパラメータ (続き)

パラメータ	説明	例	設定対象	実際の設定値
LDAP Directory Port Number	LDAP ディレクトリサーバーのデフォルトは 389	389	管理サーバー設定と Messaging Server 設定	
Administrator ID and Password	情報の設定に責任を持つ管理者 管理者のパスワード	Admin PaSsWoRd	管理サーバー設定	
User and Group Tree Suffix	ユーザーとグループのデータが格納されるディレクトリツリーの最上部の LDAP エントリの識別名	o=usergroup	comm_dssetup.pl Perl script	
Directory Manager DN and Password	UNIX の スーパーユーザー に相当する権限を持ったディレクトリ管理者 通常この管理者は、ユーザーとグループのデータに責任を持つ ディレクトリマネージャのパスワード	cn=Directory Manager pAsSwOrD	comm_dssetup.pl Perl スクリプトと Messaging Server 設定	
Administration Domain	管理制御の対象範囲	System Lab	管理サーバー設定	

管理サーバー初期実行時設定用ワークシート

Java Enterprise System を使用して管理サーバーの初期実行時設定プログラムを実行するときは、インストールパラメータを以下の表に記録します。これらのパラメータの中には、Messaging Server 初期実行時設定に必要なものがあります。いくつかの質問項目については、163 ページの「Directory Server インストール用ワークシート」も参照してください。

表 11-3　管理サーバー初期実行時設定プログラムのパラメータ

パラメータ	説明	例	実際の設定値
Fully Qualified Domain Name	ホストマシンの完全指定ドメイン名	svr1.west.sesta.com	

表 11-3 管理サーバー初期実行時設定プログラムのパラメータ (続き)

パラメータ	説明	例	実際の設定値
Server Root Definition	管理サーバーがインストールされる root ディレクトリで、サーバープログラム、設定、保守、および情報ファイルの格納専用に使用される	`/var/opt/mps/serverroot`	
UNIX System User	システムユーザーに特定の権限を指定することで、ユーザーが実行するプロセスに適切な許可を与えることができる。値は常に `root` となる	`root`	
UNIX System Group	特定の UNIX ユーザーが属するグループ。値は常に `other` となる	`other`	
Configuration Directory Server	「Directory Server インストール用ワークシート」で指定されるホストとポート	Host `svr1.west.sesta.com` Port `390`	
Configuration Directory Server Administrator and Password	「Directory Server インストール用ワークシート」で指定される管理者 ID 管理者 ID のパスワード	`Admin` PaSsWoRd	
Administration Domain	管理制御の対象範囲 Messaging Server と Directory Server を同じマシンにインストールしている場合は、「Directory Server インストール用ワークシート」で同じ管理ドメインを選択する必要がある	System Lab2	
Administrative Server Port	管理サーバー専用の固有のポート番号	`5555`	

設定する Messaging Server コンポーネントの選択

Messaging Server ソフトウェアをインストールするときに、Java Enterprise System インストーラによりすべての Messaging Server がインストールされます。次に、Messaging Server 設定プログラムを使用して、Messaging ホスト上で適切な Messaging Server コンポーネント (MTA、メッセージストア、Messenger Express、MMP) を選択します。

以下の表で、それぞれのタイプのメッセージングホストでインストールする必要のあるコンポーネントを示します。

表 11-4 Messaging Server で設定するコンポーネントの選択

設定するメッセージングホストのタイプ	設定プログラムで選択されるコンポーネント
MTA リレー	メッセージ転送エージェント
メッセージストア (バックエンド)	メッセージ転送エージェント、メッセージストア、Messenger Express 注 : 設定の終了後、MEM プロキシの保存を設定する必要がある
Messenger Express (フロントエンドのみ、保存または SMTP の機能なし)	Messenger Express、Messaging Multiplexor 注 : Messenger Express だけを設定する場合は、メッセージストアと MTA を選択するか、少なくとも既存の MTA を指定する
Messenger Multiplexor (フロントエンドのみ、保存または SMTP の機能なし)	Messaging Multiplexor

注 LMTP 配信メカニズムを設定するには、リレーマシンとバックエンドストアの両方の設定が必要です。LMTP 設定の指示については、『Sun Java System Messaging Server 管理ガイド』を参照してください。

sendmail デーモンを無効にする

Messaging Server をインストールする前に、sendmail デーモンを無効にしておく必要があります。Messaging Server SMTP サーバーが実行する Dispatcher には、ポート 25 を割り当てる必要があります。ポート 25 で sendmail デーモンが実行されていると、Dispatcher をポート 25 に割り当てることができません。

➤ sendmail デーモンを無効にするには

1. /etc/init.d ディレクトリに移動します。

```
cd /etc/init.d
```

2. sendmail が実行されている場合は、停止します。

```
./sendmail stop
```

3. /etc/default/sendmail に MODE="" を追加します。

   sendmail ファイルが存在しない場合は、ファイルを作成し、MODE="" を追加します。

   ユーザーが誤って sendmail start を実行したり、パッチにより sendmail が再起動されたりした場合でも、この修正を追加することで、sendmail がデーモンモードに移行しなくなります。

用語集

このガイドで使用されている用語の完全なリストについては、『Java Enterprise System Glossary』(http://docs.sun.com/doc/816-6873) を参照してください。

索引

J

L

M

N

O

P

R

S

T

V

W

し

す

せ

そ

た

ち

つ

て

と

な

に

ね

は

ひ

ふ

へ

ほ

ま

め

も

ゆ

り

る

ろ

わ