

Single Sign-On for Kerberized Linux and UNIX Applications

AUTHOR:

Manny Vellon
Chief Technology Officer
Likewise Software

Abstract

This document describes how Likewise facilitates the implementation of enterprise single sign-on (SSO). It explains how Kerberos-aware applications can be configured to exploit the authentication infrastructure provided by Likewise. It explains the concepts as well as outlining the specific steps that must be taken to enable single sign-on support in applications.

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

Table of Contents

Introduction	4
What Is Single Sign-On?	5
What Is Kerberos?	6
How Likewise Supports SSO	8
Example: Open SSH	10
The SSH Service Principal Name.....	10
System Keytab Generation.....	10
User Keytab Generation.....	11
Configuring OpenSSH.....	11
Testing SSO.....	12
Example: Apache Tomcat	13
Summary	14
For More Information	15

Introduction

Likewise allows Linux and UNIX computers to authenticate and authorize users through Microsoft Active Directory™. This provides many benefits:

- A single username and password for users, regardless of whether they are using Microsoft Windows™ or non-Windows systems.
- Greatly simplified user account management. System administrators can provision users, maintain passwords and deprovision users using a single identity management system.
- Improved security. Likewise extends numerous Active Directory account policies to Linux and UNIX systems. Administrators can configure minimum password lengths, password complexity requirements, password expiration policies and other settings that are applied to both Windows and non-Windows systems.
- Granular authorization supporting separation of duties. Likewise extends Active Directory Group Policy features to Linux and UNIX and provides policy settings to control the provisioning of standardized SUDOer configuration files.

An additional benefit is that Likewise allows *Kerberized* applications, applications that have been written with support for Kerberos authentication, to exploit its authentication infrastructure and to provide *single sign-on* (SSO) support.

The rest of this document describes how to configure Kerberized applications to authenticate against Microsoft Active Directory.

What Is Single Sign-On?

In spite of its perfectly descriptive nature, "single sign-on" is a frequently misused term. SSO means, literally, that you only need to provide a username and password once and that, subsequently, software will recognize your established *credentials* and not prompt for them again. The "single" part of SSO is typically satisfied when a user first logs in to a computer. The user types his/her username and password and establishes a set of credentials that are then respected by software that is aware of the SSO mechanism.

"Single username/password" is not SSO. Having a single username name and password that is synchronized across multiple systems but have to be, occasionally, re-entered is *not* SSO. Similarly, password caches or password *key rings* are also not examples of SSO. Key rings are essentially "typing aids" that automatically enter passwords for you; they do not reflect an authentication infrastructure that understands SSO.

When SSO is provided across an entire network of computers and applications, it is often referred to as *enterprise single sign-on*. This is the most valuable form of SSO and the one that this document explores. References to "SSO" should be understood to apply across the enterprise.

What Is Kerberos?

Kerberos is an *authentication protocol* that facilitates the implementation of SSO. It was developed at MIT in the late 1980's as part of the *Athena* project. It was originally described in *RFC 1510* but its modern incarnation is described by *RFC 4120*. Both of these documents can be viewed at <http://www.rfc-archive.org/>.

Kerberos is an accepted standard for SSO. It is considered a secure authentication mechanism, having been designed to withstand common network attacks (for example, *man-in-the-middle* and *replay* attacks). It is available on all modern operating systems and supported by many software applications. Microsoft has supported Kerberos since the advent of *Microsoft Windows™ 2000*.

A full description of the Kerberos protocol is beyond the scope of this document. Interested readers should consult the documents listed above or, for a more readable treatment, *The Moron's Guide To Kerberos* available at <http://www.isi.edu/~brian/security/kerberos.html>.

In order to understand the rest of this document, however, it is necessary to understand a few basic tenets of Kerberos:

- Kerberos uses encrypted tickets to represent credentials.
- The encryption technique relies on a shared secret (a key) known to the Kerberos client and the Kerberos key distribution center (KDC). This secret is based on an account password. When a user account is created at the KDC, Kerberos stores the shared secret and uses it to encrypt tickets sent to a client on behalf of the user. When the user logs into the client machine, he/she provides a username and password, establishing the shared secret on the client as well. This allows the KDC and client machines to communicate in a safe, encrypted, fashion.
- Applications that want to use Kerberos will need to be associated with service accounts that establish shared secrets on the KDC. This allows the KDC to encrypt tickets in a form that can only be understood by relevant applications.

- User and applications keys (shared secrets) are usually stored in a key table (keytab) for subsequent use. These keytabs need to be available to software that needs them to decrypt Kerberos data. A user's keytab should be established when the user logs into to a computer. Application keytabs are longer-lived and are only created when passwords are changed on service accounts.
- When a user needs to access a Kerberized application, he/she (indirectly, via a client application) asks the KDC for a service ticket for that application. A portion of this ticket is encrypted with the user's shared secret and another portion is encrypted with the application's shared secret. This allows both the user's client computer and the application's client computer to verify that the incoming ticket is valid.
- Kerberized applications frequently support other forms of authentication and make it necessary for application clients to negotiate what type of authentication they're going to perform. Operating systems typically provide software to facilitate this negotiation. Windows systems provide SSPI whereas Linux/UNIX provide GSSAPI. These two systems are, mostly, interoperable.

Although Kerberos facilitates the implementation of SSO, it can be extremely difficult and frustrating to get it to work properly. Many individual steps are involved and mistakes anywhere along the way typical only manifest themselves by a failure of authentication (SSO doesn't SO!). It can be difficult to diagnose where errors might have crept into the process.

Likewise greatly facilitates the use of Kerberos by automating its configuration and use.

How Likewise Supports SSO

Likewise allows Linux and UNIX computers to authenticate users with Microsoft Active Directory (AD).

Since Microsoft Windows 2000, AD's primary authentication protocol has been Kerberos. When a user logs into a Microsoft Windows computer that is joined to a domain, under the covers, the operating system is using the Kerberos protocol to establish a key and to request a ticket for the user. During this operation, AD is the Kerberos KDC.

Likewise allows Linux and UNIX computers to interact with AD in a similar manner. It allows these machines to be joined to a domain and it allows users to log in to these machines using their AD credentials. It requests a ticket for the user that can be subsequently used to implement SSO with other applications.

To accomplish these goals, Likewise must assure that Linux and UNIX are properly configured for Kerberos authentication. A side benefit of this is that Likewise assures that Kerberos is properly configured for use by other applications.

Here's a brief list of things that Likewise does to ensure that the Kerberos authentication infrastructure is properly configured and "healthy":

- Assures that DNS is properly configured to resolve names associated with AD.
- Provides tools to join Linux/UNIX computers to AD.
- Performs secure *dynamic* DNS updates to assure that Linux/UNIX computer names can be resolved with AD-integrated DNS servers.
- Configures Kerberos. In an environment with multiple KDCs, assures that Kerberos will select the appropriate server.
- Configures SSHD to support SSO through Kerberos (by using GSSAPI).

- Creates a keytab for the computer (when it is joined to AD) and one for the user (during logon).
- Provides tools to facilitate the generation of keytabs for applications.

Likewise also provides software that allows Java applications and application servers (for example, Tomcat, JBoss, IBM WebSphere, etc.) to implement SSO through Kerberos.

Although it might be possible to perform some of these operations by manually configuring a Linux/UNIX system, the Likewise solution greatly simplifies the process. It is easy to spend hours tinkering with a system to accomplish what Likewise does in an instant.

Example: Open SSH

Although Likewise automatically configures OpenSSH to support SSO through Kerberos, it is worthwhile to review what it is that Likewise does to enable this. Other applications might need similar configuration and understanding the process will make it easier to apply the technique to other examples.

Note: Not all versions of OpenSSH support Kerberos. Versions older than 4.2p1 may not work or may work improperly.

The SSH Service Principal Name

The first thing that needs to be considered is the Kerberos service principal name (SPN) that is used by ssh and sshd. The SPN is a string that identifies the service for which an authentication ticket is to be generated. In the case of SSH, the SPN has the form:

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

For example, when a user uses ssh to connect to a computer named fozzie.mycorp.com, the ssh program will request a service ticket for the SPN:

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

Note that the Kerberos realm name is the computer's domain name using capital letters.

System Keytab Generation

In order for Microsoft Active Directory to generate a Kerberos ticket for this SPN, a service account must exist for it. Additionally, a keytab must be created for the service account and placed on the sshd server.

Without Likewise, it would be necessary to perform various manual steps to accomplish this:

1. Create a service account in AD.

2. Run the `ktpass` utility in Windows to associate the SSH SPN with the service account and to generate a keytab for it.
3. Copy the keytab from Windows back to the `sshd` server.

Likewise completely automates this operation. When a Linux/UNIX computer is joined to AD, a machine account is created for the computer. If the computer is called `fizzie`, a machine account called `fizzie$` is created in AD. Likewise then automatically creates a keytab for the SPN and places it in the standard system location (typically, `/etc/krb5.keytab`). Note that Likewise includes a tool, `lwinet`, that can be used generate additional keytab entries for other services.

User Keytab Generation

When the user runs the `ssh` program and OpenSSH determines that it will use Kerberos authentication, it will need to access a keytab for the user so that it can obtain a service ticket for the service/computer to which it is trying to connect. This keytab must be created using the user's account name and password. Manually, this can be performed by using the Linux/UNIX `kinit` utility. Likewise, however, does it automatically when the user logs into the computer. On most systems, the user keytab is placed in the `/tmp` directory and named `krb5cc_<UID>` where `<UID>` is the numeric user id as assigned by the system.

Configuring OpenSSH

OpenSSH must be configured at both the client and server computer. On the client, the `ssh_config` file (typically, in `/etc/ssh/ssh_config`) must be modified. On the server, `sshd_config` (typically, in `/etc/ssh/sshd_config`) must be modified.

In the server, the following lines must be present in `sshd_config`:

```
GSSAPIAuthentication yes
GSSAPICleanupCredentials yes
```

On the client, these lines must be present in `ssh_config`:

```
GSSAPIAuthentication yes
GSSAPIDelegateCredentials yes
```

Likewise will add these lines to the appropriate files, if not already present.

Testing SSO

With OpenSSH properly configured, demonstrating SSO support is simple. Log in on a Linux/UNIX machine (that is running Likewise) using Active Directory credentials and then use ssh to connect to another machine (also running Likewise). OpenSSH should establish a connection without prompting for a username or password.

Example: Apache Tomcat

Configuring Tomcat to support SSO with Microsoft Active Directory shares some of the same steps as configuring OpenSSH, but is much more involved. Roughly, the steps consist of:

1. Creating a service account in AD.
2. Associating an SPN with the service account on AD and creating a keytab for the SPN.
3. Placing the keytab in the appropriate location in the Linux/UNIX file system.
4. Adding the Likewise Java authentication module (a valve class) to Tomcat.
5. Configuring the authentication module to get its Kerberos key from the generated keytab.
6. Configuring the authentication module to determine Java roles by examining AD group membership.
7. Configuring an application to restrict access to AD authenticated users in certain roles.
8. Testing Tomcat SSO by accessing restricted web sites from a Windows client running Microsoft Internet Explorer or Mozilla Firefox. Repeating this step on Linux/UNIX using Firefox.

Summary

Enterprise single sign-on provides great user convenience. In addition to only having to remember a single username and password, SSO also implies that these credentials only need to be entered *once*.

Kerberos provides a good infrastructure for enterprise SSO. The preponderance of Microsoft Active Directory, a Kerberos-aware authentication product, means that SSO should be broadly available. Unfortunately, configuring Linux and UNIX computers to properly authenticate users via Kerberos is difficult and error-prone.

Likewise allows Linux and UNIX computers to authenticate users with Microsoft Active Directory. It also configures the Kerberos infrastructure in Linux and UNIX computers to communicate properly with AD. This simplifies the work that must be done to enable Kerberized applications to support SSO. Likewise automatically configures system login to authenticate with AD and to support SSO when using SSH.

Likewise Software also provides software components to facilitate the implementation of SSO by other applications.

For More Information

For more information on Likewise or to download a free 30-day trial version, visit the Web site at <http://www.likewisoftware.com>.

For general questions, call (800) 378-1330 or e-mail info@likewisoftware.com.

For technical questions or support for the 30-day free trial, e-mail support@likewisoftware.com.

ABOUT LIKEWISE

Likewise® Software solutions improve management and interoperability of Windows, Linux, and UNIX systems with easy to use software for Linux administration and cross-platform identity management.

Likewise provides familiar Windows-based tools for system administrators to seamlessly integrate Linux and UNIX systems with Microsoft Active Directory. This enables companies running mixed networks to utilize existing Windows skills and resources, maximize the value of their Active Directory investment, strengthen the security of their network and lower the total cost of ownership of Linux servers.

Likewise Software is a Bellevue, WA-based software company funded by leading venture capital firms Ignition Partners, Intel Capital, and Trinity Ventures. Likewise has experienced management and engineering teams in place and is led by senior executives from leading technology companies such as Microsoft, F5 Networks, EMC and Mercury.