Sun Java™ System

Calendar 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: 817-7090

目 次

表目次 **7**

図目次 **9**

はじめに **11**

表記規則 12

第 1 章 Calendar Server について **17**
Calendar Server の概要 17
Calendar Server によるビジネスニーズへの対応 20
Calendar Server 配備の可用性を高める方法 21
Calendar Server とともに Portal Server を使用する方法 21
配備プロセスについて 22
配備とアーキテクチャの設計 22
配備の目的 22
Calendar Server 配備チーム 23
Calendar Server のエンドユーザー 24
必要なエンドユーザーのパフォーマンス 24
開発およびカスタマイズ 25
プロトタイプとテスト 26
本稼動システムのロールアウト 26

第 2 章 要件の分析 **27**
配備目標の特定 27
ビジネス要件 28
運用要件 28
企業文化と社内力学 28

技術要件 29
既存の使用パターンのサポート 29
サイトの分散 29
ネットワーク 29
既存のインフラストラクチャ 30
従業員のサポート 30
財務要件 31
サービスレベル契約 (SLA) 31
プロジェクト目標の決定 32
拡張計画 32
総所有コストの理解 32

第 3 章 ネットワークインフラストラクチャの必要性の決定 35
既存のネットワークについて 35
ネットワークインフラストラクチャコンポーネントについて 36
ルーターとスイッチ 36
ファイアウォール 37
ロードバランサ 37
ストレージエリアネットワーク (SAN) 38
DNS 39
ネットワークインフラストラクチャレイアウトの計画 39
非武装地帯 (DMZ) 39
イントラネット 40
内部ネットワーク 41
プロキシ 41
ファイアウォールの設定 42
モバイルユーザー 42

第 4 章 Calendar Server の構成の計画 43
Calendar Server の考慮事項 43
単一サーバーの最小構成 46
ネットワークフロントエンドとデータベースバックエンドのサーバー構成 48
複数のフロントエンドサーバーとバックエンドサーバーの構成 51

第 5 章 Calendar Server スキーマおよびプロビジョニングのオプションについて 53
Calendar スキーマの選択について 53
使用するスキーマの決定 54
LDAP Schema 1 54
Schema 2 (ネイティブモード) 55
Schema 2 互換モード 56
Calendar Server プロビジョニングツールについて 56
LDAP プロビジョニングツール 57

ユーザー管理ユーティリティ 57
プロビジョニングツールのオプションの比較 57

第 6 章 安全な Calendar Server の設計 59
セキュリティ戦略の作成 59
物理的セキュリティ 61
サーバーのセキュリティ 61
ネットワークのセキュリティ 61
Calendar のセキュリティの概要 62
セキュリティ戦略の監視 63
ユーザー認証の計画 63
プレーンテキストと暗号化パスワードによるログイン 64
SSL (Secure Sockets Layer) を使用する証明書ベースの認証 64
セキュリティの誤解について 65
他のセキュリティリソース 66

第 7 章 インストール前の考慮事項 67
Calendar Server のインストール 67
設定が必要な Calendar Server コンポーネント 68
Calendar Server の管理者の計画 69
Calendar Server 管理者 (calmaster) 69
Calendar Server ユーザーおよびグループ 69
スーパーユーザー (root) 70
ホストしているドメインの計画 70
インストール後の設定 72

用語集 73

索引 75

表目次

表 1-1　企業にとっての Calendar Server の利点 20
表 2-1　総所有コストの考慮事項 33
表 5-1　Calendar Server のプロビジョニングメカニズム 58
表 7-1　設定が必要な Calendar Server コンポーネント 68

図目次

図 4-1　単一サーバーの Calendar Server の最小構成 46
図 4-2　ネットワークフロントエンドとデータベースバックエンドのサーバー構成 49
図 4-3　複数のフロントエンドサーバーとバックエンドサーバーの構成 51

はじめに

『Sun Java System Calendar Server 6 2004Q2 配備計画ガイド』には、Sun Java™ System Calendar Server 6 2004Q2 の配備に必要な情報が記載されています。このマニュアルでは、Calendar Server を理解しながら、ご使用のサイトの評価と分析、および組織のニーズに対応した配備アーキテクチャの設計について説明しています。

- 対象読者
- お読みになる前に
- お読みになる前に
- Web 上の参考資料
- お問い合わせ先
- ご意見、ご要望の送付左記

対象読者

このマニュアルは、サイトで Calendar Server の計画や配備を担当するサイトプランナー、システム管理者、およびサポートスペシャリストを対象としています。

お読みになる前に

このマニュアルは、サイトで Calendar Server の評価および配備を担当する次のような方を対象にしています。

- 評価者
- 設計者
- システム管理者

このマニュアルをお読みなる前に、次の概念について理解しておく必要があります。

- エンタープライズレベルのソフトウェア製品のインストール方法
- DWP、WCAP、および LDAP プロトコル
- Solaris システムの管理およびネットワーク機能

表記規則

次の表は、このマニュアルで使用される文字表記の規則を示しています。

表 1 文字表記の規則

表記	意味	例
AaBbCc123 (モノスペース)	API および言語の要素、HTML タグ、Web サイトの URL、コマンド名、ファイル名、ディレクトリパス名、画面出力の表示、サンプルコード	.login ファイルを編集します。 ls -a を使用してすべてのファイルを表示します。 % You have mail.
AaBbCc123 (太字のモノスペース)	画面出力の表示に対し、ユーザーが入力する文字	% **su** Password:
AaBbCc123 (イタリック)	参照する書名 新しい単語または用語 強調する単語 実際の名前または値で置き換えられるコマンド行の変数	これらは *class* オプションと呼ばれています。 そのファイルは、*cs_svr_base*/cal/sbin ディレクトリにあります。

次の表は、このマニュアルで使用される可変部分の規則を示しています。

表 2 可変部分の表記規則

項目	意味	例
product_base	製品がインストールされるディレクトリを表す	*cs_svr_base*/bin ディレクトリは /opt/SUNWics5/cal/sbin である可能性があります。

次の表は、このマニュアルで使用される記号の表記規則を示しています。

表 3 記号の表記規則

記号	意味	表記法	例
[ ]	省略可能なコマンドオプションを含む	O[*n*]	O4, O
{ }	必要なコマンドオプションの選択肢を含む	d{y|n}	dy
|	コマンドオプションの選択肢を区切る		
+	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で、同時に押すキーを連結する		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で、連続して押すキーを連結する		Esc-S
>	グラフィカルユーザーインタフェースで選択するメニュー項目を示す		「ファイル」>「新規」 「ファイル」>「新規」>「テンプレート」

Web 上の参考資料

このマニュアルのほかにも、Sun Java™ System Calendar Server 6 のマニュアルを参照することができます。次の URL を使用してこれらのマニュアルを参照してください。

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

このマニュアルには、補足的な関連情報を提供するために、サードパーティの URL も記載されています。

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 Web サイトの利用可能性について責任を負いません。Sun は、このようなサイトまたはリソースで得られるあらゆる内容、広告、製品、およびその他素材を保証するものではなく、責任または義務を負いません。Sun は、このようなサイトまたはリソースで得られるあらゆるコンテンツ、製品、またはサービスによって生じる、または生じたと主張される、または使用に関連して生じる、または信頼することによって生じる、いかなる損害または損失についても責任または義務を負いません。

お問い合わせ先

Calendar Server で問題が発生した場合は、次のいずれかの方法でご購入先のカスタマーサポートに連絡してください。

- 次のアドレスにある、ご購入先のソフトウェアサポートサービス

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

  このサイトには、ナレッジベース、オンラインサポートセンター、および ProductTracker へのリンクと、メンテナンスプログラムおよびサポート連絡番号へのリンクが掲載されています。

- メンテナンス契約に関連付けられている電話番号

最善の問題解決のため、サポートに連絡する際は次の情報を提供してください。

- 問題の説明。問題が発生した状況や、その問題が操作に及ぼす影響など
- マシンのタイプ、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。問題に影響を及ぼしている可能性のあるパッチその他のソフトウェアなど
- 問題を再現するための詳細な手順の説明
- エラーログまたはコアダンプ

ご意見、ご要望の送付左記

Sun では、常にマニュアルの向上を心がけ、ユーザーの皆様のご意見、ご提案をお待ちしております。次の Web ベースの書式を利用して、Sun までフィードバックをお寄せください。

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

マニュアルの正式名称と Part No. を該当するフィールドにご記入ください。Part No. はマニュアルのタイトルのページまたは最上部に記載されている 7 桁または 9 桁の番号です。たとえば、この『Calendar Server 配備計画ガイド』の Part No. は 817-7090 です。

第 1 章

Calendar Server について

この章では、Sun Java™ System Calendar Server 6 2004Q2 の概要、Calendar Server の配備が役立つビジネス上の理由、および配備プロセスについて説明します。

この章で説明する内容は次のとおりです。

- Calendar Server の概要
- Calendar Server によるビジネスニーズへの対応
- 配備プロセスについて

Calendar Server の概要

Sun Java System Calendar Server 6 2004Q2 (旧称 Sun™ ONE Calendar Server) は、高性能な、インターネット標準ベースのカレンダサーバーで、中規模および大規模な企業から、さらに非常に大規模なインターネット、遠隔通信、およびエンタープライズのサービスプロバイダまで、各ユーザーの必要に対応したスケーラビリティを考慮し設計されています。ネイティブな Web ブラウザインタフェースまたはコネクタを使用して、Microsoft Outlook などの他のカレンダクライアントに接続することで Calendar Server は、家庭または職場のコンシューマにグループスケジュール機能および個人用のカレンダ機能を提供すると同時に、インターネットを介して他のユーザーとのカレンダ情報の共有を可能にします。ユーザーインタフェース (UI) をカスタマイズして、電子商取引用の Web リンク、バナー広告、ロゴ、またはカレンダサーバーユーザーのブランドなどを含めることができます。

Calendar Server は、オープンで相互運用可能かつ高性能な、業界最高レベルの時間管理およびリソース管理ソリューションです。そのスケーラビリティ、パフォーマンス、信頼性によって、ほかのソリューションに比べて低い総所有コストで、必要な機能を得ることができます。iCalendar 標準のネイティブサポートによって、ユーザーは簡単にインターネットで共有できる形式で予定をスケジュールすることができます。Java System Calendar Server は次のような標準規格とプロトコルを採用しています。

- Internet Calendaring (iCalendar)
- iCalendar Transport-Independent Interoperability Protocol (iTIP)
- iCalendar Message-based Interoperability Protocol (iMIP)
- Extensible Markup Language (XML)
- Lightweight Directory Access Protocol (LDAP)
- HyperText Transport Protocol (HTTP)

Calendar Server のアーキテクチャは、垂直方向 (システムごとの CPU の数を増大させる) と水平方向 (ネットワークにサーバーを追加する) の両方向で、柔軟性に富み、拡張可能で、スケーラブルなものです。その結果、Calendar Server は、さまざまなニーズに対応した設定が可能なサーバーから構成されるシステムと見なすことができます。スタンドアロンのカレンダサーバーとして単独で使用することもでき、さまざまなサービスをサーバー間で重複または分割させる、多くのインスタンスで設定することもできます。

Calendar Server は、プラグインを利用して外部のサービスを取得します。さらに、LDAP ベースおよびアイデンティティベースの配備もサポートしており、Sun Java™ System Identity Server (旧称 Sun™ ONE Identity Server)、Sun Java™ System Portal Server (旧称 Sun™ ONE Portal Server)、および Sun Java™ System Instant Messaging (旧称 Sun™ ONE Instant Messaging) と統合して追加の機能を提供します。

Calendar Server は次の利点を提供します。

- Web ベースのグループスケジュール機能、空き時間 - 予定ありの検索、および企業のディレクトリの検索
- 会議室、プロジェクタ、および他のリソースの Web ベースのリソーススケジュール機能
- XML ベースのカスタマイズ機能 (配色、ログイン、ユーザーインタフェース、ロゴ、ブランド設定など)
- XML または iCalendar 形式で配信される標準ベースの予定およびカレンダデータフィードのサポート。これによって通信機能が強化され、商取引リンクやバナー広告から新しい収入の可能性が提供される
- 他のディレクトリサービス用の API を含む、ネイティブ LDAP のサポート
- Microsoft Outlook や Evolution を含む追加のカレンダクライアントへのコネクタ。これによって、クライアントは Calendar Server 上でスケジュール機能を実行することができる
- ホストしているドメインのサポート
- システム管理、オンラインのバックアップと復元、およびデータベース全体のバックアップと復元の簡略化

- 仕事、家族、友人などの複数のカレンダのサポート
- 公開および非公開のカレンダのサポートのほか、公開、非公開、および機密の個々の予定のサポート
- カレンダの階層化表示のサポート。これによってユーザーは 2 つ以上のカレンダを 1 つの表示に統合でき、コミュニケーションや生産性を向上することができる
- 選択した受信者にアポイント、招待、およびアラームの電子メール通知が自動的に送信され、Java System Instant Messaging との統合でポップアップアラームが自動的に提供される
- 各カレンダで複数の所有者がサポートされ、プロジェクトチームやコミュニティグループでコミュニケーションが簡単になり生産性が向上する
- 一次所有者の代わりに行使する他者にカレンダの所有権を委譲する機能
- 日ごと、週ごと、月ごと、年ごと、および比較の各ビュー
- スケーラブルで、ネットワーク化された、サーバー間、クライアントサーバーアーキテクチャによって、何十万ものユーザーをサポート
- Secure Sockets Layer (SSL) 暗号化、LDAP 認証、認証プラグイン、および Java System Identity Server でアイデンティティ対応のシングルサインオン (SSO) をサポート

Calendar Server の概念の詳細については、次の Web サイトから入手できる『Sun Java System Calendar Server 管理ガイド』を参照してください。

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

Calendar Server によるビジネスニーズへの対応

Calendar Server は、オープンで相互運用可能かつ高性能な、業界最高レベルの時間管理およびリソース管理ソリューションです。Calendar Server によって、ほかのソリューションに比べて低い総所有コストで、必要な機能を得ることができます。Calendar Server のアーキテクチャは、柔軟で拡張可能なので、垂直方向 (システムごとの CPU の数を増大させる) と水平方向 (ネットワークにサーバーを追加する) の両方向で拡張性があります。

次の表では、企業が Calendar Server から得られる利点を要約しています。

表 1-1 企業にとっての Calendar Server の利点

主な機能	企業への利点
高いパフォーマンスおよびスケーラビリティ	効率的な通信が可能になり企業と ISP の両方のサービス品質が向上
豊富なセキュリティ機能	通信とデータの整合性および従業員、顧客、提携業者のプライバシーの保護と、業界の規則の遵守
スケーラブル、堅牢、さらに拡張可能なコンポーネント	一体化した通信サービスの配備が可能になり、電話サービスとともに、電子メール通知、ファックス、ポケットベルなどの技術も提供可能
予定のスケジュール、作業およびリソースの管理のための拡張可能なコラボレーションプラットフォーム	Calendar Server で時間とリソースの管理が改善されユーザーの生産性が向上
会議や予定のグループスケジュール機能	Calendar Server で企業全体のチームコラボレーションや通信が改善
ハイパーリンクで予定または作業の情報を共有	Calendar Server では作業または予定に関連した情報の交換によってコラボレーションを促進
オープン、モジュール方式、および標準ベースのアーキテクチャ	顧客がカスタマイズした、パーソナライズしたソリューションを配備可能

Calendar Server 配備の可用性を高める方法

Calendar Server には、Sun™ Cluster サービスをサポートする高可用性オプションがあります。このオプションを使うと、一次システムが保守のためにオフラインになったり、障害のために停止する場合、二次 Calendar Server ホストがユーザーに対してサービスを提供します。

さらに、冗長コンポーネントを使用して、Calendar Server を高可用性設定で配備できます。この種の配備により、サービスの稼働時間を高レベルにすることができます。このように可用性の高い配備を行うには、サービスアーキテクチャの各コンポーネントで冗長性が必要になります。このようなコンポーネントには、二重のデータストアサーバー、二重のネットワークインタフェースカード、および二重のシステム記憶装置が含まれます。

注 このマニュアルでは、Calendar Server 用の高可用性配備で Sun Cluster を使用するための詳細を説明していません。この点については、Sun Cluster および Calendar Server のマニュアルを参照してください。

Calendar Server とともに Portal Server を使用する方法

Calendar Server を Portal Server とともにインストールすると、ポータルページでカレンダポートレットにアクセスできます。このポートレットでは、カレンダスケジュールおよびアドレス帳情報が提供されます。Portal Server との統合によって Portal Server と Calendar Express Web クライアント間のシングルサインオン機能に加え、他の Messenger Express と Communications Express クライアント間のシングルサインオン機能も可能になります。

Portal Server の次の 2 つのコンポーネントによって、基本的な Calendar Server の配備に機能を追加することができます。

- Portal Server Desktop: Portal Server にインストールされた Calendar Server によって、ユーザーは Calendar Express を起動できます。
- Sun Java™ System Portal Server 6、Secure Remote Access: これにより、リモートエンドユーザーは、インターネットを介して Secure Remote Access 組織のネットワークやそのサービスに安全に接続できます。エンドユーザーは、Secure Remote Access ゲートウェイを介して、Web ベースの Portal Server Desktop にログインすることで Secure Remote Access にアクセスします。Portal Server に設定された認証モジュールで、エンドユーザーが認証されます。エンドユーザーのセッションが Portal Server との間で確立されると、エンドユーザーの Portal Server Desktop へのアクセスが有効になります。

注	このマニュアルでは、ポータル環境でポータル配備する Calendar Server について説明していません。詳細については Portal Server のマニュアルを参照してください。

配備プロセスについて

Calendar Server の配備プロセスは、通常、次のフェーズから構成されます。

- 配備設計
- 開発
- プロトタイプテスト
- 製品のロールアウト

配備フェーズは固定的なものではなく、配備プロセスは反復して行われます。ただし次の各節では、配備フェーズをそれぞれ個別に説明しています。

配備とアーキテクチャの設計

通常、配備設計フェーズでは、要件分析フェーズで指定された配備シナリオに基づいて配備アーキテクチャを構築します。その目的は、配備シナリオで指定されたシステム要件を満たすように、論理構築ブロック (論理アーキテクチャ) を物理環境 (物理トポロジ) に配置することです。

この設計の 1 つの側面は、物理環境の大きさを調整して、ロード、可用性、およびパフォーマンスの各要件を満たすようにすることです。配備アーキテクチャでは、システムサーバーやアプリケーションコンポーネントを環境内の処理ノードに割り当てる際の、異なる処理ノードの能力やネットワークの帯域幅などの物理トポロジの詳細を考慮に入れます。

配備の目的

配備計画を開始する前に、次のことを確認してください。

組織が Calendar Server を配備するのはなぜか

次のようにいくつかの理由が考えられます。

- **コストの削減** : ユーザーごとの総所有コストが市販されている他のカレンダ製品を使用するより低くなります。

- **生産性の向上** : カレンダユーザーは、自分の予定や作業を管理できるほか、組織の他の職員との会議やアポイントの予定を設定することができます。また、ユーザーはカレンダグループと、会議室や機器などのリソースを管理できます。さらに、カレンダを PDA などのモバイルデバイスや Microsoft Outlook と同期させることもできます。
- **スケーラビリティおよび可用性の向上** : Calendar Server は垂直と水平の両方向で拡張性があります。組織が拡大すると、サーバーをアップグレードしたり、さらにサーバーを追加したりすることによって簡単に設定をアップグレードできます。
- **セキュリティの向上** : Solaris システムに Calendar Server を配備する場合、組織は、Windows 環境で広く見られる多くのウィルスや他のセキュリティ上の危険性を防止できます。
- **高可用性 (HA) 設定** : Sun Cluster などのクラスタソフトウェアとの統合によって、Calendar Server で高可用性サービスが行えるように設定できます。ソフトウェアやハードウェアの障害が発生すると、Calendar Server は二次サーバーへフェイルオーバーします。

Calendar Server 配備チーム

通常、Calendar Server の配備には、それぞれが異なる役割と責任を受け持つ何人かの人員が必要です。小規模な組織では、1 人でいくつかの役割を兼任することがあります。考慮すべき役割の中には次のものがあります。

- プログラムマネージャは、Calendar Server 配備全体を監督し、その成功や失敗に責任を持ちます。
- Calendar Server 管理者は、Calendar Server を管理するための毎日の管理業務を行い、さらに Calendar Server のインストールやアップグレードの責任者となる場合もあります。
- パフォーマンスエンジニアは、試験的配備および本稼動における配備の Calendar Server のパフォーマンスをテストおよび監視し、配備条件を満たしているかを確認します。
- 開発エンジニアリングでは、Calendar Server のアプリケーションやプラグインを記述し、必要に応じて、Calendar Server のユーザーインタフェース (UI) をカスタマイズします。
- マニュアルスペシャリストは、管理者やエンドユーザー向けにカスタマイズされたあらゆるマニュアルを執筆します。
- 教育およびトレーニングでは、実務講習と教材を開発します。

サポートスペシャリストは、試験的配備および本稼動における配備の両方をサポートします。

Calendar Server のエンドユーザー

エンドユーザーは、Calendar Express Web クライアント、Communications Express Web クライアント、または Sun Java™ System Connector for Microsoft Outlook を使用することによって Calendar Server に接続できます。

サイトのエンドユーザーについて、次のことを確認します。

- サイトには、全部で何人の Calendar Server のエンドユーザーがいるか
- エンドユーザーはどのように Calendar Server、Calendar Express、Microsoft Outlook、またはクライアントの組み合わせに接続しているか
- 地理的な場所はいくつ含まれているか。エンドユーザーはすべて同じタイムゾーンにいるかまたは異なるタイムゾーンにいるか
- エンドユーザーは、毎日、同じ時間に Calendar Server にログインするか
- 配備では、ピーク使用時のアクティブなエンドユーザーは何人いるか
- エンドユーザーベースはどのくらいの速さで拡張するか
- Calendar Server エンドユーザーに特有のパフォーマンス要件は何か
- シングルサインオン (SSO) 要件は何か
- ユーザーの中に、Netscape Calendar 4.x から移行しているユーザーがいるか
- エンドユーザーは Sun™ ONE Synchronization の使用を計画しているか
- エンドユーザー向けに UI のカスタマイズを計画しているか
- サイトにプロキシサーバーの使用を計画しているか
- サイトにロードバランス機能の使用を計画しているか

必要なエンドユーザーのパフォーマンス

エンドユーザーに特有のパフォーマンス要件は何でしょうか。たとえば次のようなものがあります。

- どのくらいのエンドユーザーの応答時間が許容されるか
- ピークロード時に予想されるパフォーマンスの低下を許容できるか

配備で使用することを計画しているのはどのような設定でしょうか。Calendar Server の設定シナリオには、次のものが含まれます。

- 1 つの Calendar Server インスタンス
- 1 つのバックエンドデータベースサーバーを備える 1 つのフロントエンド
- LDAP CLD プラグインを使用し、複数のバックエンドデータベースサーバーを備える複数のフロントエンドサーバー

- LDAP CLD プラグインを使用する複数のフロントエンドまたはバックエンドサーバー、あるいはその両方
- 高可用性 (HA) 設定

複数のフロントエンドサーバーの設定を計画している場合、どのようにエンドユーザーの分散を計画するでしょうか。

複数のバックエンドデータベースサーバーの設定を計画している場合、どのようにデータベースの分散を計画するでしょうか。たとえば、地理的に分散させることができます。

どのような拡張計画があるでしょうか。フロントエンドとバックエンドの両方についてはどうでしょうか。

開発およびカスタマイズ

ライフサイクルの要件分析段階で指定された論理アーキテクチャによって、ソリューションの実装に必要な開発作業の範囲が決定します。

API を使用してサービスを拡張する際や、たとえば企業のブランド設定を導入してルックアンドフィールをカスタマイズする際には、追加の作業が必要なこともあります。

ソリューションによっては、配備やカスタマイズにかなりのコストがかかり、新しいビジネスおよびプレゼンテーションサービスの開発が必要になる場合もあります。他のソリューションの場合、Portal Server デスクトップなどの既存のグラフィカルユーザーインタフェースをカスタマイズすることによって必要な機能の実現が可能な場合もあります。

製品 API の使用や製品機能のカスタマイズについては、適切なコンポーネント製品のマニュアルを参照してください。これには、次のマニュアルが含まれます。

- 『Sun Java System Communications Services Event Notification Service Guide』
- 『Sun Java System Calendar Server Developer's Guide』

プロトタイプとテスト

プロトタイプフェーズでは、テスト環境で配備アーキテクチャを実装することによって配備設計の試作品を作成します。前述のように (「開発およびカスタマイズ」を参照) 開発結果の新しいアプリケーションロジックおよびサーバーのカスタマイズを使用して、概念実証開発テストを行います。このフェーズには、分散アプリケーションのインストール、設定、および起動のほか、テスト環境に必要なあらゆるインフラストラクチャサービスが含まれます。

プロトタイプテストによって配備アーキテクチャの欠点が明らかになった場合、アーキテクチャを修正し、再度試作品を作り、テストを再び行います。この反復プロセスの結果、本稼動環境で配備可能な配備アーキテクチャが完成するはずです。

試験的配備には、配備の失敗や深刻な問題に遭遇した場合の、ロールバック計画も含める必要があります。この計画の一部として次のことを考慮します。

- カレンダデータのバックアップ手順とスケジュール。たとえば、カレンダデータを毎晩バックアップし、1 週間データを保存したい場合などです。詳細については、次の Web サイトから入手できる『Sun Java System Calendar Server 管理ガイド』を参照してください。

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

- ロールバック計画開始の条件。問題がどの程度深刻になったら、ロールバック計画を開始するのでしょうか。たとえば、データベースが破損して、復元が不可能になる場合などです。

本稼動システムのロールアウト

本稼動ロールアウトフェーズでは、配備アーキテクチャを本稼動環境で実装します。このフェーズには、分散アプリケーションのインストール、設定、および起動のほか、本稼動環境で必要なあらゆるインフラストラクチャサービスが含まれます。通常、一部の配備から開始し、組織全体への実装に移行します。このプロセスでは試験的実行を行い、ロードとストレスを増大させるテストをシステムに課します。

ロールアウトフェーズの一部として、ユーザーのプロビジョニング、シングルサインオンの実装、パフォーマンス目標を満たすためのシステム調整などの管理タスクを行うことが必要な場合もあります。配備の検証および容量計画の実行もこのフェーズの一部です。容量計画は、その中でもシステムの監視が重要な役割を果たしますが、システム拡張の長期的なニーズに対応するために必要なものです。

第 2 章

要件の分析

Calendar Server 配備を計画するには、最初に組織のビジネス要件と技術要件を分析する必要があります。この章では、Calendar Server アーキテクチャを決定するための、要件の収集および評価について説明します。

この章で説明する内容は次のとおりです。

- 配備目標の特定
- プロジェクト目標の決定
- 拡張計画

配備目標の特定

Calendar Server ソフトウェアまたはハードウェアを購入または配備する前に、配備目標を特定しておく必要があります。配備要件は組織内のさまざまな部門から出される場合があります。要件は漠然とした言葉で表現されることが多いため、特定の目標を決定するために、明確なものにする必要があります。

要件分析の結果は、配備の成功を判断できるような、明快かつ簡潔で、計測可能な目標にならなければなりません。プロジェクトのすべての利害関係者が了承する明確な目標を設定しないでプロジェクトを進めることは、避けるべきです。

要件の中には、次のように配備を計画する前に調査が必要なものもあります。

- ビジネス要件
- 技術要件
- 財務要件
- サービスレベル契約 (SLA)

ビジネス要件

ビジネスの目標は配備の各決定に影響します。具体的には、ユーザーの使用パターン、サイトの分散、および場合によっては配備に影響する、起こりうる政策上の問題を理解する必要があります。このようなビジネス要件を理解していない場合、容易に誤った前提に陥る可能性があり、それは配備設計の正確さに影響します。

運用要件

運用要件を単純明快な目標のある一連の機能要件として表現してください。通常、次についての非公式な仕様を目にすることがあります。

- エンドユーザーの機能
- エンドユーザーの応答時間
- 可用性と稼働時間
- 情報のアーカイブと保存

たとえば、「適切なエンドユーザーの応答時間」の要件を、すべての利害関係者が、何が「適切」で、どのように応答時間が計測されるかを理解できるような計測可能な用語に変換します。

企業文化と社内力学

配備には企業文化と社内力学を考慮に入れる必要があります。ある領域から要求が出され、結果的にビジネス要件になる場合があります。たとえば、次のようなものがあります。

- サイトによっては配備されたソリューションの独自の管理が必要です。このような要求により、プロジェクトのトレーニングコストや複雑さなどが高まる場合があります。
- LDAP ディレクトリに従業員のデータが収められている場合、人事部門がそのディレクトリの所有および管理を望む場合があります。

技術要件

技術要件 (機能要件) は組織のシステムニーズの詳細です。

既存の使用パターンのサポート

既存の使用パターンを、配備を実現させるための明確で計測可能な目標で表現してください。次の質問はそのような目標を決定するのに役立ちます。

- 現在のサービスはどのように利用されているか
- ユーザーを、たとえば、ときどき使用するユーザー、頻繁に使用するユーザー、または非常に多く使用するユーザーに分類できるか
- 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
- 通常、カレンダのアポイントでどのくらいの招待があるか
- ユーザーはメッセージをいくつ送信するか
- 通常、ユーザーが日ごとまたは時間ごとに作成するカレンダ予定および作業はいくつか
- ユーザーは会社のどのサイトへメッセージを送信するか

サービスにアクセスするユーザーを調査してください。ユーザーが既存のサービスをいつ使用するかなどの要因が配備要件、つまり目標を特定するための鍵です。自分の組織の経験からはこのようなパターンを見出すことができない場合、他の組織の経験を調査し、自分の組織について見積もってみてください。

組織の使用率が非常に高い地域では専用のサーバーが必要な場合もあります。一般的に、ユーザーが実際のサーバーから離れている場合、応答時間が遅くなります。応答時間が許容可能かどうかを考慮してください。

サイトの分散

次の質問を使用し、サイトの分散がどのように配備目標に影響するかを認識してください。

- サイトは地理的にどのように分散しているか
- サイト間の帯域幅はどのくらいか。集中方式では、非集中方式よりも広い帯域幅が必要とされる。ミッションクリティカルなサイトでは専用のサーバーが必要な場合もある

ネットワーク

次の質問はネットワーク要件を理解するのに役立ちます。

- 内部のネットワーク情報の暗号化が必要か
- ネットワークサービスの冗長性が必要か

- アクセス層のホストにある使用可能なデータの制限が必要か
- エンドユーザーの設定の簡略化が必要か、たとえば、エンドユーザーに変更する必要のない 1 つのメールホストを入力してもらうようにするか
- ネットワーク HTTP トラフィックを減らす必要があるか

注 これらの質問に肯定の答えをする場合、二層のアーキテクチャをお勧めします。

既存のインフラストラクチャ

より信頼できる、高帯域幅を持つ場合、サーバーの集中化が可能なことがあります。

- 既存のインフラストラクチャおよび設備はこの配備を有効にするのに適切であることを実証しているか
- DNS サーバーは余分なロードに対応可能か。ディレクトリサーバーはどうか。ネットワークはどうか。ルーターはどうか。スイッチはどうか。ファイアウォールはどうか

従業員のサポート

24 時間、1 週間毎日 (24 × 7) のサポートが利用可能なのは特定のサイトだけに限られる場合があります。少数のサーバーを使った簡単なアーキテクチャの場合、サポートも容易になります。

- 運用およびテクニカルサポートグループには、この配備をスムーズに行う十分な容量があるか
- 運用およびテクニカルサポートグループは、配備フェーズ中のロードの増大に対応できるか

財務要件

財務上の制約は配備を構築する方法に影響します。財務要件は、配備の制約または目標が定められる全体的な考え方で明確に定義される傾向にあります。

ハードウェア、ソフトウェア、保守などの明確なコスト以外に、次のような他のいくつかのコストがプロジェクト全体のコストに影響する可能性があります。

- トレーニング
- ネットワークの帯域幅やルーターなど、他のサービスや設備のアップグレード
- 配備概念を実証するために必要な従業員やリソースなどの配備コスト
- 配備したソリューションを管理するための従業員などの運用コスト

プロジェクト要件に関連付けられた多くの要因に対して、慎重に、十分な分析を行うことによって、そのプロジェクトに関連付けられた財務上の問題を回避できます。

サービスレベル契約 (SLA)

稼働時間、応答時間、メッセージ配信時間、および障害回復に関する配備について SLA を作成する必要があります。SLA 自体は、システムの概要、サポート組織の役割と責任、応答時間、サービスレベルの計測方法、変更要求などの項目を詳細に記述したものである必要があります。

システムの可用性に対する組織の期待を特定することが、SLA の範囲を決定する鍵です。システムの可用性は、通常、システム稼動時間のパーセントで表現されます。システムの可用性を算定する基本的な計算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、9999 (99.99 パーセント) の稼働時間ということは、1 か月のうちにシステムの使用不可能が許容されるのが 4 分程度であることを意味しています。

さらに、システムの停止時間とはシステムが使用不可能な合計時間です。この合計には、ハードウェアの故障やネットワークの障害などの突発的な停止時間だけではなく、予防保守、ソフトウェアのアップグレード、パッチなどの計画的な停止時間も含まれます。システムを 7x24 (1 週間毎日、1 日 24 時間) 使用可能なものとする場合、計画的および突発的な停止時間を回避し、高可用性を確保するために、アーキテクチャに冗長性を含める必要があります。

プロジェクト目標の決定

調査と分析を行ってプロジェクト要件を明らかにする必要があります。次に、明確な一連の目標を決定できる必要があります。目標およびその目標に照らしたプロジェクトの評価方法を、プロジェクトに直接関与しない従業員が理解できるような形で、これらの目標を指定してください。

プロジェクト目標は、すべての利害関係者によって承認される必要があります。プロジェクト目標は、プロジェクトの成功を見きわめるために、実装後の検査で計測される必要があります。

拡張計画

現在必要な容量を決定することに加えて、将来、計画可能な期間内にどの程度の容量が必要かを査定してください。通常、拡張のスケジュールは 6 か月から 12 か月の範囲のものです。拡張の見込みと使用特性の変更は、拡張に対応するため考慮に入れる必要があります。

ユーザーとメッセージの数が増大するにつれ、容量計画の正しいガイドラインの概要をまとめる必要があります。さまざまなサーバーのメッセージトラフィックの増大、ユーザー数の増大、メールボックスサイズの拡張などについて計画する必要があります。ユーザー数の増大とともに、使用特性は変わります。配備目標 (配備設計) は、将来の発展性に対応している必要があります。

理想的には、将来の拡張に容易に対応するアーキテクチャを設計する必要があります。本稼動フェーズでは、配備の拡張がいつどの程度必要になるかを理解するために、配備の監視も重要になります。

総所有コストの理解

容量計画に影響するもう 1 つの要因としては、総所有コスト (TCO) が挙げられます。これには Calendar Server を配備するハードウェアの選択が含まれます。33 ページの表 2-1 では、小規模のハードウェアシステムを配備するか、または少数の規模の大きいハードウェアシステムを配備するかのいずれかについて考慮すべきいくつかの要因を示しています。

表 2-1　総所有コストの考慮事項

ハードウェアの選択	利点	欠点
多数の小規模ハードウェアシステム	• 通常、小規模ハードウェアシステムはコストも少ない • 多数の小規模ハードウェアシステムは、多くの場所の各箇所に配備して分散ビジネス環境をサポートできる • 多数の小規模ハードウェアシステムの場合、保守中のサーバーがあっても、トラフィックを引き続きオンラインの他のサーバーへルーティングすることができるので、システムの保守、アップグレード、および移行のための停止時間を少なくすることができる	• 小規模ハードウェアシステムの方が容量に限界があるので、より多くのシステムが必要になる。維持、管理、および保守のコストはハードウェアシステムの数が増えるにつれて増大する • 多数の小規模ハードウェアシステムの方が保守するシステムが多くあるので、システム保守が多く必要になる
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、サーバーごとの固定管理コストが少ない。管理コストが、内部または ISP からに関係なく、毎月定期的に請求される場合、管理するハードウェアシステムが少数なのでコストが低くなる • さらに少数のハードウェアシステムの方が、保守するシステムが少数なので、システムの保守、アップグレード、および移行を簡単に行うことができる	• 通常、大規模ハードウェアシステムの方が、当初はコストがかかる • さらに少数のハードウェアシステムの方が、保守、アップグレード、および移行のためのシステム停止時間が長い可能性がある • 専門の熟練したシステム管理者が必要

第 3 章

ネットワークインフラストラクチャの必要性の決定

ネットワークインフラストラクチャはシステムの根本的な基盤です。ネットワークインフラストラクチャが形成する各サービスによってネットワークの動作構造が作り出されます。Calendar Server の配備では、プロジェクト目標からネットワークインフラストラクチャを決定すると、確実に拡張および成長が可能なアーキテクチャを備えることができます。

この章で説明する内容は次のとおりです。

- 既存のネットワークについて
- ネットワークインフラストラクチャコンポーネントについて
- ネットワークインフラストラクチャレイアウトの計画

既存のネットワークについて

既存のネットワークインフラストラクチャを理解して、それが配備目標のニーズに十分対応可能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して、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、新しいネットワークコンポーネントを購入したりする必要があるかどうかを確認します。次の分野をカバーすることによって既存のネットワークの完全なマップを作成する必要があります。

1. ケーブルの長さ、種類などの物理通信リンク
2. アナログ、ISDN、VPN、T3 などの通信リンク、およびサイト間で使用可能な帯域幅と応答時間
3. 次のようなサーバーについての情報
    - ホスト名
    - IP アドレス

- ドメインメンバーシップの Domain Name System (DNS) サーバー

4. ネットワークの次のデバイスの場所
   - 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、各サイトでのユーザーの数

この一覧表を完成させた後、プロジェクト目標に関連した情報を検討し、配備を正常に実現するために必要とされる変更を判断する必要があります。

ネットワークインフラストラクチャコンポーネントについて

次の一般的な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功に直接影響します。

- 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同士を接続し、システム間の通信ができるようにします。プロジェクトで計画された拡張や用途に対応するため、配備後のルーターに予備の容量を確保する必要があります。

同様に、スイッチはネットワーク内のシステム同士を接続します。

ルーターやスイッチがフル稼働しているとボトルネックの増大をまねく傾向があり、その結果、クライアントが異なるネットワークのサーバーにメッセージを送信するのにかなり多くの時間を費やすことになります。このような場合、ルーターまたはスイッチをアップグレードする見通しや費用がないと、そのコストに比較にならないほど従業員の生産性に影響する可能性があります。

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間に配置されアクセス制御を行います。元々ファイアウォールは、信頼できないネットワーク (インターネット) から信頼ネットワーク (使用しているもの) を保護するために使用されていました。最近では、信頼できないネットワーク (ネットワークおよびインターネット) から自分の (信頼される、分離した) ネットワークのアプリケーションサーバーを保護するために広く使われるようになっています。

ルーターを設定すると、ファイアウォールに入ってくるデータを選別することで全体的なファイアウォールの能力が増強されます。ルーターを設定すると、場合によっては好ましくない、NFS、NIS などの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できないホストやネットワークからのトラフィックをブロックすることができます。

さらに、インターネットや信頼できないネットワークに公開している環境で Sun サーバーをインストールする場合、Solaris のインストールは、ホストするアプリケーションをサポートするのに必要な最小限のパッケージ数まで減らしてください。サービス、ライブラリ、およびアプリケーションを最小限にすることができれば、保守を必要とするサブシステムの数を少なくできるので、セキュリティの強化に役立ちます。Solaris™ Security Toolkit は、Solaris オペレーティング環境のシステムの最小化、堅固化、および安全性の向上を図るための柔軟で拡張可能な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は、このような課題に対して指針を与える必要があります。

ロードバランサ

ロードバランサを使用して、ロード全体を Web サーバーやアプリケーションサーバーに分散させたり、実行するタスクの種類に応じて要求を分散させてください。たとえば、さまざまな専用アプリケーションがあり、そのために異なるアプリケーションサーバーを持っている場合、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することができます。

複数のデータセンターが存在する場合は、地理的なロードバランスを考慮する必要があります。地理的なロードバランスでは、要求、サイトの容量、およびユーザーにもっとも近いの場所などに応じて、ロードを分散させます。1 つのセンターが停止した場合、地理的なロードバランサによってフェイルオーバー機能が働きます。

Web ファーム上のロードバランサの場合、ハードウェアロードバランサをサーバーの前およびルーターの後ろに配置し、ルーティングされたトラフィックが適切なサーバーに誘導されるようにします。ソフトウェアのロードバランスソリューションは Web サーバー自体に常駐します。ソフトウェアソリューションを使用すると、通常、サーバーの 1 つがトラフィックスケジューラとして機能します。

ロードバランスソリューションは、入力パケットのヘッダーと内容を読み取ることができます。これにより、ユーザーや要求のタイプなど、パケット内の情報の種類に応じてロードバランスを行うことができます。パケットヘッダーを読み取るロードバランスソリューションによって、権限のあるユーザーを識別し、所定のタスクを処理するサーバーに要求を送信することができます。

要求に応じるすべてのサーバーとロードバランサが動的にどのように通信するかを調査する必要があります。スケジューラは、ロードデータを確認するために、それぞれのサーバーを ping しますか。あるいは、サーバーに常駐する「ライブ」エージェントを作成しますか。また、ロードバランサがどのように TCP パケットをパースするのかも調べる必要があります。どのくらい短時間でロードバランサがパケットの処理をできるかに注目してください。ロードバランサによっては他より効率的なものとなります。ロードバランスの効率は、通常、スループットで計測されます。

ストレージエリアネットワーク (SAN)

ストレージシステムのデータ要件を理解することは、配備を成功させるために必要です。ストレージに関連して使用されるサーバーにストレージが依存しないように、ますます SAN が配備されています。SAN の配備によって、ストレージドライブの場所を変更することなくマシンを置き換えることができるので、機能しないサーバーからの復元時間を短くすることができます。

次の質問を使用して、配備ストレージ要件が SAN を通じて最適に処理されているかを評価してください。

- 読み取りまたは書き込みは広く行われるようになっているか
- 高速 I/O ストレージが必要か。ストライピングが最善のオプションか
- 高レベルの稼働時間が必要か。ミラーリングが最善のオプションか
- どのようにデータはバックアップされるか。いつバックアップされる予定か。

DNS

DNS 照会の使用率が非常に高いサーバーでは、ローカル DNS キャッシュサーバーを装備して、検索応答時間およびネットワークトラフィックを低減させる必要があります。

要件を決定する場合、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のホスト名の割り当てを考慮してください。すべてのホスト名が現在 1 つ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。このような方法で設定されるサービスを使用すると、代替ハードウェアへのサービスの再配置による変更の影響がかなり少なくなります。

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トポロジを導き出す際、次の点について考慮しておく必要があります。

- 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)

今日では、ほとんどの企業のネットワークに DMZ が設定されています。DMZ はインターネットから企業のネットワークを分離します。DMZ は安全性が強化された領域であり、インターネットのサービスや設備を提供する、Web サーバーなどのサーバーをそこに配置します。これらのマシンは受ける可能性のある攻撃に耐えるよう強化されます。そのような攻撃によるセキュリティの侵害の場合にも被害を抑えるため、通常、これらのサーバーには内部ネットワークについての情報が含まれていません。たとえば、ネームサーバーの設備には、サーバーとインターネットへのルーターだけが含まれています。

DMZ の実装場所は、ファイアウォールのセキュリティと設備の堅固さが増すにつれ、徐々にファイアウォールの背後のセグメントに移動しています。しかし、DMZ はまだ内部ネットワークから分離された状態にあります。引き続き、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を DMZ セグメントに配置する必要があります。

簡単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用に、別々の DMZ セグメントを設定するだけの場合もあります。しかし、VPN アクセスおよびリモートアクセスのトラフィックにはセキュリティの問題があります。これらのタイプの該当する接続をネットワークの他の部分から分離しておく必要があります。

DMZ の分離を可能にするファイアウォールによって、DMZ 内でサービスを提供している、対応するサービスポートおよびホスト宛てのインバウンドパケットだけが許容されます。さらに、インターネットに向けて開始されるアウトバウンドトラフィックは、たとえば DNS やメールなど、提供するサービスを実行するためにインターネットへのアクセスが必要なマシンに限定してください。接続要求の特定のタイプについて、インバウンドのみの DMZ と アウトバウンドのみの DMZ に分離することを考えることもできます。ただし、DNS や電子メールを中断させるサービス拒否攻撃の可能性がある場合は、インバウンドサーバーとアウトバウンドサーバーを別々に設置してこれらのサービスを提供することを考慮してください。電子メールベースのトロイの木馬またはワームがアウトバウンドメールサーバーを制御不能にしたり、オーバーランする場合でも、インバウンドの電子メールは引き続き受信することができます。同じ手法を DNS サーバーにも適用してください。

イントラネット

DMZ はインターネットにサービスを提供するホストのネットワークセグメントになります。この設計により、内部ホストは外部からの攻撃によって危険にさらされる可能性があるホストと同じセグメントにないため保護されます。内部においても、Web、メール、ファイルサービス、内部 DNS などの同様のサービスが内部ユーザーに対してのみ提供されます。ちょうどインターネットサービスが分離されているのと同様に、内部サービスも分離されます。このようなサービスの分割によって、より緊密な制御がルーターのフィルタリングで行われます。

インターネット向けのサービスをセキュリティのために DMZ に分離するのと同様に、非公開の内部サービスも独自の 内部 DMZ に置く必要があります。

サービスやネットワークのサイズに応じて、複数の DMZ が有用であるように、複数のイントラネットも役立つことがあります。

分離を可能に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するルールと同様に設定する必要があります。インバウンドトラフィックは、内部メールサーバーに渡されるインバウンド電子メールなどの DMZ から情報を中継するマシンおよび内部ネットワークに常駐するマシンからのみ来るはずです。

内部ネットワーク

残りのセグメントから内部ネットワークセグメントが構成されます。これらのセグメントにはユーザーのマシンや部門のワークステーションが収容されています。これらのマシンはイントラネットに常駐するホストから情報を要求します。開発、研究、およびテストの各ネットワークセグメントもこのリストに含まれます。部門間のセキュリティを追加するため、それぞれの内部ネットワークの間に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ってください。これらのセグメントのそれぞれに使用されてい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特定し、内部ファイアウォールが有用であるかどうかを判断してください。

これらのマシンはインターネット上のマシンと直接通信しないようにする必要があります。できれば、これらのマシンと DMZ のマシンとの直接通信を回避します。つまり、必要とされるサービスがイントラネットのホストに常駐している必要があります。次に、イントラネットのホストは DMZ のホストと通信し、アウトバウンド電子メールや DNS などのサービスを完了させることができます。このような間接的な通信が許容されます。

プロキシ

インターネットのマシンと直接通信するマシンだけが DMZ に常駐する必要があります。ただし、ユーザーがインターネットにアクセスを要求した場合、これは前のトポロジの決定からすると問題を引き起こします。このような状況では、プロキシが役立ちます。プロキシを内部ネットワークセグメント、または可能なら、イントラネットセグメントに配置してください。インターネットへのアクセスが必要なマシンは、その要求をプロキシに渡すことができ、次にプロキシはそのマシンの代わりに要求を発行します。インターネットへのこのリレーアウトは、発生する可能性のある危険からマシンを保護するのに役立ちます。

プロキシはインターネットのマシンと直接通信するため、DMZ に常駐する必要があります。ただし、このことは内部のマシンが DMZ のマシンと直接通信することを防止するという要求とは相反します。この通信の間接性を保持するため、二重のプロキシシステムを使用してください。イントラネットに常駐する 2 番目のプロキシは内部マシンの接続要求を DMZ のプロキシに渡し、次に DMZ のプロキシはインターネットとの実際の接続を確立します。

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に加えて、ほとんどのファイアウォールには IP スプーフィングを防止する機能があります。可能なあらゆる箇所で IP スプーフィングに対する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ネットワークへのエントリポイントが 1 つのみで、インターネットから受信したパケットに内部マシンの 1 つのソースアドレスが付いてくる場合、なりすましの可能性があります。ネットワークトポロジからすると、内部マシンの ソース IP アドレスを含むパケットはネットワーク内部からのみ来るはずで、インターネットから来るはずがないからです。IP スプーフィングを防止することによって、この可能性が排除され、IP アドレスベースの認証や他のファイアウォールフィルタリングのルールが無視される可能性が減少します。同じ IP スプーフィング保護機能をあらゆる内部ファイアウォールに同様に使用してください。

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がいる場合、それらのユーザーが設備にアクセスする方法にも注意してください。それらのユーザーがアクセスできない設備があるでしょうか。どのような種類のセキュリティポリシーに注意を向ける必要があるでしょうか。認証に SSL は必要でしょうか。さらに、モバイルユーザーの人数は一定なのか、または将来増加する見込みなのかを調査してください。

第 4 章

Calendar Server の構成の計画

この章では、3 つの基本的な Calendar Server の構成について説明しており、この構成は使用しているサイトに特有の要件に応じて変更することができます。

この章で説明する内容は次のとおりです。

- Calendar Server の考慮事項
- 単一サーバーの最小構成
- ネットワークフロントエンドとデータベースバックエンドのサーバー構成
- 複数のフロントエンドサーバーとバックエンドサーバーの構成

Calendar Server の考慮事項

Calendar Server は次の主な 5 つのサービスから構成されています。

- HTTP サービス (`cshttpd`)。HTTP 要求を待機する。HTTP サービスはユーザー要求を受け取り、データを呼び出し元に返す
- 管理サービス (`csadmind`)。Calendar Server のそれぞれのインスタンスに必要とされる。管理サービスはシングルポイントの認証および Calendar Server の管理を提供し、ほとんどの管理ツールを提供する
- 通知サービス (`csnotify`)。電子メールまたは予定通知サービスのいずれかを使用して、予定および作業の通知を送信する
- 予定通知サービス (`enpd`)。予定アラームのブローカとして機能する
- 分散データベースサービス (`csdwpd`)。同じ Calendar Server システム内の複数のデータベースサーバー間でリンクを張り、分散型のカレンダ格納を形成する

スケーラブルな Calendar Server の配備の場合、HTTP サービスと管理サービスのインスタンスを Calendar フロントエンドシステムとして配備します。この設定では、マシンごとに cshttpd のインスタンスを 1 つ配備します。各マシンでは、マシンの CPU ごとに 1 つの cshttpd プロセスを設定します。通知サービス、予定通知サービス、分散データベースサービス、および管理サービスのインスタンスは、Calendar バックエンドシステムとして配備されます。

認証および XML と XSLT の変換は、多大なロードを生じさせる 2 つのカレンダサービスのアクティビティです。サービス品質の要件を満たすために CPU を追加することができます。

Calendar バックエンドサービスには、通常、Calendar フロントエンドサービスの CPU の半数が必要とされます。Calendar フロントエンドシステムによってサービス品質をサポートするには、フロントエンドの CPU の 2/3 前後を Calendar バックエンドシステムで使用する必要があります。

カレンダサービスをフロントエンドサービスとバックエンドサービスに分割することは、配備の初期の段階で考慮する必要があります。

通常、フロントエンドサービスのコンポーネントである Calendar Server HTTP プロセスは、CPU 時間を多く使用します。このことは、カレンダのピーク使用率を十分に考慮および配慮し、予測されるピーク HTTP セッションに対応するため、十分なフロントエンドの処理能力を選択することを示唆しています。通常、冗長性、つまり複数のフロントエンドホストを配備することによって、Calendar Server フロントエンドの使用可能性が向上します。フロントエンドシステムは持続的なカレンダデータをまったく保持しないので、Sun Cluster などの HA ソリューションの使用にはあまり向いていません。さらに、そのようなソリューションを使用する際のハードウェアの追加や管理オーバーヘッドにより、HA の Calendar Server フロントエンドへの配備のコストと時間がかかります。

注 本来の HA ソリューションを保証する Calendar フロントエンドの唯一の構成は、Messaging Server MTA ルーターを含む同じホストに Calendar フロントエンドを配備している場合です。ただし、この構成でも、そのようなソリューションのオーバーヘッドについては、利点がわずかなことからして、注意深く比較検討する必要があります。

Calendar Server フロントエンドのハードウェアの適切な選択は、1 つまたはデュアルのプロセッサ SPARC あるいは Intel サーバーです。マシンごとに Calendar Server cshttpd プロセスのインスタンスを 1 つ配備します。そのような配備によってコスト効率の良いソリューションが提供され、一定レベルの初期のクライアント並行性機能から開始し、ピーク使用率レベルがわかるにつれ、既存の構成にクライアントセッション機能を追加していくことができます。

複数のフロントエンドを配備する場合、フロントエンドサービス全体にロードを分散するにはスティッキ接続や持続的接続を備えるロードバランサが必要です。

Calendar Server バックエンド サービスは、リソースの消費で十分にバランスが取れているので、CPU あるいはディスクまたはネットワークなどの I/O のいずれにおいても、ボトルネックが形成されるという証拠はありません。このため、バックエンドのハードウェアな適切な選択は、1 つのストライプボリュームを備える SPARC サーバーになります。そのようなマシンはピーク時の大量のカレンダロードに対してかなりの容量を提供します。

要件の中に高可用性がある場合、バックエンドには持続的データが含まれているので、Calendar Server バックエンドを Sun Cluster で配備するのが妥当です。

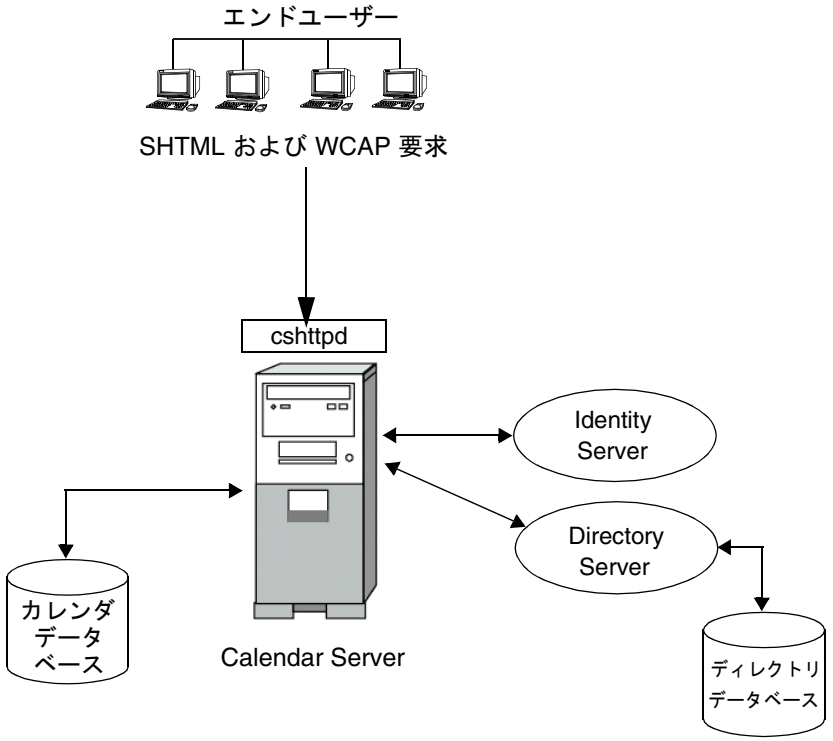
注 フロントエンドおよびバックエンドの Calender Server ホストの両方を持つ構成では、ホストはすべて次の場所で稼動している必要があります。

- 同じオペレーティングシステム
- 同じリリースの Calendar Server (パッチやホットフィックスのリリースを含む)

単一サーバーの最小構成

46 ページの図 4-1 に示す単一サーバーの最小構成では、すべての Calendar Server サービス (プロセス) が 1 つのサーバーの 1 つの CPU (プロセッサ) または複数の CPU で稼動します。ディレクトリサーバーと Sun Java System Identity Server のプロセスは、同じサーバーまたは異なるサーバーで実行できます。

図 4-1 単一サーバーの Calendar Server の最小構成



単一サーバー上の Calendar Server インスタンスには、次のサービスが含まれます。

- 管理サービス (`csadmind` プロセス)。Calendar Server の起動と停止、カレンダユーザーまたはリソースの作成と削除、カレンダのフェッチと格納を行うコマンドなど、管理機能をサポートする
- HTTP サービス (`cshttpd` プロセス)。受け取った SHTML および WCAP 要求を処理する

Calendar Server サービスの詳細については、『Sun Java System Calendar Server 管理ガイド』を参照してください。

最小構成ではデータベースは同じサーバーに配置されるため、カレンダデータベースが別のサーバーに配置されている環境でネットワーク機能を提供する DWP (データベースワイヤプロトコル) サービス (`csdwpd` プロセス) は必要ありません。

Calendar Server は、ユーザーの認証とユーザー設定の格納に使用するディレクトリサーバーを必要とします。通常は、Sun Java System Directory Server などの LDAP ディレクトリサーバーを使用します。ただし、Calendar Server API (CSAPI) を使用して、LDAP 以外のディレクトリサーバーを使用するためのプラグインを記述することもできます。この API については『Sun Java System Calendar Server Developer's Guide』を参照してください。

ディレクトリサーバーは、Calendar Server が稼動しているサーバーに配置することも、リモートサーバーに配置することもできます。

Sun Java System Identity Server (リリース 2003Q4 (6.1) 以降) には次の機能があります。

- commadmin ユーティリティ : Calendar Server を含む Sun Java System コミュニケーションサーバーの、ホストしている (仮想) ドメイン、ユーザー、グループ、組織、リソース、ロールをプロビジョニングおよび管理するときは、この CLI ユーティリティを使用します。

  `commadmin` ユーティリティについては、『Sun Java System Communications Services User Management Utility Administration Guide』を参照してください。

- シングルサインオン (SSO): Identity Server の使用または信頼できるサークルテクノロジによって、Calendar Server や Messaging Server を含む Sun Java Enterprise System サーバーに SSO を実装することができます。Identity Server は、Sun Java Enterprise System サーバーの SSO ゲートウェイとして機能します。ユーザーは Identity Server にログインすると、すべてのサーバーで SSO が適切に設定されている限り、その他のサーバーにもアクセスできます。

- Sun Java System LDAP Schema 2: このバージョンのスキーマを利用するには、Identity Server (リリース 2003Q4 以降) が必要です。

上記内容の詳細については 『Sun Java System Calendar Server 管理ガイド』を参照してください。

Identity Server は、Calendar Server が稼動しているサーバーに配置することも、リモートサーバーに配置することもできます。

エンドユーザーは、2 つの Web ユーザーインタフェース (UI) の 1 つ、つまり Sun Java System Calendar Express または Sun Java System Communications Express のいずれかを使用して、クライアントマシンから Calendar Server に接続します。各ユーザーインタフェースの詳細については、それぞれのインタフェースのオンラインヘルプを参照してください。さらに、次の Web サイトから入手できる『Sun Java System Communications Express 管理ガイド』も参照してください。

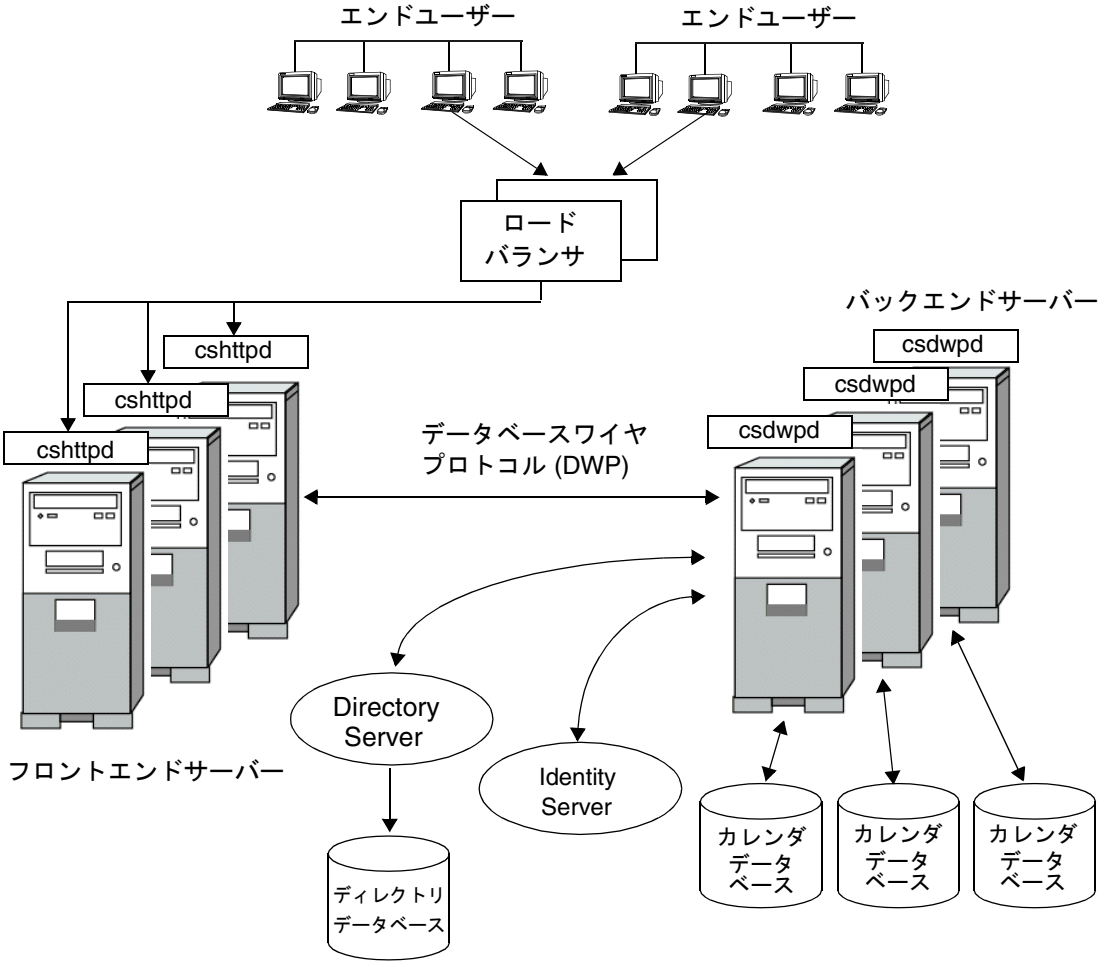
http://docs.sun.com/db/prod/entsys?l=ja

ネットワークフロントエンドとデータベースバックエンドのサーバー構成

Calendar Server は、複数のフロントエンドサーバーとバックエンドサーバーに設定を分配することにより、スケーラビリティを実現します。サーバーごとに、Calendar Server サービス (プロセスまたはデーモン) を複数の CPU (プロセッサ) に配布することもできます。

49 ページの図 4-2 に示すネットワークフロントエンドとデータベースバックエンドのサーバー構成では、ユーザーはフロントエンドサーバーにログインし、DWP (データベースワイヤプロトコル) サービス (`csdwpd` プロセス) を使用してバックエンドサーバーに接続します。カレンダデータベースは、バックエンドサーバーだけに接続されています。

図 4-2　ネットワークフロントエンドとデータベースバックエンドのサーバー構成



フロントエンドサーバーとバックエンドサーバーの両方で実行される Calendar Server プロセスは次のとおりです。

- ユーザーはロードバランサによってフロントエンドサーバーに誘導され、そこでログインします。それぞれのフロントエンドサーバーは次のサービスを必要とします。
  - 管理サービス (`csadmind` プロセス)
  - HTTP サービス (`cshttpd` プロセス)

- 各バックエンドサーバーにはカレンダデータベースが接続されるため、各バックエンドサーバーは次のサービスを必要とします。
  - 管理サービス (csadmind プロセス)
  - 予定通知サービス (enpd および csnotifyd プロセス)
  - カレンダデータベース用にフロントエンドサーバーにネットワーク機能を提供する DWP (データベースワイヤプロトコル) サービス (csdwpd プロセス)

この構成では、ユーザーはバックエンドサーバーにログインしないため、HTTP サービス (cshttpd プロセス) は必要ありません。

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

スケーラブルな Calendar Server の構成には、ユーザーの認証とユーザー設定の格納に使用するディレクトリサーバーが必要です。

Identity Server (リリース 6.1 (リリース 6 2003Q4) 以降) を使用して、シングルサインオン (SSO) の実装、Sun Java Enterprise System LDAP Schema 2 の使用、ホストしている (仮想) ドメイン、ユーザー、グループ、組織、リソース、ロールのプロビジョニングと管理を行うことができます。

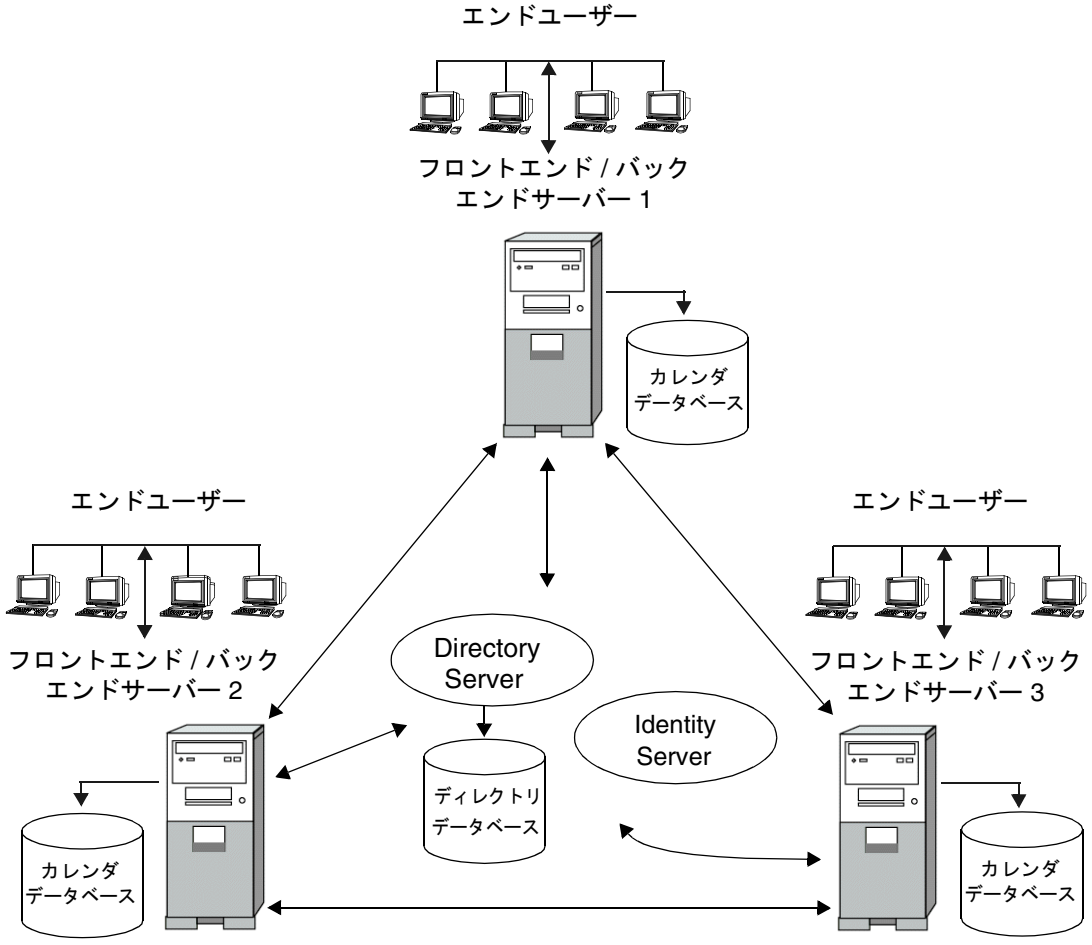
エンドユーザーは、2 つの Web ユーザーインタフェース (UI) の 1 つ、つまり Sun Java System Calendar Express または Sun Java System Communications Express のいずれかを使用して、クライアントマシンから Calendar Server に接続します。各ユーザーインタフェースの詳細については、それぞれのインタフェースのオンラインヘルプを参照してください。さらに、次の Web サイトから入手できる『Sun Java System Communications Express 管理ガイド』も参照してください。

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

複数のフロントエンドサーバーとバックエンドサーバーの構成

51 ページの図 4-3 に示す複数のフロントエンドサーバーとバックエンドサーバーの構成では、ユーザーは特定のサーバーにログインし、各サーバーはカレンダデータベースに接続されます。この構成では、カレンダを物理的に配布することができます。各サーバーにはカレンダが配置され、その所有者が Calendar Server にログインします。

図 4-3　複数のフロントエンドサーバーとバックエンドサーバーの構成



フロントエンドおよびバックエンドの各サーバーには次のすべての Calendar Server サービスが必要です。管理サービス (csadmind プロセス)、HTTP サービス (cshttpd プロセス)、予定通知サービス (enpd および csnotifyd プロセス)、DWP (データベースワイヤプロトコル) サービス (csdwpd プロセス)。

Calendar Server サービスの詳細については、『Sun Java System Calendar Server 管理ガイド』を参照してください。

複数のフロントエンドサーバーとバックエンドサーバーの構成には、ユーザーの認証とユーザー設定の格納に使用するディレクトリサーバーが必要です。

Identity Server (リリース 6.1 (リリース 6 2003Q4) 以降) を使用して、シングルサインオン (SSO) の実装、Sun Java Enterprise System LDAP Schema 2 の使用、ホストしている (仮想) ドメイン、ユーザー、グループ、組織、リソース、ロールのプロビジョニングと管理を行うことができます。

エンドユーザーは、2 つの Web ユーザーインタフェース (UI) の 1 つ、つまり Sun Java System Calendar Express または Sun Java System Communications Express のいずれかを使用して、クライアントマシンから Calendar Server に接続します。各ユーザーインタフェースの詳細については、それぞれのインタフェースのオンラインヘルプを参照してください。さらに、次の Web サイトから入手できる『Sun Java System Communications Express 管理ガイド』も参照してください。

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

第 5 章

Calendar Server スキーマおよびプロビジョニングのオプションについて

この章では、Calendar Server のスキーマおよびプロビジョニングのオプションについて説明しています。Calendar Server のプロビジョニングは複雑なので、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。

この章で説明する内容は次のとおりです。

- Calendar スキーマの選択について
- Calendar Server プロビジョニングツールについて

Calendar スキーマの選択について

この節では、Calendar Server で使用可能な、サポートされている 2 つのスキーマオプションと、どちらを使用するかを決定する方法について説明しています。

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』を参照してください。

将来のリリースでは、Schema 1 のインストールおよびプロビジョニングのサポートが廃止され、削除される予定です。しかし、独自のプロビジョニングツールをご使用のお客様は、引き続き LDAP Schema 1 を使用することができます。

使用するスキーマの決定

インストールに適したスキーマの選択は、使用するプロビジョニングの次のようなニーズに左右されます。

- Calendar Server を、シングルサインオン機能を搭載している Sun Java System Portal Server または Sun Java System Identity Server などの他の Java Enterprise System コンポーネント製品と統合するか

  答えが「はい」の場合、Schema 2 ネイティブモードを使用する必要があります。

- 初めて Calendar Server をインストールするのか、または旧バージョンからアップグレードするのか

  Calendar を初めてインストールする場合は Schema 2 ネイティブモードを使用します。

- プロビジョニングまたはシングルサインオンのいずれかに、Sun Java System Identity Server CLI ユーティリティの使用を計画しているか

  答えが「はい」の場合、Schema 2 のネイティブモードまたは互換モードを使用します。

- Calendar Server の旧バージョンからアップグレードする場合は、Schema 1 または Schema 2 のネイティブモードまたは互換モードのいずれでも使用することができる

- Calendar Server `csdomain` ユーティリティを使用してドメインをプロビジョニングする予定か

  答えが「はい」の場合、Schema 2 のネイティブモードまたは互換モードを使用します。Identity Server 6.1 の機能を使用しない場合、あるいは Calendar Server とその他の Java Enterprise System 製品を統合しない場合は、Schema 1 を使用します。

- プロビジョニングに Sun Java System Identity Server や Calendar Server CLI のいずれのユーティリティも使用しない場合、新規のインストールには Schema 2 ネイティブモード、既存の Calendar Server のインストールには Schema 1 または Schema 2 互換モードのいずれかが使用できる

LDAP Schema 1

LDAP Schema 1 は、組織ツリーと DC ツリーの両方から構成されるプロビジョニングスキーマです。当時は単に「スキーマ」と呼ばれた、このスキーマのセットは、以前の Calendar Server 5.x バージョンでサポートされていました。

Calendar Server はユーザーやグループのエントリを検索する場合、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調べ、`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のエントリが入っている組織サブツリーへの DN 参照が保持されています。

Calendar Server の旧バージョンをインストール済みのサイトだけが、Schema 1 を使用する必要があります。

注	将来、他の Sun Java System 製品とともに Calendar Server をインストールすることを計画している場合、Schema 2 への移行は必須です。

サポートされているプロビジョニングツール

Schema 1 は LDAP プロビジョニングツールをサポートしています。詳細については、56 ページの「Calendar Server プロビジョニングツールについて」を参照してください。

Schema 2 (ネイティブモード)

Schema 2 は、新しく定義された一連のプロビジョニングの定義で、Directory Server LDAP を使用して、エントリとして格納可能な情報のタイプを説明します。

このネイティブモードは、検索テンプレートを使用して LDAP ディレクトリサーバーを検索します。ドメイン検索テンプレートを使用していったんドメインが見つかると、特定のユーザーまたはグループの検索にはユーザーまたはグループ検索テンプレートが使用されます。

初めて Calendar Server をインストールし、マシンに 2 つのツリープロビジョニングモデルに依存する他のアプリケーションがない場合、ネイティブモードを使用する必要があります。Java Enterprise System 製品群の他の製品をインストールする場合も、このモデルを使用する必要があります。

Schema 1 を使用する既存の Calendar Server 5.x がインストールされており、Calendar Server と他の Java Enterprise Server 製品を統合する場合は、Calendar Server 6 にアップグレードしたあとに使用しているディレクトリを Schema 2 に移行する必要があります。LDAP Schema バージョン 1 から 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』を参照してください。

注	Schema 2 ネイティブモードは、Java Enterprise System 製品群の Sun Java System の全製品で推奨されているプロビジョニングモデルです。

サポートされているプロビジョニングツール

Schema 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。詳細については、56 ページの「Calendar Server プロビジョニングツールについて」を参照してください。

Schema 2 互換モード

Schema 2 互換モードは、Schema 1 と Schema 2 ネイティブモードの間の暫定的なモードです。Schema 2 互換モードは、両方のスキーマをサポートし、すでに使用している既存の 2 つのツリー設計を維持することができます。さらに Schema 2 互換モードは、Messaging Server のインストールに先立って Identity Server をインストールしていることを前提としています。

Schema 1 が必要な既存のアプリケーションを使用し、かつ、Identity Server やシングルサインオンなどの Schema 2 を必要とする機能も必要な場合には Schema 2 互換モードを使用してください。

注 Schema 2 互換モードは、Schema 2 ネイティブモードへの移行の便宜を図り提供されています。最終的なスキーマの選択として Schema 2 互換モードを使用しないでください。Schema 1 から Schema 2 互換モードへ、そして最終的には Schema 2 ネイティブモードへの移行プロセスは、Schema 1 から Schema 2 ネイティブモードへ単に移行するより複雑です。詳細については、『Sun Java System Communications Services Schema Migration Guide』を参照してください。

Calendar Server プロビジョニングツールについて

サポートされている Calendar Server プロビジョニングツールを使って、LDAP ディレクトリのユーザー、グループ、およびドメインのエントリ情報の照会、変更、追加、および削除を行うことができます。この節では、これらの Calendar Server プロビジョニングツールについて説明します。

54 ページの「使用するスキーマの決定」で扱われている質問に加えて、58 ページの表 5-1 を使用してスキーマおよびプロビジョニングツールのオプションを評価する必要があります。

注 Calendar Server のインストールおよび設定に先立って、Calendar Server エントリをプロビジョニングするためのスキーマおよびツールを決定する必要があります。

次の節では、サポートされているプロビジョニングツールについてレベルの高い情報を提供しています。

- 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
- プロビジョニングツールのオプションの比較

LDAP プロビジョニングツール

Schema 1 のユーザーやグループは、LDAP Directory ツールを使用してプロビジョニング可能です (Schema 2 ではサポートされていない)。委任された管理者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しなくても、LDAP を介して LDIF レコードの追加、削除、および変更を行うことによって、ユーザーおよびグループを直接プロビジョニングすることができます。

ユーザー管理ユーティリティ

Sun Java System Identity Server は Schema 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は Schema 2 を使用するので、Communications Services 6 ユーザー管理ユーティリティを使用してください。Java Enterprise System 製品を複数使用する場合や Calendar Server の新規インストールを実行する場合には、特にそのようにする必要があります。

注	Identity Server をインストールする場合でも、Calendar Server と互換性があるグラフィカルユーザーインタフェースはありません。したがって、インタフェースでユーザーやグループをプロビジョニングするのに使用できるのは、ユーザー管理ユーティリティのみです。

インストールの詳細については、『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のオプションの比較

58 ページの表 5-1 は、種々のサポートされているスキーマ、プロビジョニングツール、プロビジョニングの制約、および詳細な情報を得るために推奨されているマニュアルを示しています。

表 5-1　Calendar 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約	参照するマニュアル
LDAP プロビジョニングツール 使用するスキーマ : Schema 1	LDAP エントリを直接変更する、またはカスタムプロビジョニングツールを作成するためのツール	• Sun Schema 2 および他の Java Enterprise System 製品と互換性がない	『Sun ONE Calendar Server 5.2 Provisioning Guide』および『Sun ONE Messaging and Collaboration スキーマリファレンス』を参照 Sun LDAP Schema 1 プロビジョニングモデルを説明している さらに、これらのマニュアルは LDAP プロビジョニングツールの使用方法および特定の属性やオブジェクトクラスの使用についても説明している
ユーザー管理ユーティリティ 使用するスキーマ : Schema 2	ユーザー、グループ、ドメイン、およびメーリングリストを管理するための管理者用のコマンド行インタフェースを提供する 他の Java Enterprise System 製品と互換性がある	• Sun Schema 1 との下位互換性がない • Sun Java System Identity Server で使用する GUI プロビジョニングツールがない • このコマンド行インタフェースを有効にするには Sun Java System Identity Server のインストールが必要	『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について説明している

第 6 章

安全な Calendar Server の設計

この章では、セキュリティ方法の概要と、広く見られるセキュリティ上の危険性について説明し、セキュリティの必要性を分析する手順を概説しています。

この章で説明する内容は次のとおりです。

- セキュリティ戦略の作成
- Calendar のセキュリティの概要
- ユーザー認証の計画
- セキュリティの誤解について
- 他のセキュリティリソース

セキュリティ戦略の作成

セキュリティ戦略の作成は、配備を計画する際のもっとも重要な手順の 1 つです。計画する戦略は、組織のセキュリティニーズに対応し、ユーザーを威圧することなく、セキュリティ保護されたカレンダ環境を提供する必要があります。

さらに、セキュリティ戦略は、単純で管理可能なものである必要があります。複雑なセキュリティ戦略は、ユーザーが自分のメールにアクセスできなくなるような間違いにつながる場合があります。または、アクセスを許可していない情報の変更や検索をユーザーや権限のない侵入者に許してしまう可能性があります。

RFC 2196、『Site Security Handbook』には、セキュリティ戦略を開発するための次のような 5 つの手順が示されています。

1. 保護しようとしている対象を識別する

   たとえば、リストには、ハードウェア、ソフトウェア、データ、従業員、マニュアル、ネットワークインフラストラクチャ、組織の評判などを含めることができます。

2. 何から保護しようとしているかを特定する

   たとえば、カレンダ、予定、または作業に対する不正なアクセスなどがあります。

3. システムにおける危険性の度合いを予測する

   規模の大きなサービスプロバイダの場合、セキュリティの危険性の度合いは規模の小さい組織より大きくなります。さらに、組織の性格もセキュリティの危険性を誘発することがあります。

4. コスト効率の良い方法で資産を保護する方策を実装する

   たとえば、SSL 接続を設定する際の余分のオーバーヘッドは、Calendar の配備のパフォーマンスに負担になる可能性があります。セキュリティ戦略を設計する際、セキュリティの必要性とサーバーの容量とのバランスを取る必要があります。

5. 継続的に使用している戦略を検討し、弱点を見つけしだい改善を図る

   定期的に監査を行い、セキュリティポリシー全体の効率を検証します。Calendar Server のログファイルを調査することによってこれを行うことができます。詳細については、『Sun Java System Calendar Server 管理ガイド』を参照してください。

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

さらに次の項目についてセキュリティ戦略を計画する必要があります。

- 物理的セキュリティ
- サーバーのセキュリティ
- ネットワークのセキュリティ
- Calendar のセキュリティの概要

物理的セキュリティ

インフラストラクチャの重要な部分への物理アクセスを制限してください。たとえば、盗用、改ざん、または他の誤用を防止するため、ルーター、サーバー、ワイヤリングクローゼット、サーバールーム、またはデータセンターに物理的な境界線を設けます。権限のない人物がサーバールームに入室し、ルーターのプラグを抜くことができるなら、ネットワークおよびサポートのセキュリティはまだ解決されていない問題ということになります。

サーバーのセキュリティ

Solaris™ Security Toolkit などのセキュリティ製品を使用してオペレーティングシステムの堅固することを考慮してください。さらに、サーバーへの Telnet、FTP、rlogin によるアクセスなどの使用していないサービスや設備を除去します。

多層環境では、バックエンドサーバーを設定して、適切なサービスを既知のフロントエンドサーバーのみに提供するようにしてください。

重要なオペレーティングシステムのアカウントやデータへのアクセス制限も、あらゆるセキュリティ戦略の一部です。オペレーティングシステムで使用可能な認証やアクセス制御メカニズムによって保護を実現できます。

さらに、数か月に一度またはベンダーからのセキュリティ警告に応じて、オペレーティング環境の最新のセキュリティパッチをインストールし、パッチの更新を設定する必要があります。

ネットワークのセキュリティ

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部分です。通常、ネットワークへのアクセス全体はファイアウォールの使用により制限されています。しかし、電子メールはサイトの外部でも使用できるようにする必要があります。SMTP はそのようなサービスの 1 つです。

ネットワークのセキュリティを保護するには、次のことをする必要があります。

- 使用していないポートを待機するオペレーティングシステムに搭載されたすべてのサービスを停止する
- 可能な場合、`telnet` を `sshd` に置き換える
- サーバーを、内部のソース IP アドレスを持つ外部パケットを排除する、パケットフィルタの後ろに設置する。パケットフィルタは、明示的に指定されたポートを除き、外部からの接続すべてを禁止する

Calendar のセキュリティの概要

セキュリティは、今日の企業の日々の業務の中で重要な役割を果たしています。セキュリティの侵害は、企業秘密を危険にさらす可能性だけでなく、停止時間、データの破損、および運用コストの増大を招く可能性もあります。Calendar Server は、ユーザーを盗聴、不許可の使用、または外部からの攻撃から保護するためにいくつかのセキュリティレベルを提供します。基本レベルのセキュリティは認証によるものです。Calendar Server は、デフォルトの設定で LDAP 認証を使用していますが、代替の認証方法が必要とされる場合、認証プラグインの使用もサポートしています。さらに Identity Server と統合する場合、Calendar Server はそのシングルサインオン機能を利用することができます。

セキュリティにはユーザーが本人であることを確かめる以上のことが関係しています。セキュリティにはデータの機密保護を確実にすることも含まれます。このため、Calendar Server はログインまたはログインとデータに対して、SSL 暗号化技術の使用をサポートしています。つまり、Web クライアントからサーバーへ、ログインのみが暗号化される場合と、ログインを含むセッション全体が暗号化される場合があります。

Java System Portal Server、Secure Remote Access 製品との統合によっても SSL 暗号化技術が可能になりますが、プロキシゲートウェイを介します。さらに、ポータルゲートウェイとの統合により提供される URL 書き換え機能により、外部のエンティティからさらに Calendar Server を隔離することが可能になります。Calendar Server は、ゲートウェイを介さない Calendar Server と直接接続ができないようにする、ポータルゲートウェイとともに配備することができます。この場合、すべての URL が書き換えられるため、Calendar Server の本当の URL の特定が困難になります。ユーザーが認証される場合でも、それによって、そのユーザーに他のカレンダユーザーのデータへのアクセス権があるわけではありません。

カレンダドメインの中には、認証されたユーザーが他の認証されたユーザーのカレンダデータにアクセスできないようにする、他のセキュリティ層があります。セキュリティの方策の 1 つに、Calendar Server アクセス制御のエントリによる方法があります。アクセス制御によって、カレンダユーザーは、自分のカレンダを閲覧可能な人、予定を自分のカレンダにスケジュール設定可能な人、自分のカレンダを変更可能な人、自分のカレンダから予定を削除可能な人を指定することができます。さらにアクセス制御によって、ユーザーは、自分の代わりに招待に応答することのできる人、予定のスケジュール設定または変更ができる人、予定を削除できる人を選択することができます。最後に、アクセス制御を使用すると、ユーザーのドメインの範囲を調整できるので、あるドメインのユーザーが別のドメインのユーザーによる予定のスケジュール設定を防止したり、または可能にしたりすることができます。

ただし、アクセス制御に加えて、Calendar Server は、カレンダフロントエンドとデータベースバックエンドを分割する配備に、データベースプロトコルレベルの他のレベルのセキュリティを装備することができます。このセキュリティレベルは、データベースワイヤプロトコル (DWP) 認証と呼ばれ、ユーザーの名前とパスワードのペアを利用して DWP 接続を認証します。DWP 接続が認証されるためには、ユーザー ID とパスワードのペアが、フロントエンドとデータベースバックエンドの両方で同じである必要があります。

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の重要な部分です。システムへの攻撃を識別するため、メッセージキューサイズ、CPU 使用率、ディスクの可用性、およびネットワーク使用率を監視してください。メッセージキューサイズの異常な増大やサーバー応答時間の異常な減少から、攻撃のいくつかは識別できます。さらに、異常なシステムロードパターンや異常な接続を調査してください。毎日ログを検査して何らかの異常な動作がないかを確認します。

ユーザー認証の計画

ユーザー認証によって、ユーザーはカレンダクライアントを介してログインし、自分のカレンダ情報を取得できます。ユーザー認証の方法には次のものがあります。

- プレーンテキストと暗号化パスワードによるログイン
- SSL (Secure Sockets Layer) を使用する証明書ベースの認証

プレーンテキストと暗号化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 ディレクトリに格納されています。最小限の長さなど、パスワードのセキュリティ基準はディレクトリポリシーの要件で決定されます。パスワードのセキュリティ基準は Calendar Server の管理範囲外のものです。ディレクトリサーバーのパスワードポリシーについては、次の Web サイトから入手できる『Sun Java System Directory Server 配備計画ガイド』を参照してください。

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

プレーンテキストと暗号化パスワードによるログインが使用可能です。

SSL (Secure Sockets Layer) を使用する証明書ベースの認証

Calendar Server は SSL プロトコルを使用して、暗号化通信やクライアントと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 は公開鍵暗号化技術の概念に基づいています。機能的に TLS (Transport Layer Security) は SSL のスーパーセットですが、その名前は同じように使用されています。

高いレベルでは、SSL サポートするサーバーには、証明書、公開鍵、非公開鍵、証明書、鍵、およびセキュリティのデータベースが必要とされます。これによって、メッセージの認証、プライバシ、および完全性が確保されます。

SSL で認証するため、カレンダクライアントはサーバーと SSL セッションを確立し、ユーザーの証明書をサーバーに送信します。次に、サーバーは送信された証明書の信頼性を評価します。証明書が検証された場合、ユーザーは認証されたとみなされます。

認証に SSL を使用する場合は、Calendar Server のサーバー証明書を取得する必要があります。クライアントや他のサーバーは、その証明書によってサーバーを識別します。サーバーは、自らを明確にする複数のサーバー証明書を保持することができます。さらにサーバーは、信頼されている認証局 (CA) の証明書をいくつでも保持し、クライアントの認証に使用することができます。

SSL の詳細については 次の Web サイトから入手できる『Sun Java System Calendar Server 管理ガイド』を参照してください。

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

セキュリティの誤解について

この節では、配備のセキュリティニーズに逆効果になる一般的なメッセージングの誤解について説明します。

- **製品名およびバージョンの隠ぺい**

  製品名およびバージョンの隠ぺいは、せいぜいあまり本気ではない攻撃者を阻止するくらいです。最悪の場合、それによって誤ったセキュリティ感覚を持ってしまい、管理者が本当のセキュリティ問題を突き止めることに熱心ではなくなる原因となることがあります。

  実際のところ、製品情報およびバージョン番号を削除すると、ベンダーのサポート組織は、ソフトウェアの問題が自社のソフトウェアの問題なのか、または他のソフトウェアの問題なのかを確認することが困難になります。

  意志の固い攻撃者は、他のプロトコルの動作を使用し、隠ぺいするための何らかの努力に関係なくベンダーの名前およびバージョンを特定することができます。

- **内部マシンの名前の隠ぺい**

  内部 IP アドレスとマシン名を隠ぺいすると、次のことがさらに困難になります。

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

  やり手の攻撃者ならば、いったんネットワークの情報を不正に取得する方法を見つけると、何の問題もなくマシン名やマシンの IP アドレスを発見することができます。

- Network Address Translation (NAT)

  NAT を使用して一種のファイアウォールを装備している場合、システム間の終端間接続がありません。その代わりに、その中間に第 3 のノードがあります。この NAT システムは仲介役のような働きをし、セキュリティホールの原因となる恐れがあります。

- Calendar Server のプロセス

  普通は、Calendar Server のプロセスを `root` として実行していません。

他のセキュリティリソース

セキュリティ保護された Messaging の配備については、次の 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 7 章

インストール前の考慮事項

この章では、Calendar Server のインストール前に考慮が必要な事項について説明します。

この章で説明する内容は次のとおりです。

- Calendar Server のインストール
- Calendar Server の管理者の計画
- ホストしているドメインの計画
- インストール後の設定

Calendar Server のインストール

Calendar Server のインストールと設定は、以前の Calendar Server リリース (2003Q4 以前のバージョン) に比べ大幅に変更されました。Calendar Server 用のスタンドアロンのインストーラはなくなりました。

まだ Calendar Server 2003Q4 (6.0) をインストールしていない場合は、Sun Java Enterprise System インストーラを使用して、2004Q2 バージョンを取得する必要があります。このインストーラを使用すると、他の Sun コンポーネント製品およびパッケージもインストールできます。Java Enterprise System インストーラについては、『Sun Java Enterprise System 2004Q2 インストールガイド』を参照してください。

Calendar Server 6 2003Q4 から Calendar Server 6 2004Q2 にアップグレードする場合のアップグレードプロセスについては、『Sun Java Enterprise System 2004Q2 インストールガイド』の「Java Enterprise System 2003Q4 からのアップグレード」で説明されています。

Calendar Server の旧バージョン (バージョン 5.x まで) からの移行については、『Calendar Server 管理ガイド』の「移行ユーティリティ」の章を参照してください。ここでは、5.x までをカバーしています。5.x 以降のバージョンからの移行については、Sun サポート担当者に連絡してください。

設定が必要な Calendar Server コンポーネント

Calendar Server ソフトウェアをインストールする際、Java Enterprise System インストーラは Calendar Server パッケージをすべてインストールします。次いで、Calendar Server 設定プログラムを使い、適切な Calendar Server コンポーネントを Calendar ホストに設定します。

次の表では、それぞれのタイプの Calendar ホストで、設定が必要なコンポーネントを示しています。

表 7-1 設定が必要な Calendar Server コンポーネント

構成対象の Calendar ホストのタイプ	設定プログラムで選択が必要なコンポーネント
フロントエンド	HTTP サービスおよび管理サービス
バックエンド	通知サービス、予定通知サービス、分散データベースサービス、および管理サービス

分散データベースサービス (csdwpd) は、バックエンドサーバー、つまりカレンダデータベースのあるサーバーにのみ必要とされ、ユーザーアクセスサービス (cshttpd) を提供しません。これは、カレンダデータベースのないフロントエンドサーバーには必要とされません。csdwpd サービスを使用することで、同じ Calendar Server 設定内のフロントエンドとバックエンドのサーバーをリンクし、分散型のカレンダ格納を形成することができます。

Calendar Server の管理者の計画

Calendar Server の管理者には、次の管理者が含まれます。

- Calendar Server 管理者 (calmaster)
- Calendar Server ユーザーおよびグループ
- スーパーユーザー (root)

Calendar Server 管理者 (calmaster)

Calendar Server 管理者とは、ユーザー名とそれに関連付けられたパスワードの組み合わせのうち、Calendar Server の管理権限を付与されているユーザーのことです。たとえば、Calendar Server 管理者は Calendar Server サービスの起動と停止、ユーザーの追加と削除、カレンダの作成と削除などを実行できます。このユーザーは Calendar Server の管理権限を持ちますが、ディレクトリサーバーの管理権限を持つとは限りません。

Calendar Server 管理者のデフォルトのユーザー ID は `calmaster` ですが、Calendar Server の設定時に別のユーザーを指定することもできます。インストール後に別のユーザーを指定する場合は、`ics.conf` ファイルの `service.admin.calmaster.userid` パラメータの設定を変更します。

Calendar Server 管理者として指定するユーザー ID は、ディレクトリサーバー内の有効なユーザーアカウントである必要があります。Calendar Server の設定時に Calendar Server 管理者のユーザーアカウントがディレクトリサーバーに存在していない場合には、設定プログラムがアカウントを自動的に作成します。

`ics.conf` ファイルの Calendar Server 管理者の設定パラメータの全リストについては、CS AG を参照してください。

Calendar Server ユーザーおよびグループ

Solaris オペレーティングシステムでは、これらの特別なアカウントは Calendar Server の実行に使用されるユーザー ID とグループ ID を示しています。特別なアカウントが存在しないときは、設定プログラムによって自動的に作成されるデフォルト値 `icsuser` および `icsgroup` を使用することをお勧めします。ただし、Calendar Server 設定プログラムの実行時に `icsuser` および `icsgroup` 以外の値を指定することもできます。これらの値は、それぞれ `ics.conf` ファイルの `local.serveruid` および `local.servergid` パラメータに格納されます。

スーパーユーザー (root)

Solaris™ オペレーティング システムでは、Calendar Server をインストールするには superuser (root) としてログインするか、あるいは superuser になる必要があります。コマンド行ユーティリティを使用して superuser として実行し、Calendar Server を管理することもできます。ただし、一部のタスクについては Calendar Server ファイルへのアクセスの問題を回避するために、superuser としてではなく、icsuser および icsgroup (または選択した値) として実行する必要があります。

ホストしているドメインの計画

Calendar Server はホストしている (または仮想) ドメインをサポートしています。ホストしているドメインのインストールでは、各ドメインが Calendar Server の同じインスタンスを共有するため、1 つのサーバーに複数のドメインが存在できます。各ドメインはネームスペースを定義し、1 つのネームスペースではすべてのユーザー、グループ、リソースが一意です。各ドメインには、変更可能な属性とユーザー設定もあります。

ホストしているドメインのインストールおよび設定には、Schema 2 だけを使用してください。

ホストしているドメインをサーバーにインストールおよび設定するには、次の高レベルの手順を実行します。

1. Directory Server をインストールおよび設定する
2. Web Server 6 をインストールおよび設定する
3. Identity Server をインストールおよび設定する

   ユーザー管理ユーティリティが Identity Server とともにインストールされます。
4. Calendar Server をインストールする
5. comm_dssetup.pl スクリプトを実行する

   このスクリプトの実行手順については、次の Web サイトから入手できる『Sun Java System Calendar Server 管理ガイド』の第 2 章を参照してください。

   http://docs.sun.com/db/prod/entsys?l=ja
6. Communications Services ユーザー管理ユーティリティを設定する

   commadmin の設定や使用の手順については、『Sun Java System Communications Services User Management Utility Administration Guide』を参照してください。
7. デフォルトドメインおよびサイト管理者 (calmaster) を作成する

デフォルトドメインは commadmin の設定時に作成されます。ただし、ドメインエントリを変更して、Calendar または Mail サービスを追加する必要があることに注意してください。また、サイトカレンダ管理者 (calmaster) を設定する必要があります。これらの 2 つのタスクの実行方法の手順については、『Sun Java System Calendar Server 管理ガイド』の「インストール後の設定」を参照してください。

8. Calendar Server を設定する

   csconfiguratior.sh プログラムの実行手順については、『Sun Java System Calendar Server 管理ガイド』の第 3 章を参照してください。

9. Calendar Server のホストしているドメインの設定パラメータを設定する

   設定パラメータおよびその値のリストについては、『Sun Java System Calendar Server 管理ガイド』の「ホストしているドメインの設定」を参照してください。

10. commadmin を使用してサイトのホストしているドメインを作成する

11. commadmin を使用してホストしているドメインにユーザーおよびリソースを配置する

12. Calendar Server サービスを起動する

    手順については、『Sun Java System Calendar Server 管理ガイド』を参照してください。

上記の各手順の詳細については、該当する主要およびコンポーネント製品のマニュアルを参照してください。

注　常時 Communications Services ユーザー管理ユーティリティインタフェースを使用して、Schema 2 のプロビジョニングを行ってください。

Schema 1 のプロビジョニングツールはホストしているドメインをサポートしていません。

インストール後の設定

Calendar Server 6 2004Q2 をインストールしたら、その設定をする必要があります。この手順は、従来インストールプロセスの一部として行われましたが、現在インストーラから分離されています。

Calendar Server をインストールしたあと、次のように Calendar Server を設定する必要があります。

1. Directory Server 設定スクリプト (`comm_dssetup.pl`) を実行し、Sun Java System Directory Server 5.x を設定します。
2. Calendar Server 設定プログラム `csconfigurator.sh` を実行してサイト固有の要件を設定し、新しい `ics.conf` 設定ファイルを作成します。`ics.conf` ファイルのパラメータの説明については、次の Web サイトから入手できる『Sun Java System Calendar Server 管理ガイド』を参照してください。

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

`comm_dssetup.pl` と `csconfigurator.sh` は、どちらも `/opt/SUNWics5/cal/sbin` ディレクトリに格納されています。

Java Enterprise System インストーラおよび Calendar Server 設定ユーティリティ (`csconfigurator.sh`) が実行しない、構成の設定や変更がいくつかあります。次の項目は手動で変更する必要があります。

- **DWP および CLD の設定** : `ics.conf` ファイルを編集して、CLD キャッシュオプションを有効にしてください。このキャッシュがカレンダユーザーの DWP ホストサーバー情報を格納することで、LDAP ディレクトリサーバーに対する呼び出しを減らすことができます。
- **デフォルトのタイムゾーン** : デフォルトのタイムゾーンがアメリカ / ニューヨークでない場合、`ics.conf` ファイルを編集して変更してください。さらに、`/opt/SUNWics5/cal/bin/html/default_user_prefs.xml` ファイルも変更して、`ics.conf` ファイルと同期するようにする必要があります。
- **クライアント側のレンダリング** : Calendar Server では、XSLT 処理をエンドユーザーのブラウザにダウンロードすることで、クライアント側のレンダリングを実行します。このため、Calendar Server が実行する必要のある処理は減少します。Calendar Server は、ブラウザが XSLT 処理のレンダリングに対応している場合にだけ XSLT 処理をダウンロードします。現在のリリースでは、この機能は Internet Explorer 6.0 以降だけに適用されます。`ics.conf` ファイルを編集して、このようなパフォーマンスの改善をクライアント側のレンダリングで行ってください。
- **tmpfs の設定** : `tmpfs` の設定を編集してパフォーマンスを向上させてください。

これらの変更の詳細については 次の Web サイトから入手できる『Sun Java System Calendar Server 管理ガイド』を参照してください。

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

用語集

このマニュアルで使用している用語の全リストについては、『Java Enterprise System Glossary』(http://docs.sun.com/doc/816-6873) を参照してください。

索引

C

Calendar Server
  Calendar Express, 24
  Portal Server とともに使用 , 21
  エンドユーザー , 24
  業界標準のサポート , 17
  高可用性 , 21
  考慮事項 , 43
  コンポーネントのインストール , 68
  サービス , 43
  配備チームの創設 , 23
  配備プロセス , 22
  バックエンドシステム , 44
  プラグイン , 18
  フロントエンドシステム , 44
  利点 , 18
Calendar Server 設定プログラム , 72
CERT, 66
`comm_dssetup.pl` スクリプト , 72
Computer Emergency Response Team, 66
`csconfigurator.sh` スクリプト , 72

D

DC ツリー , 54
Directory Server 設定スクリプト , 72
DMZ, 39
DNS, 36
DNS 照会 , 39
DWP, 48, 50

H

HTTP サービス , 43

I

`ics.conf` 設定ファイル , 72
Identity Server, 52, 57
IP スプーフィング、保護 , 42

L

`local.servergid` パラメータ , 69
`local.serveruid` パラメータ , 69

M

Microsoft Outlook, 24

P

Portal Server, 21

S

Schema 1
- Calendar Server のサポート , 53
- 委任された管理者 , 55
- 説明 , 54

Schema 2
- Identity Server, 57
- 互換モード , 56
- 選択 , 54
- ネイティブモード , 55

Secure Sockets Layer, 64

あ

暗号化メール , 64

い

インストールと設定、計画 , 67

う

運用要件 , 28

え

エンドユーザーのパフォーマンス , 24

か

拡張計画 , 32
管理サービス , 43

こ

公開鍵 , 64
高可用性 , 21
構成例、水平スケーラビリティ , 48
互換モード、Schema 2, 56

さ

サービスレベル契約 , 31
サイドバーテキスト , 12
サポート要件 , 30

し

システムの監視 , 63
使用パターン , 29
証明書 , 64
シングルサインオン , 47

す

スイッチ , 36
スキーマバージョン、選択 , 54
ストレージエリアネットワーク , 38

せ

セキュリティ

Computer Emergency Response Team, 66
IP アドレスの隠ぺい , 65
Network Address Translation, 65
Secure Sockets Layer, 64
製品名およびバージョンの隠ぺい , 65
ネットワークへのアクセス制限 , 61
ハードウェアへの物理的アクセスを制限 , 61
パスワード , 64
評価の必要性 , 59
保護ソフトウェア , 61

そ

組織ツリー , 54

ち

地理的考慮事項 , 29

つ

通知サービス , 43
ツール
  比較 , 57
  プロビジョニング , 56

て

データベースワイヤプロトコル , 48, 50

な

内部ネットワーク , 41

ね

ネットワーク
  考慮事項 , 29
  スイッチ , 36
  非武装地帯 , 39
  ファイアウォール , 37
  ルーター , 36

は

ハードウェア、選択 , 32
配備
  運用要件 , 28
  拡張計画 , 32
  コストの制約 , 31
  サービスレベル契約 , 31
  サポート要件 , 30
  使用パターン , 29
  地理的考慮事項 , 29
  ネットワークの考慮事項 , 29
  ハードウェアのコスト , 32
  文化の側面 , 28
  目標の特定 , 27
配備プロセス , 22
配備目標の特定 , 27
パスワード
  暗号化 , 64
  プレーンテキスト , 64
  問題 , 64

ひ

非公開鍵 , 64
非武装地帯 , 39
表記規則
  このマニュアルでの使用 , 12
表記上の規則 , 12
  サイドバーテキスト , 12
  モノスペースフォント , 12

ふ

ファイアウォール
DMZ の分割 , 40
Network Address Translation, 65
設定 , 42
目的 , 37
プロキシ , 41
プロキシサーバー , 36, 41
プロジェクト目標の決定 , 32
プロビジョニングのオプション
LDAP ディレクトリツール , 57
スキーマバージョンの決定 , 54
ツールの比較 , 57
プロビジョニングツール , 56
文化の考慮事項 , 28
分散データベースサービス , 43

ほ

ホストしているドメイン , 70

も

モノスペースフォント , 12
モバイルユーザー , 42

よ

要件
技術 , 29
財務 , 31
ビジネス , 28
予定通知サービス , 43

る

ルーター , 36

ろ

ロードバランサ , 38, 45
ロードバランス , 37