IBM Software Group | Lotus Expeditor 6.1.1 Education

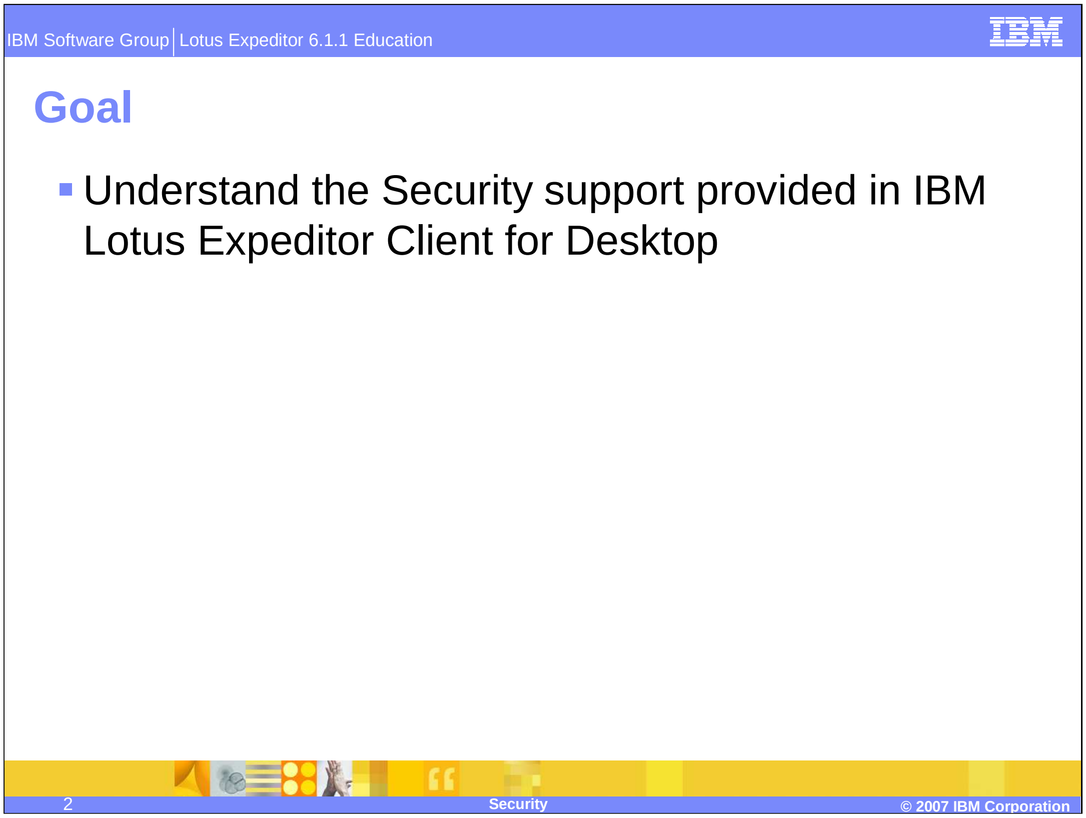
IBM Lotus® Expeditor Client for Desktop

Security

Lotus software

business on demand software

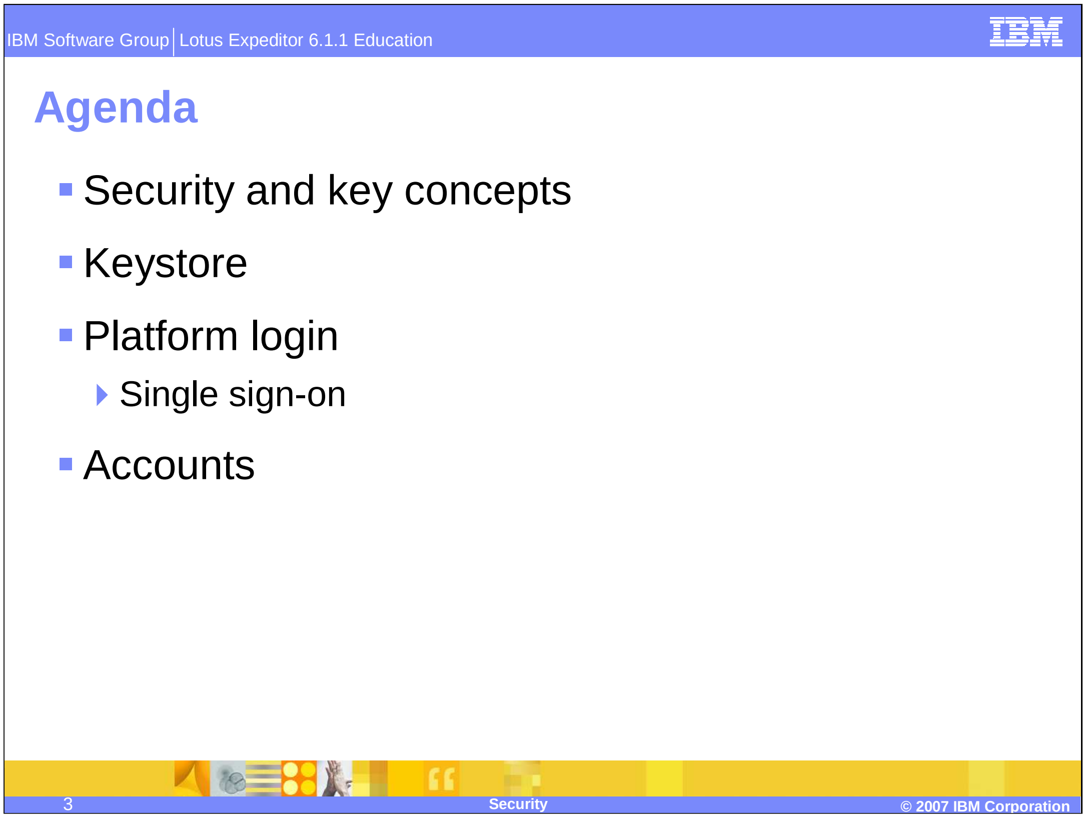
This presentation explains the Security support in IBM Lotus Expeditor Client for Desktop.



Goal

- Understand the Security support provided in IBM Lotus Expeditor Client for Desktop

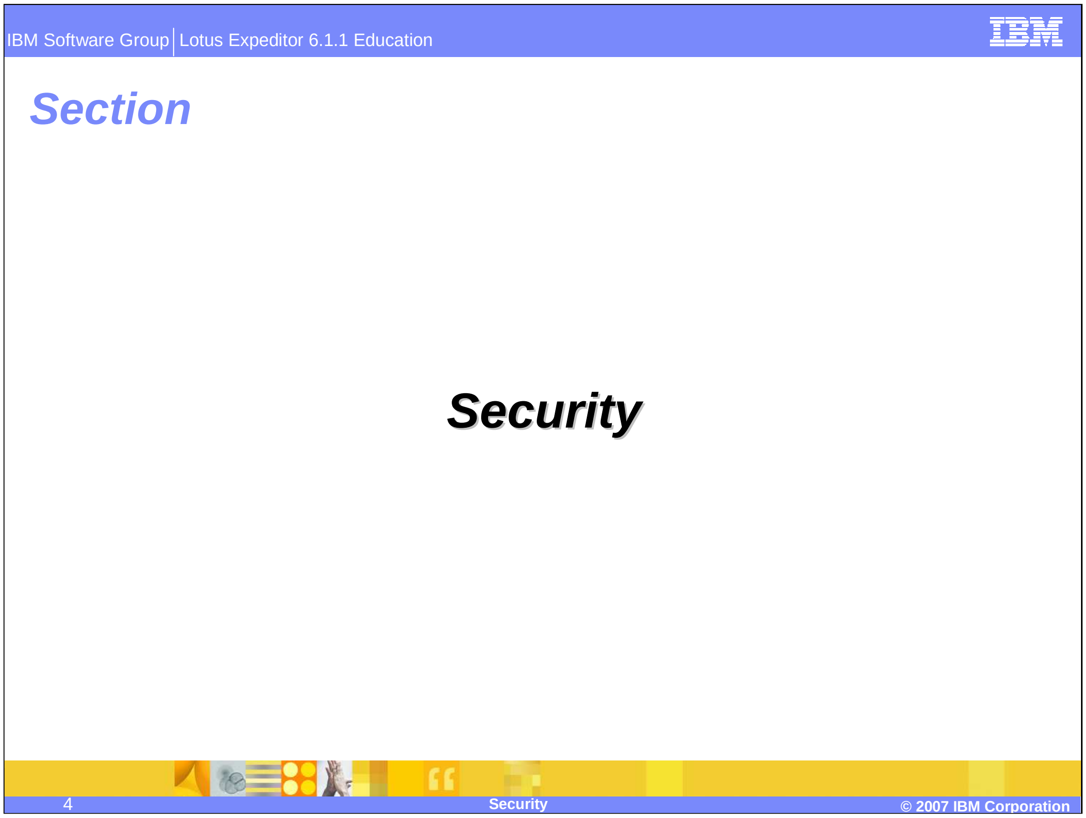
The goal of this presentation is to understand the Security support provided in IBM Lotus Expeditor Client for Desktop.



Agenda

- Security and key concepts
- Keystore
- Platform login
  - Single sign-on
- Accounts

The agenda of this presentation is to explain the security capabilities the client platform provides to you, the infrastructure and plug-ins that enable these capabilities, and details about the security components supported by IBM Lotus Expeditor Client for Desktop.



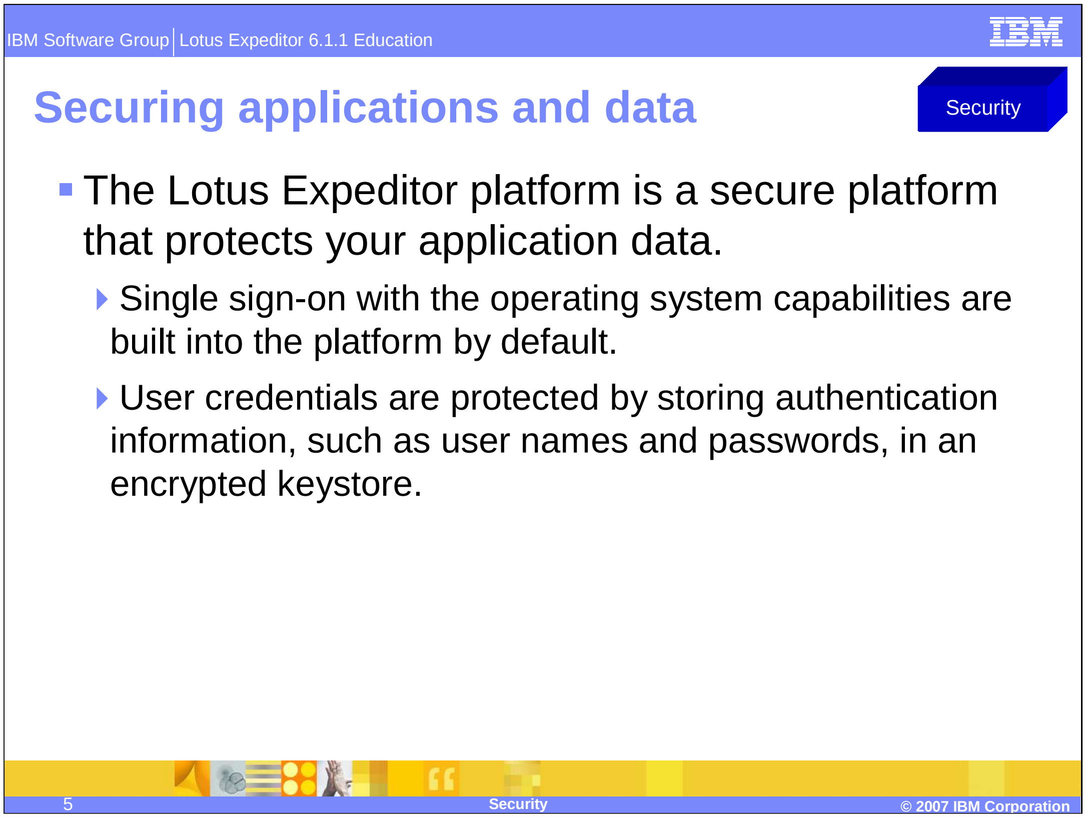
IBM Software Group | Lotus Expeditor 6.1.1 Education

Section

Security

4 Security © 2007 IBM Corporation

For the success of the Expeditor platform, it is very critical that it provides the default authentication mechanisms that can be extended by third parties to implement their own means of authentication, if required. The Expeditor platform provides multiple JAAS login modules that can be utilized by the Accounts API to meet the authentication needs of the consumer. The Expeditor platform provides login modules for HTTP Basic Authentication, J2EE Form-based Authentication, and Key store password reader and writer.



IBM Software Group | Lotus Expeditor 6.1.1 Education

IBM

Securing applications and data

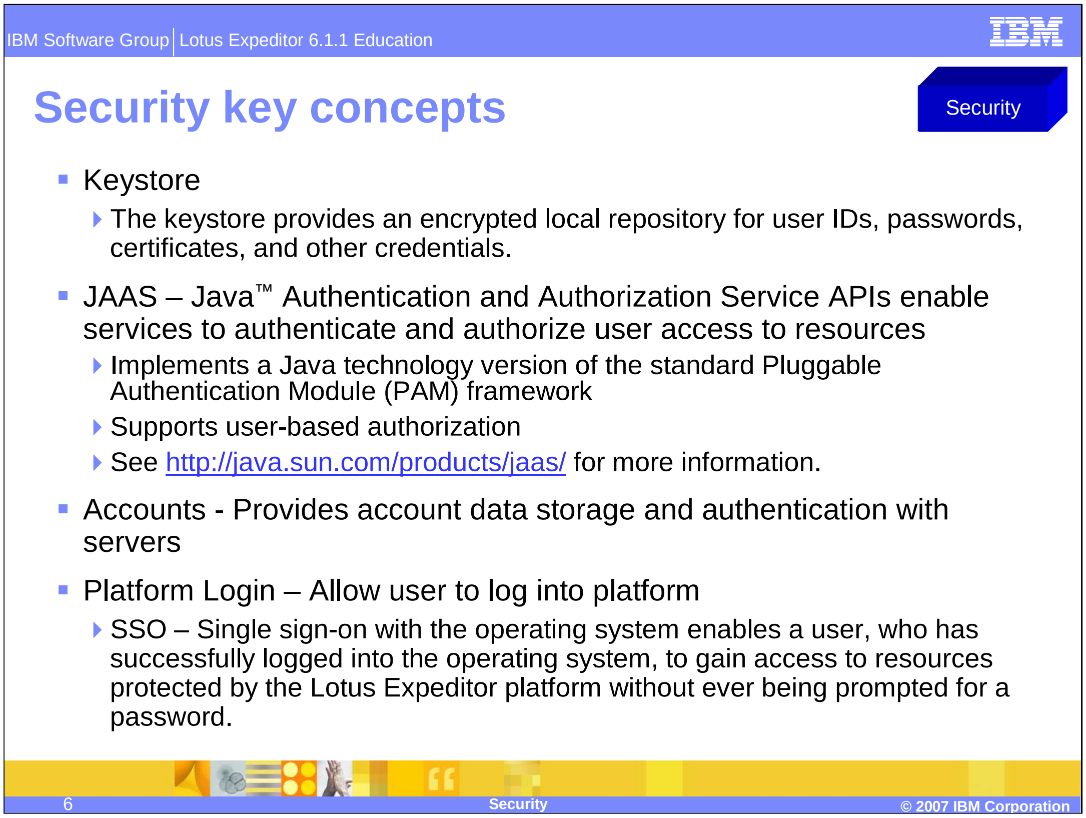
Security

- The Lotus Expeditor platform is a secure platform that protects your application data.
  - Single sign-on with the operating system capabilities are built into the platform by default.
  - User credentials are protected by storing authentication information, such as user names and passwords, in an encrypted keystore.

5 Security © 2007 IBM Corporation

The Lotus Expeditor platform is a secure platform that protects your application data. Single sign-on with the operating system capabilities are built into the platform by default.

It secures the applications and application data running on the client by protecting user credentials by storing authentication information, such as user names and passwords, in an encrypted key store.



Security key concepts

- Keystore
  - The keystore provides an encrypted local repository for user IDs, passwords, certificates, and other credentials.
- JAAS – Java™ Authentication and Authorization Service APIs enable services to authenticate and authorize user access to resources
  - Implements a Java technology version of the standard Pluggable Authentication Module (PAM) framework
  - Supports user-based authorization
  - See http://java.sun.com/products/jaas/ for more information.
- Accounts - Provides account data storage and authentication with servers
- Platform Login – Allow user to log into platform
  - SSO – Single sign-on with the operating system enables a user, who has successfully logged into the operating system, to gain access to resources protected by the Lotus Expeditor platform without ever being prompted for a password.

This slide describes the key concepts for Security:

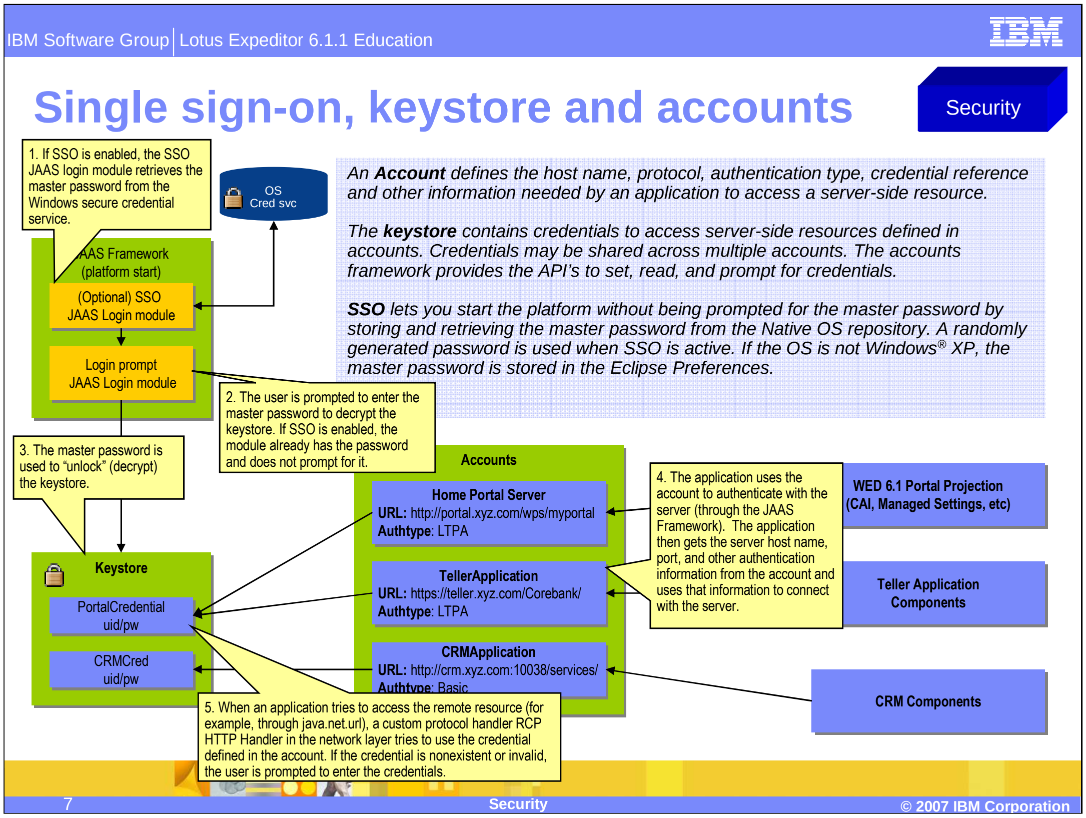
The key store provides an encrypted local repository for user IDs, passwords, certificates, and other credentials.

Java Authentication and Authorization Service (or JAAS) are APIs that enable services to authenticate and authorize user access to resources. JAAS implements a Java technology version of the standard Pluggable Authentication Module (PAM) framework. It supports user-based authorization. See http://java.sun.com/products/jaas/ for more information.

The Platform Login allows users to log into the platform.

Single sign-on with the operating system enables a user who has successfully logged into the operating system to gain access to resources protected by the Lotus Expeditor platform without ever being prompted for a password.

Accounts provide account data storage and authentication with servers.



Single sign-on, keystore and accounts

Security

*An **Account** defines the host name, protocol, authentication type, credential reference and other information needed by an application to access a server-side resource.*

*The **keystore** contains credentials to access server-side resources defined in accounts. Credentials may be shared across multiple accounts. The accounts framework provides the API's to set, read, and prompt for credentials.*

***SSO** lets you start the platform without being prompted for the master password by storing and retrieving the master password from the Native OS repository. A randomly generated password is used when SSO is active. If the OS is not Windows® XP, the master password is stored in the Eclipse Preferences.*

1. If SSO is enabled, the SSO JAAS login module retrieves the master password from the Windows secure credential service.

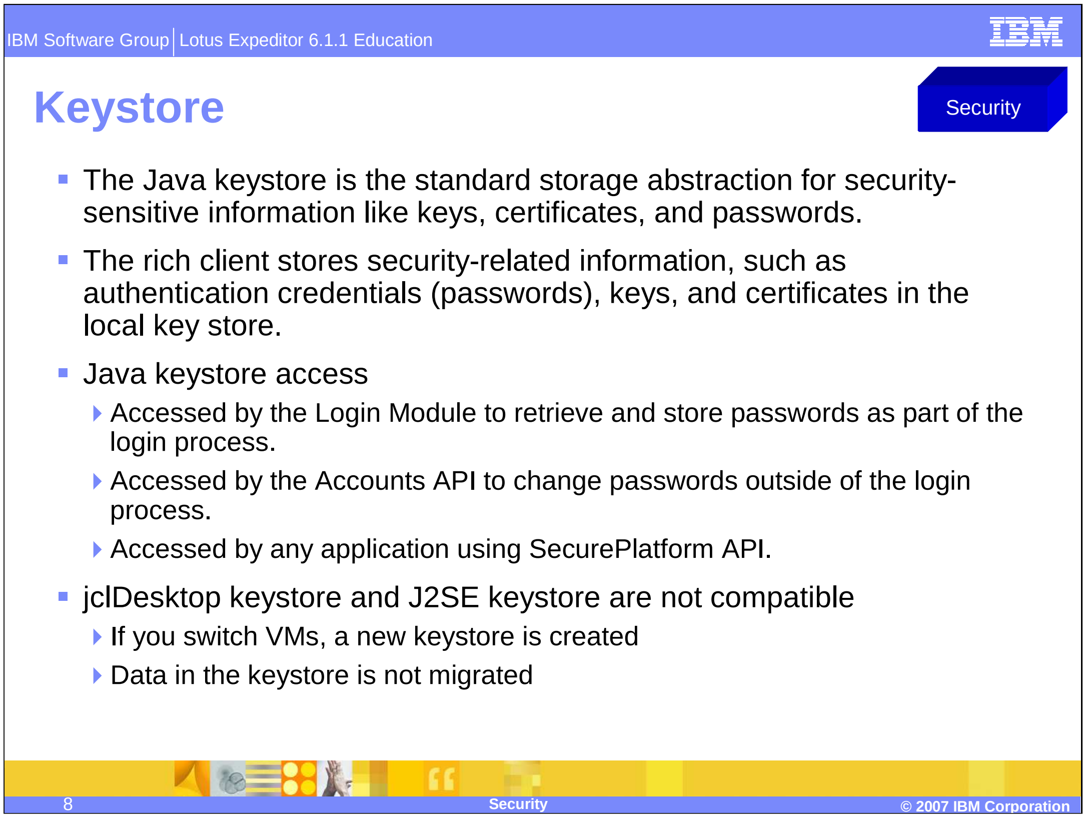
2. The user is prompted to enter the master password to decrypt the keystore. If SSO is enabled, the module already has the password and does not prompt for it.

3. The master password is used to "unlock" (decrypt) the keystore.

4. The application uses the account to authenticate with the server (through the JAAS Framework). The application then gets the server host name, port, and other authentication information from the account and uses that information to connect with the server.

5. When an application tries to access the remote resource (for example, through java.net.url), a custom protocol handler RCP HTTP Handler in the network layer tries to use the credential defined in the account. If the credential is nonexistent or invalid, the user is prompted to enter the credentials.

This slide shows the steps for logging into the client, using the Accounts and keystore to store credentials, And using the account data to connect to and authenticate to a server or service.



IBM Software Group | Lotus Expeditor 6.1.1 Education

IBM

Keystore

Security

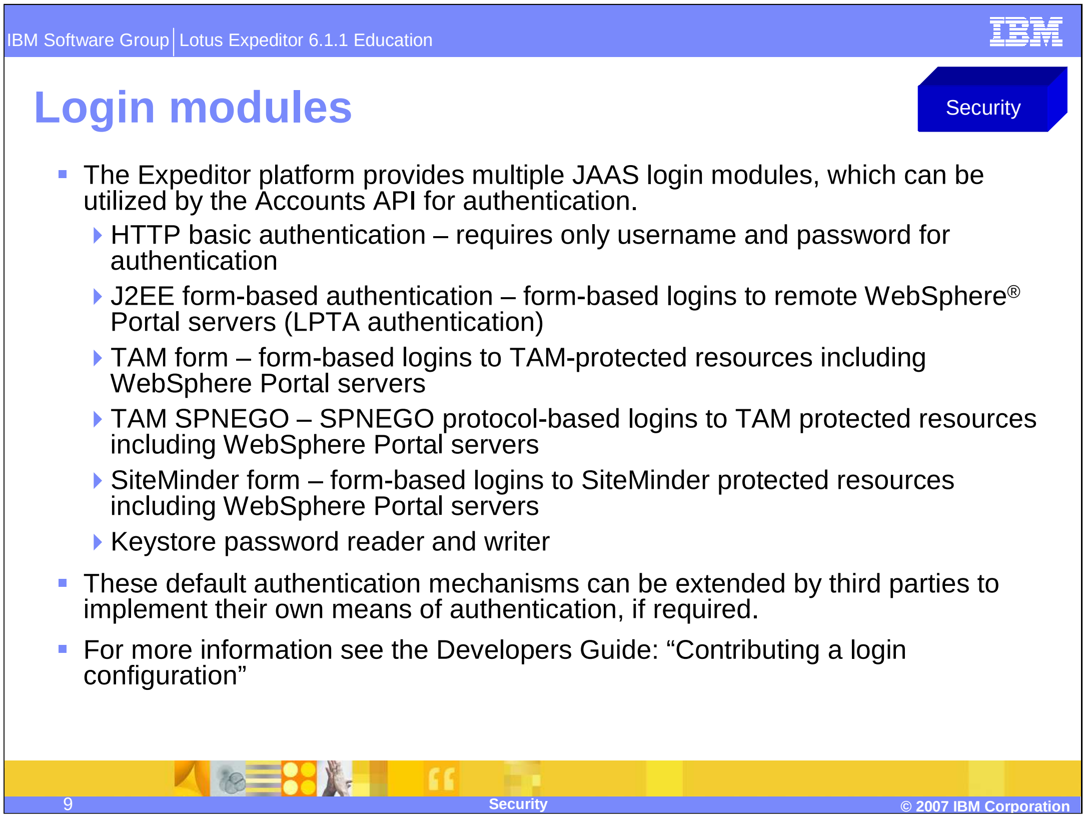
- The Java keystore is the standard storage abstraction for security-sensitive information like keys, certificates, and passwords.
- The rich client stores security-related information, such as authentication credentials (passwords), keys, and certificates in the local key store.
- Java keystore access
  - Accessed by the Login Module to retrieve and store passwords as part of the login process.
  - Accessed by the Accounts API to change passwords outside of the login process.
  - Accessed by any application using SecurePlatform API.
- jclDesktop keystore and J2SE keystore are not compatible
  - If you switch VMs, a new keystore is created
  - Data in the keystore is not migrated

8 Security © 2007 IBM Corporation

The Java keystore is the standard storage abstraction for security-sensitive information such as keys, certificates and passwords. The Java keystore used is JCEKS on J2SE and jks on jclDesktop. The Java keystore is accessed by the Login Modules to retrieve and store passwords as part of the login process. The Java keystore is also accessed by the Accounts API to change passwords outside of the login process.

The Java keystore must be unlocked to access any data stored inside of it. The keystore can be unlocked using Platform Login. By default, Lotus Expeditor calls the Platform Login to unlock the keystore at startup time. This can be changed programmatically to unlock the keystore when needed.

The jclDesktop keystore and J2SE keystore are not compatible. If you switch JVMs, a new keystore is created and data in the keystore is not migrated; you must re-create data.



IBM Software Group | Lotus Expeditor 6.1.1 Education

IBM

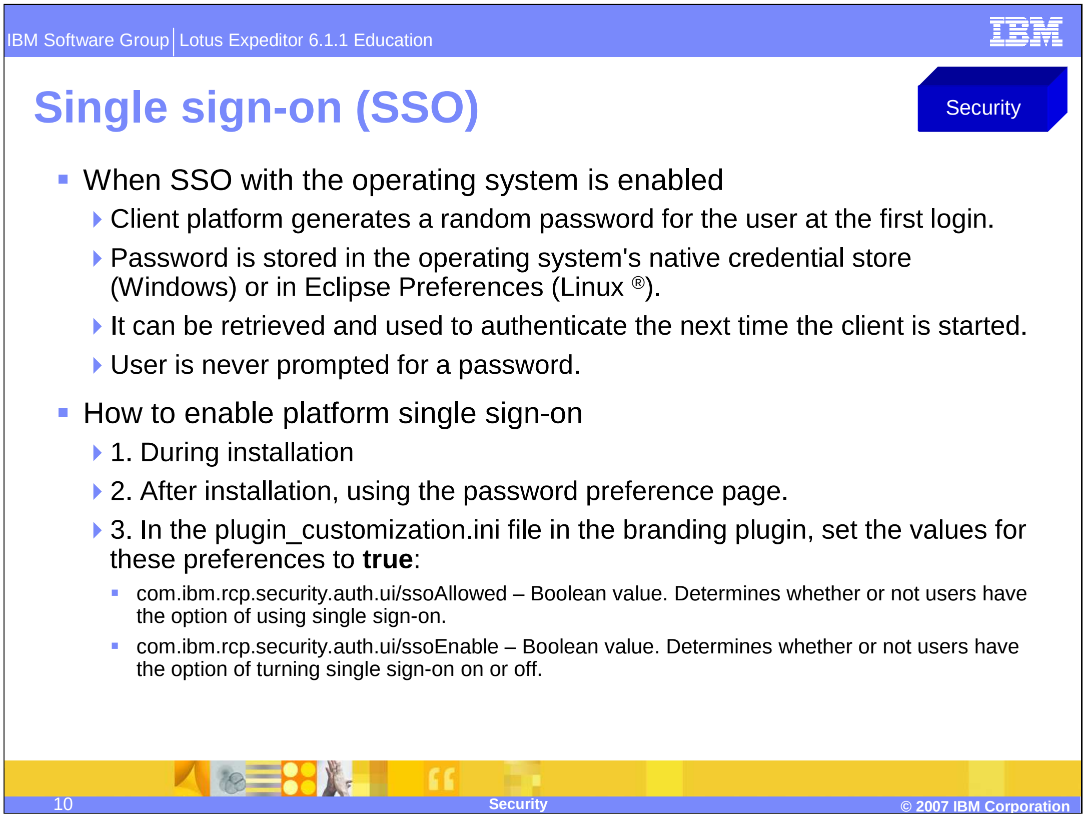
Login modules

Security

- The Expeditor platform provides multiple JAAS login modules, which can be utilized by the Accounts API for authentication.
  - HTTP basic authentication – requires only username and password for authentication
  - J2EE form-based authentication – form-based logins to remote WebSphere® Portal servers (LPTA authentication)
  - TAM form – form-based logins to TAM-protected resources including WebSphere Portal servers
  - TAM SPNEGO – SPNEGO protocol-based logins to TAM protected resources including WebSphere Portal servers
  - SiteMinder form – form-based logins to SiteMinder protected resources including WebSphere Portal servers
  - Keystore password reader and writer
- These default authentication mechanisms can be extended by third parties to implement their own means of authentication, if required.
- For more information see the Developers Guide: “Contributing a login configuration”

9 Security © 2007 IBM Corporation

For the success of the Expeditor platform, it is critical that it provides the default authentication mechanisms, which can be extended by third parties to implement their own means of authentication, if required. The Expeditor platform provides multiple JAAS login modules, which can be utilized by the Accounts API to meet the authentication needs of the consumer. The Expeditor platform provides login modules for HTTP Basic Authentication; J2EE Form based Authentication, TAM Form based Authentication, TAM SPNEGO Authentication and Site Minder Form Authentication, as well as, Key store password reader and writer.

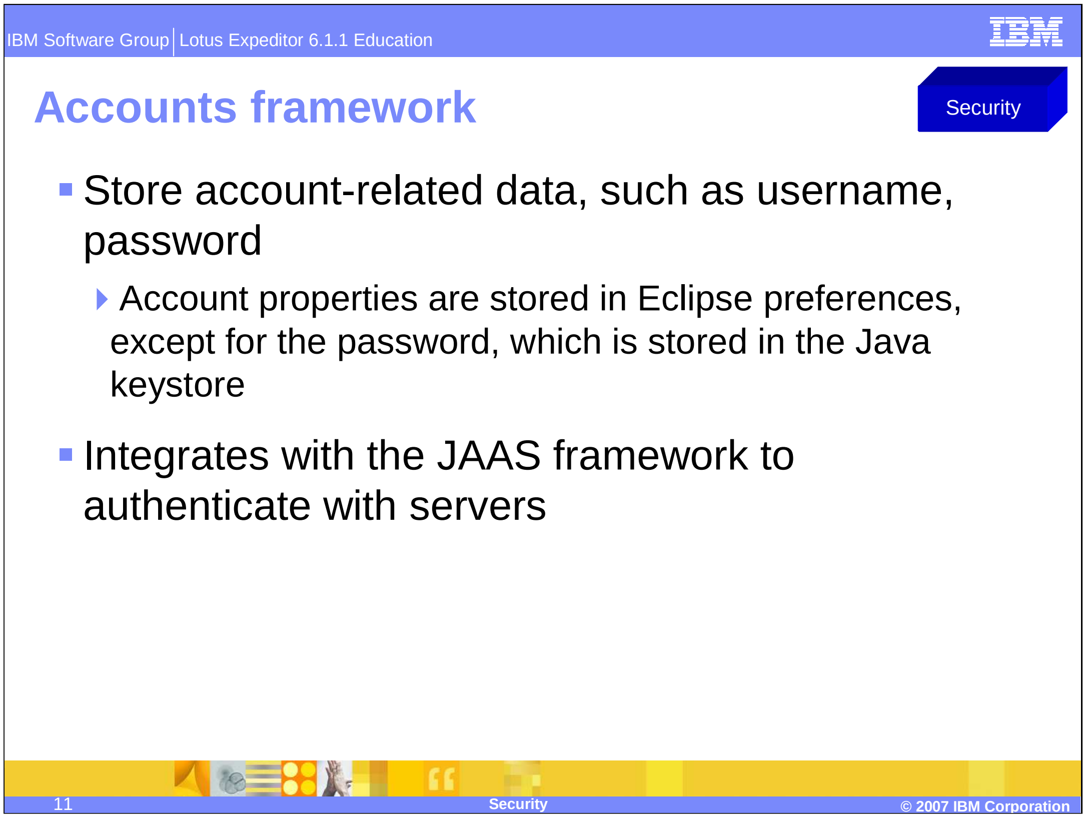


Single sign-on (SSO)

Security

- When SSO with the operating system is enabled
  - Client platform generates a random password for the user at the first login.
  - Password is stored in the operating system's native credential store (Windows) or in Eclipse Preferences (Linux ®).
  - It can be retrieved and used to authenticate the next time the client is started.
  - User is never prompted for a password.
- How to enable platform single sign-on
  - 1. During installation
  - 2. After installation, using the password preference page.
  - 3. In the plugin_customization.ini file in the branding plugin, set the values for these preferences to **true**:
    - com.ibm.rcp.security.auth.ui/ssoAllowed – Boolean value. Determines whether or not users have the option of using single sign-on.
    - com.ibm.rcp.security.auth.ui/ssoEnable – Boolean value. Determines whether or not users have the option of turning single sign-on on or off.

Single sign-on (SSO) authenticates users by prompting them for a user name and password a single time. Enabling platform single sign-on gives users secure access to the platform key store without displaying additional authentication prompts. To enable platform single sign-on, perform the steps provided on this slide.

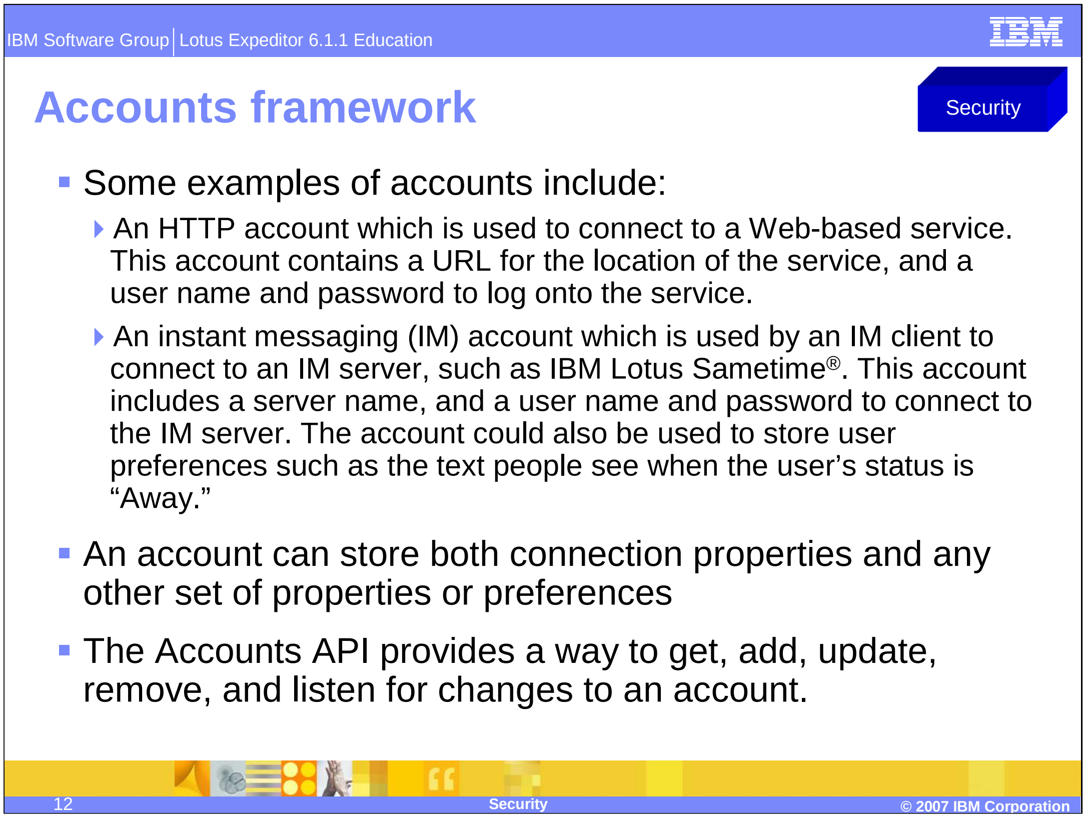


Accounts framework

Security

- Store account-related data, such as username, password
  - Account properties are stored in Eclipse preferences, except for the password, which is stored in the Java keystore
- Integrates with the JAAS framework to authenticate with servers

The Accounts framework enables you to store, access, and use properties that are required to make a connection to, and communicate with, a local or remote service.



Accounts framework

Security

- Some examples of accounts include:
  - An HTTP account which is used to connect to a Web-based service. This account contains a URL for the location of the service, and a user name and password to log onto the service.
  - An instant messaging (IM) account which is used by an IM client to connect to an IM server, such as IBM Lotus Sametime®. This account includes a server name, and a user name and password to connect to the IM server. The account could also be used to store user preferences such as the text people see when the user's status is "Away."
- An account can store both connection properties and any other set of properties or preferences
- The Accounts API provides a way to get, add, update, remove, and listen for changes to an account.

Some examples of accounts include:

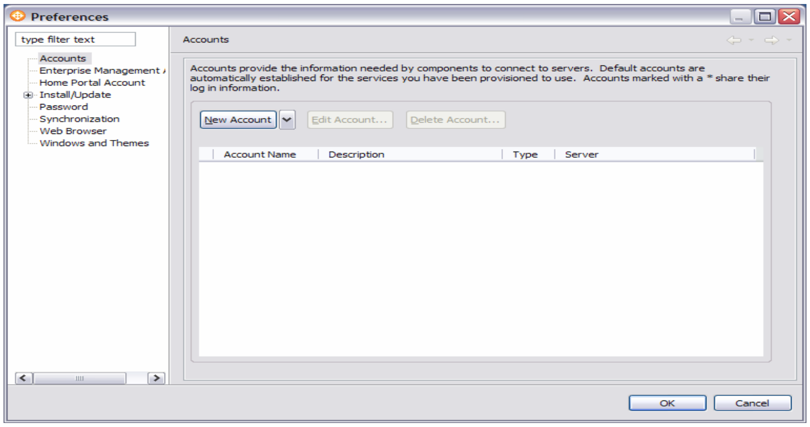
•An HTTP account which is used to connect to a Web-based service. This account contains a URL for the location of the service and a user name and password to log onto the service.

•An Instant Messaging account which is used by an instant messaging client to connect to an IM server, such as IBM Lotus Sametime. This account includes a server name, and a user name and password to connect to the instant messaging server. The account could also be used to store user preferences such as the text people see when the user's status is "Away."

Creating an account

Security

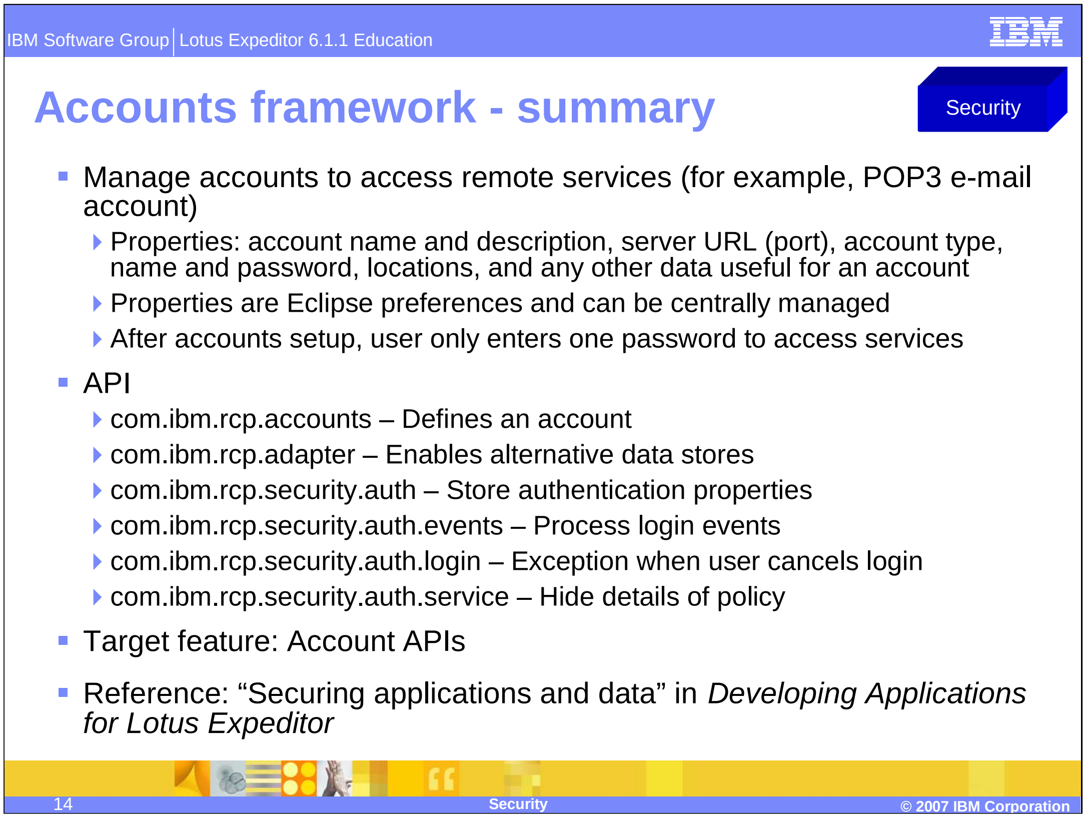
- Accounts Preferences page
  - Create a new account



Accounts provide the information needed by components to connect to servers. Default accounts are automatically established for the services you have been provisioned to use. Accounts marked with a * share their log in information.

Account Name	Description	Type	Server

You can create a new account using the Account Preferences page

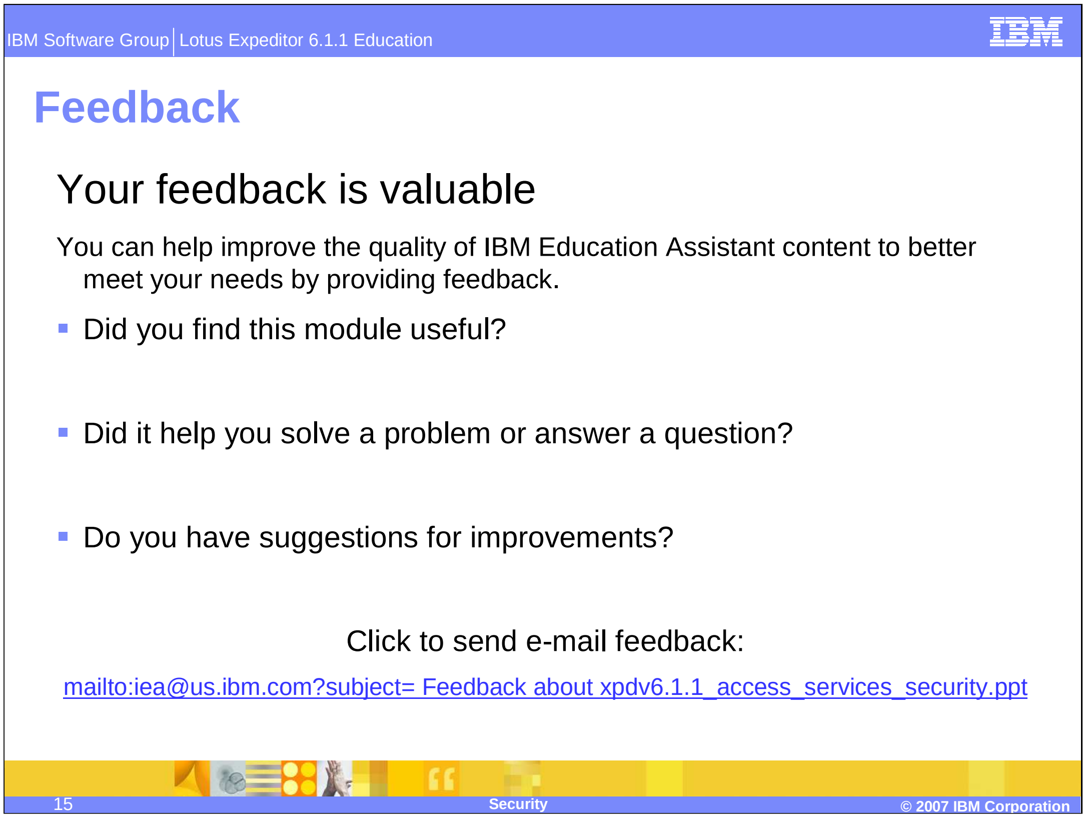


Accounts framework - summary

Security

- Manage accounts to access remote services (for example, POP3 e-mail account)
  - Properties: account name and description, server URL (port), account type, name and password, locations, and any other data useful for an account
  - Properties are Eclipse preferences and can be centrally managed
  - After accounts setup, user only enters one password to access services
- API
  - com.ibm.rcp.accounts – Defines an account
  - com.ibm.rcp.adapter – Enables alternative data stores
  - com.ibm.rcp.security.auth – Store authentication properties
  - com.ibm.rcp.security.auth.events – Process login events
  - com.ibm.rcp.security.auth.login – Exception when user cancels login
  - com.ibm.rcp.security.auth.service – Hide details of policy
- Target feature: Account APIs
- Reference: “Securing applications and data” in *Developing Applications for Lotus Expeditor*

In Summary, the Accounts API provides a way to get, add, update, remove, and listen for changes to an account. The extension points support adding accounts and account adapters. More information can be found in *Developing Applications for Lotus Expeditor*.



Feedback

Your feedback is valuable

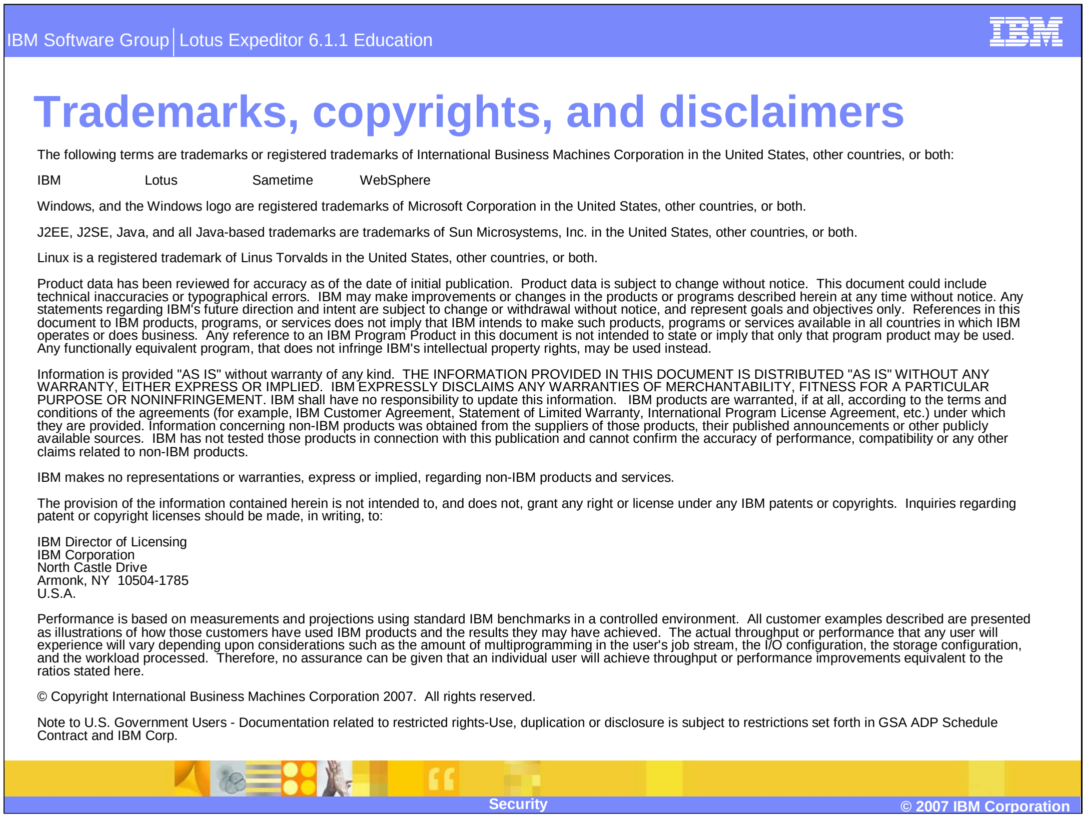
You can help improve the quality of IBM Education Assistant content to better meet your needs by providing feedback.

- Did you find this module useful?
- Did it help you solve a problem or answer a question?
- Do you have suggestions for improvements?

Click to send e-mail feedback:

mailto:iea@us.ibm.com?subject= Feedback about xpdv6.1.1_access_services_security.ppt

You can help improve the quality of IBM Education Assistant content by providing feedback.



Trademarks, copyrights, and disclaimers

The following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both:

IBM Lotus Sametime WebSphere

Windows, and the Windows logo are registered trademarks of Microsoft Corporation in the United States, other countries, or both.

J2EE, J2SE, Java, and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Product data has been reviewed for accuracy as of the date of initial publication. Product data is subject to change without notice. This document could include technical inaccuracies or typographical errors. IBM may make improvements or changes in the products or programs described herein at any time without notice. Any statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only. References in this document to IBM products, programs, or services does not imply that IBM intends to make such products, programs or services available in all countries in which IBM operates or does business. Any reference to an IBM Program Product in this document is not intended to state or imply that only that program product may be used. Any functionally equivalent program, that does not infringe IBM's intellectual property rights, may be used instead.

Information is provided "AS IS" without warranty of any kind. THE INFORMATION PROVIDED IN THIS DOCUMENT IS DISTRIBUTED "AS IS" WITHOUT ANY WARRANTY, EITHER EXPRESS OR IMPLIED. IBM EXPRESSLY DISCLAIMS ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NONINFRINGEMENT. IBM shall have no responsibility to update this information. IBM products are warranted, if at all, according to the terms and conditions of the agreements (for example, IBM Customer Agreement, Statement of Limited Warranty, International Program License Agreement, etc.) under which they are provided. Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products in connection with this publication and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products.

IBM makes no representations or warranties, express or implied, regarding non-IBM products and services.

The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents or copyrights. Inquiries regarding patent or copyright licenses should be made, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. The actual throughput or performance that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput or performance improvements equivalent to the ratios stated here.

Note to U.S. Government Users - Documentation related to restricted rights-Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract and IBM Corp.