

## Kerberos対応のLinux/UNIXアプリケーションに適したシングルサインオン

**著者:**

**Manny Vellon**  
*CTO*  
Likewise Software

**要約**

本書では、Likewiseがいかにしてエンタープライズシングルサインオン (SSO) の実装をスムーズに進めるかについて解説する。また、Likewiseが提供する認証インフラストラクチャを活用するためにKerberos対応アプリケーションをどのように構成できるかについて説明する。さらにその概念について解説するだけでなく、アプリケーションにおけるシングルサインオンをサポートできるようにする上で必要とされる具体的なステップについて概説する。

The information contained in this document represents the current view of Likewise Software on the issues discussed as of the date of publication. Because Likewise Software must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Likewise, and Likewise Software cannot guarantee the accuracy of any information presented after the date of publication.

These documents are for informational purposes only. LIKewise SOFTWARE MAKES NO WARRANTIES, EXPRESS OR IMPLIED.

Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in, or introduced into a retrieval system, or transmitted in any form, by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of Likewise Software.

Likewise may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from Likewise, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

© 2007 Likewise Software. All rights reserved.

Likewise and the Likewise logo are either registered trademarks or trademarks of Likewise Software in the United States and/or other countries. All other trademarks are property of their respective owners.

Likewise Software  
15395 SE 30th Place, Suite #140  
Bellevue, WA 98007  
USA

## 目次

|                                    |    |
|------------------------------------|----|
| はじめに .....                         | 4  |
| シングルサインオンとは .....                  | 5  |
| Kerberos とは .....                  | 6  |
| Likewise はどのように SSO をサポートするのか..... | 8  |
| 例: OpenSSH .....                   | 10 |
| SSH のサービスプリンシパル名 .....             | 10 |
| システムのキータブ生成.....                   | 10 |
| ユーザーのキータブ生成.....                   | 11 |
| OpenSSH の構成 .....                  | 11 |
| SSO のテスト .....                     | 12 |
| 例: Apache Tomcat .....             | 13 |
| まとめ .....                          | 14 |

## はじめに

Likewise を実装すると、Linux/UNIX コンピュータは、Microsoft Active Directory™ 経由でユーザーの認証や認可を行えるようになる。その結果、多数のメリットがもたらされる。

- Microsoft Windows™ ユーザーであるか非 Windows システム・ユーザーであるかに関係なく、ユーザーへ単一のユーザー名 / パスワードを提供
- ユーザーアカウント管理を大幅に簡素化。システム管理者は単一の ID 管理システムを利用し、ユーザーに対するプロビジョニング、パスワードの保守管理、プロビジョン解除を行える。
- セキュリティの強化。Likewise によって、多数の Active Directory アカウント・ポリシーが Linux、UNIX システムへ拡張される。管理者は、最短パスワード長、パスワードの複雑性要件や有効期間ポリシーなどを設定できる。これらの設定は Windows システムと非 Windows システムの両方に適用される。
- きめ細かな認可で職務分立をサポート。Likewise によって、Active Directory グループ・ポリシー機能を Linux や UNIX システムへ拡張でき、また、標準化された SUDOer 環境設定ファイルのプロビジョニングを管理するためのポリシー設定を行える。

もう1つのメリットとして、Likewise は、Kerberos 認証をサポートできるように記述されている Kerberos 対応アプリケーションがその認証インフラストラクチャを利用できるようにし、シングルサインオン (SSO) をサポートできるようにしていることが挙げられる。

本書ではここから、Microsoft Active Directory を背景にした認証を行うための Kerberos 対応アプリケーションの設定方法について説明する。

## シングルサインオンとは

「シングルサインオン」という用語は、実に記述的ではあるがよく誤用される。SSO はその文字どおり、ユーザーはユーザー名とパスワードを 1 度入力すればよく、その後、ソフトウェアがユーザーの確立済み証明書を認識し、認識後はそれを再要求しないという意である。SSO の「シングル」部分は通常、ユーザーのコンピュータに対する初回ログインを意味する。ユーザーは個々のユーザー名およびパスワードを入力して一連の証明書を認証させる。認証されたら、SSO メカニズムに認識されているソフトウェアにもそれが適用される。

「単一のユーザー名 / パスワード」では SSO といえない。つまり、複数のシステム全体において同期化された単一のユーザー名 / パスワードを設定しても、時々それを再入力する必要があるれば SSO ではないのである。同様に、パスワードキャッシュやパスワードのキーリングも SSO ではない。キーリングは本来、ユーザーに代わってパスワードを自動入力する「入力補助機能」である。つまり、SSO を認識する認証インフラストラクチャを参照しないのである。

SSO は、コンピュータとアプリケーションで構成されるネットワーク全体に実装された場合、よくエンタープライズシングルサインオンと呼ばれる。エンタープライズシングルサインオンは最も価値の高い SSO 種であり、本書ではこのタイプについて取り上げる。全社的に適用するためには「SSO」に対する参考文献について理解すべきである。

## Kerberos とは

Kerberos は、SSO を実装しやすくする **認証プロトコル** の 1 つである。このプロトコルは、1980 年代後半にマサチューセッツ工科大学 (MIT) が *Athena* プロジェクトの一環として開発したものである。もともとは *RFC 1510* に規定されていたが、最新版は *RFC 4120* である。両 RFC は <http://www.rfc-archive.org/> で参照できる。

Kerberos は、SSO 向けの標準として承認されている。そして、一般的なネットワーク攻撃 (例: *中間者攻撃* や *反射攻撃*) に耐えられるように設計されたセキュアな認証メカニズムであると考えられている。このプロトコルは、最新のあらゆるオペレーティングシステムで利用でき、多くのソフトウェア・アプリケーションでもサポートされている。Microsoft は、*Microsoft Windows™2000* の投入以来、Kerberos をサポートしている。

本書では Kerberos プロトコルについて完全に解説することはしないが、前出の文書類を参考にすべきである。また、さらに有益な情報を得たい場合、<http://www.isi.edu/~brian/security/kerberos.html> にアクセスして *The Moron's Guide To Kerberos* を参考にされるとよい。

本書においてこれから解説する内容を理解するためには、Kerberos のいくつかの基本原則を知っておく必要がある。

- Kerberos は、暗号化されたチケットを使って証明書を示す。
- 暗号化技術は、Kerberos クライアントおよび Kerberos キー配布センター (KDC) に知られている共有秘密鍵 (キー) に依存している。この秘密鍵はアカウントパスワードをベースにしている。KDC にユーザーアカウントが生成されると、Kerberos は共有秘密鍵を格納し、それを使ってチケットを暗号化し、ユーザーに代わってそのチケットをクライアントへ送信する。そしてユーザーがそのクライアントマシンへログインする際にユーザー名とパスワードを入力すると、そのクライアントの共有秘密鍵も確立されるのである。この方式によって、KDC とクライアントマシンは安全性の高い暗号化方式でやりとりを行える。
- Kerberos に対応させたいアプリケーションは、KDC に共有秘密鍵を確立するサービスアカウントへの関連付けが必要になる。このことによ

て KDC は、関連付けられたアプリケーションによって認識されるほかない方法でチケットを暗号化できる。

- ユーザーキーとアプリケーションキー (共有秘密鍵) は通常、その後の使用に備えてキーテーブル (キータブ) に保管される。これらのキータブは、Kerberos データの復号にそれを必要とするソフトウェアが利用できなければならない。ユーザーのキータブは、ユーザーがコンピュータへログインしたときに確立されるはずである。アプリケーションキータブは、より永続性が高く、サービスアカウントにおいてパスワードが変更されたときに生成されるのみである。
- ユーザーが Kerberos 対応アプリケーションへアクセスする必要がある場合、ユーザーは (間接的に、クライアントアプリケーション経由で) そのアプリケーション用のサービスチケットを KDC に要求する。このチケットの一部は当該ユーザーの共有秘密鍵を用いて暗号化され、残りの部分はアプリケーションの共有秘密鍵を用いて暗号化されている。このことによって、ユーザーのクライアントコンピュータとアプリケーションのクライアントコンピュータはどちらも、受け取るチケットの妥当性を検査できる。
- Kerberos 対応アプリケーションは、ほかの認証方式をサポートすることがよくあり、アプリケーションクライアントは実行予定の認証の種類について情報を交換する必要がある。オペレーティングシステムは通常、ソフトウェアがこのような情報交換をスムーズに行えるようにする。Windows システムの場合は SSPI、Linux/UNIX では GSSAPI が用意されている。両システムの相互運用性はほぼ確保されている。

Kerberos は SSO を実装しやすくするが、それを適切に機能させる取り組みは極めて困難かつ厄介になる可能性がある。独自のステップが多数関係するだけでなく、処理中にあちこちでエラーが発生する。このようなエラーは一般的に認証失敗によって唯一明らかになる (SSO ではない!)。エラーがプロセス内に潜行する可能性のある場所の診断は困難かもしれない。

Likewise は、Kerberos の構成や利用を自動化することによって Kerberos の簡便性を大幅に高めることができる。

## Likewise はどのように SSO をサポートするのか

Likewise を実装すると、Linux/UNIX コンピュータは、Microsoft Active Directory (AD) を用いてユーザー認証を行うことができる。

Microsoft Windows 2000 以来、AD のプライマリ認証プロトコルは Kerberos である。ドメインに加わってその制御下にある Microsoft Windows コンピュータにユーザーがログインすると、オペレーティングシステムは Kerberos プロトコルを用いてキーを確立し、ユーザーに代わりチケットを要求する。この演算中、AD は Kerberos の KDC である。

Likewise を実装すると、Linux および UNIX コンピュータは同様の方法で AD とやりとりできる。また、これらのマシンをドメインに加えることができる上、ユーザーは個々の AD 証明書を使ってこれらのマシンにログインできる。そして、ユーザーの代わりにチケットを要求し、チケットはその後、SSO をその他のアプリケーションに実装する場合に使うことができる。

こうした目的を達成するために、Likewise は Linux と UNIX の構成が Kerberos 認証に適していることを確認する必要がある。その間接的なメリットの 1 つとして、Kerberos の構成がほかのアプリケーションによる利用に適しているかを Likewise が確認できることが挙げられる。

Kerberos 認証インフラストラクチャが適正に構成されていて「健全」と確認するための Likewise の取り組みについて、以下に簡単にまとめる。

- AD に関連付けられた名前を解析するために DNS が正しく構成されていることの確認
- Linux/UNIX コンピュータを AD に加えるためのツールの提供
- AD に統合された DNS サーバを用いて Linux/UNIX コンピュータ名を解析できるようにするためのセキュアかつ動的な DNS アップデートの実行
- Kerberos の構成。複数の KDC で構成された環境において、Kerberos が適切なサーバを選択することを確認。
- Kerberos 経由で SSO をサポートするために SSHD を構成 (GSSAPI を利用)

- コンピュータ用のキータブを生成 (AD へ加わったとき)、および、ユーザー用のキータブを生成 (ログオン中)
- アプリケーション用のキータブを生成しやすくするツールを提供

Likewise は、Java アプリケーションおよびアプリケーションサーバ (例: Tomcat, JBoss, IBM WebSphere) が Kerberos 経由で SSO を実装できるようにするためのソフトウェアも提供している。

Linux/UNIX システムを手動構成することによって以上の取り組みの一部を行える可能性もあるが、Likewise ソリューションの実装によってそのプロセスを大幅に簡素化できる。Likewise で瞬時にできることを成し遂げるためにシステムをいじくりまわして時間を浪費することは非効率的であることはいうまでもない。

### 例: OpenSSH

Likewise は、Kerberos 経由で SSO をサポートするために OpenSSH の自動構成を行えるが、どのような形で Likewise がそれを可能にしているのかについて検討してみたい。これ以外のアプリケーションでも同様の構成が必要になる可能性があり、また、このプロセスを理解すると、この技法をほかの事例に適用しやすくなるであろう。

**注:** OpenSSH の一部のバージョンは Kerberos をサポートしていない。4.2p1 未満のバージョンの場合には動作しないか、エラーになる可能性がある。

#### SSH のサービスプリンシパル名

最初に検討する必要がある項目は Kerberos のサービスプリンシパル名 (SPN) である。これは ssh と sshd によって用いられる。SPN は、認証チケットの発行先となるサービスを識別する文字列である。SSH の場合、SPN は以下のような形式である。

```
host/<server name>@<REALMNAME>
```

例えば、ユーザーが ssh を使って fozzie.mycorp.com という名のコンピュータへアクセスする場合、ssh プログラムは SPN 用のサービスチケットを要求する。

```
host/fozzie.mycorp.com@MYCORP.COM
```

Kerberos のレルム名は、コンピュータのドメイン名を大文字表記にしたものであることに留意されたい。

#### システムのキータブ生成

Microsoft Active Directory がこの SPN 用の Kerberos チケットを生成するためには、適切なサービスアカウントが存在しなければならない。さらに、1つのキータブを当該サービスアカウント向けに作成し、それを sshd サーバ上に配置する必要もある。

Likewise を実装しない場合、それを実現するためには以下のようなさまざまなステップを手動で行う必要があるであろう。

1. AD にサービスアカウントを作成

2. SSH SPN とサービスアカウントの関連付けおよびそれに適したキータブの生成を行うために Windows の ktpass ユーティリティを実行
3. Windows で生成されたキータブをまた sshd サーバへコピー

Likewise ではこの作業が完全に自動化される。Linux/UNIX コンピュータが AD に加わると、そのコンピュータ向けのマシンアカウントが生成される。そのコンピュータが `fozzie` という名前であれば、`fozzie$` と呼ばれるマシンアカウントが AD に生成される。Likewise はそして当該 SPN 用のキータブを自動生成し、それを標準的なシステムディレクトリに格納する (通常、`/etc/krb5.keytab`)。Likewise は「`lwinet`」というツールを備えていることに留意する。`lwinet` は、ほかのサービス向けのキータブエントリを追加生成する際に利用できる。

### ユーザーのキータブ生成

ユーザーが `ssh` プログラムを実行し、OpenSSH によって Kerberos 認証の利用が決定された場合、そのユーザー用のキータブにアクセスして、接続したいサービス/コンピュータ向けのサービスチケットを入手できるようにすることが必要になる。このキータブは当該ユーザーのアカウント名およびパスワードを用いて生成しなければならない。手動の場合、Linux/UNIX の `kinit` ユーティリティを用いて行える。しかしながら Likewise の場合、ユーザーがコンピュータにログインしたときに自動で生成できる。たいていのシステムにおいて、ユーザーのキータブは `/tmp` ディレクトリに格納されて `krb5cc_<UID>` という名が付けられる。`<UID>` の部分には、システムによってアサインされた数値のユーザー ID が入る。

### OpenSSH の構成

OpenSSH はクライアントとサーバコンピュータの双方において構成する必要がある。クライアント側では、(通常、`/etc/ssh/ssh_config` に格納されている) `ssh_config` ファイルを変更しなければならない。サーバ側では、(通常、`/etc/ssh/sshd_config` に格納されている) `sshd_config` ファイルを変更する。

サーバの場合、以下が `sshd_config` に表示されているはずである。

```
GSSAPIAuthentication yes
GSSAPICleanupCredentials yes
```

また、クライアントには `ssh_config` に以下が表示される。

```
GSSAPIAuthentication yes  
GSSAPIDelegateCredentials yes
```

これらが表示されていない場合は、Likewise は適切なファイルにそれらを追加する。

### SSO のテスト

OpenSSH が適切に構成されていれば、SSO サポートの実証は簡単である。Active Directory の証明書を用いて (Likewise を実行中の) Linux/UNIX マシンにログインし、ssh を利用して (Likewise を実行中の) ほかのマシンへ接続する。OpenSSH は、ユーザー名やパスワードの入力を求めることなく接続を確立するはずである。

## 例: Apache Tomcat

SSO をサポートするために Microsoft Active Directory を用いて Tomcat を構成する場合、構成ステップは OpenSSH の場合と一部は同じである。しかし、ステップの数は増える。大まかには、以下のようなステップで構成される。

1. AD にサービスアカウントを作成
2. AD 上の SPN とサービスアカウントの関連付けおよび SPN に適したキータブの生成
3. Linux/UNIX ファイルシステムの適切なディレクトリにキータブを格納
4. Likewise Java 認証モジュール (valve クラス) を Tomcat に追加
5. 生成されたキータブから Kerberos キーを取得するためにその認証モジュールを構成
6. AD グループのメンバー資格の検査によって Java のロールを決定するために認証モジュールを構成
7. 特定ロールの AD 認証済みユーザーに対するアクセスを制限するためにアプリケーションを構成
8. Microsoft Internet Explorer や Mozilla Firefox を実行させ、制限のかけられたウェブサイトへ Windows クライアントからアクセスすることによって Tomcat SSO をテスト。Firefox を使って Linux/UNIX 上でも同様にテスト。

### まとめ

エンタープライズシングルサインオンによってユーザーの簡便性は大幅に高まる。また、単一のユーザー名とパスワードを記憶させられるだけでなく、SSOの導入は、こうした証明書の入力が入力が1回で済むということを意味する。

Kerberos は、エンタープライズ SSO に適した申し分のないインフラストラクチャである。Kerberos 対応の認証製品である Microsoft Active Directory が優勢であるということは、SSO が当然普及することを意味する。しかし不幸なことに、Kerberos 経由でユーザーを正しく認証するために Linux/UNIX コンピュータを構成することは困難である上にエラーが発生しやすい。

Likewise を実装すると、Linux/UNIX コンピュータは、Microsoft Active Directory 経由でユーザーの認証を行えるようになる。また、AD と正しくやりとりできるようにするために Linux/UNIX コンピュータ上の Kerberos インフラストラクチャを構成する。この製品によって、Kerberos 対応アプリケーションが SSO をサポートできるようにする上で必要な作業が簡素化される。Likewise は、AD での認証と、SSH を用いた場合の SSO サポートを行うためにシステムロゲインを自動構成する。

Likewise Software も、その他のアプリケーションによる SSO 実装をスムーズに進めるソフトウェアコンポーネントを提供している。