Oracle IRM ‘Wrapper’
Any-format file encryption/decryption

Version 1.5.1

OVERVIEW

The Oracle IRM Wrapper application is an extension to the core Oracle IRM product (see the Oracle IRM pages on the Oracle Technology Network) that enables the manual encryption/decryption of files of any format on any Java-capable operating system, while leveraging the same classification-based rights model as the underlying Oracle IRM product.

> Note: Oracle IRM Wrapper only supports Oracle IRM 11g servers (not 10g).

Oracle Information Rights Management (IRM) is a Fusion Middleware security service that uses encryption to secure and track all copies of an organization's most sensitive documents and emails, regardless of how many copies are made, or where those copies are stored and used – *even when those copies are sent outside the firewall*. In Oracle IRM terminology the process of encrypting documents and emails is usually referred to as “sealing”. End user access to sealed documents is controlled by an easy to understand and highly scalable role- and classification-based rights model.

Oracle IRM servers expose a comprehensive set of IRM web services to enable the easy integration of “sealing” into the workflows of content management repositories, collaborative web applications, content filtering/monitoring systems, etc., and on end user computers the Oracle IRM desktop agent integrates with applications such as Microsoft Office and Adobe Reader so that “sealed'” documents can be used transparently, without the need to explicitly decrypt them. But this desktop integration means that Oracle IRM currently only supports a finite set of the most popular application formats on Windows-only operating systems.

Oracle IRM Wrapper introduces the related concept of “wrapping". It uses the same core IRM web services and classification-based rights model to manually encrypt and decrypt files in any format on any Java-capable operating system. The actual encryption wrapper is identical and wrapping (or unwrapping) is controlled by the same classification-based rights model as the core IRM product, but wrapped files require explicit unwrapping before they can be used, whereas sealed files can be transparently opened in desktop applications. Sealed files are identified by sealed file extensions, e.g. **myfile.sdoc**, wrapped files by an appended wrapped extension, e.g. **myfile.doc.irm**. The simple idea behind the Oracle IRM Wrapper application is that it enables Oracle IRM customers to apply their existing classification-based file encryption services to any file format on any Java-capable operating system.

WHAT’S NEW IN v1.5.1

- **Support for older versions of Mac OS X** – Oracle IRM Wrapper 1.5.1 adds support for legacy Mac OS X 10.4 and 10.5 operating systems, which are stuck on older Java versions (1.5) that does not support web services.

  Note: Users of more up-to-date versions of Mac OS X (that support Java 6) should remain on Oracle IRM Wrapper 1.5.0.

CONTENTS

INSTALLING ORACLE IRM WRAPPER

General

Oracle IRM Wrapper is a pure Java application so in theory it can be installed and run on any Java-capable operating system (with some caveats re the version of Java supported). The basic command line for wrapper is simple:

```
java –jar <path to wrapper.jar> <path to wrap.properties> <path to file> <optional default context>
```

So all you need to do is obtain the **wrapper.jar** Java archive and the **wrap.properties** configuration file from one of the platform-specific installation archives (e.g. **wrapper.zip** for Windows), edit wrap.properties to match your Oracle IRM installation, and you are ready to wrap/unwrap files. The wrapper.jar archive contains a sample wrap.properties configuration file which may be extracted using any ZIP utility.

You may need to pass some additional parameters to your Java runtime if you are using a SSL connection to the Oracle IRM server (recommended). The example below shows how to pass the Java runtime a Java keystore containing a trusted self-signed certificate (see “Oracle IRM 11g and HTTPS”) on Windows:

```
set IRM_W=C:\Documents and Settings\fred\Wrapper
set TRUST_W=-Djavax.net.ssl.trustStore="%IRM_W%\TrustMyOwnSelf.jks"
java %TRUST_W% -jar "%IRM_W%\wrapper.jar" "%IRM_W%\wrap.properties"
```

Note: Each of the operating system specific installers described in the following sections will generate a platform-specific launch script for Oracle IRM Wrapper. Edit these launch scripts to add parameters such as trusted certificate keystores.

Custom installers are provided for Windows, Mac OS X and Linux. These installers will register wrapped file icons and file types, launch scripts, right click wrap/unwrap menus, etc. as described in the following sections. If you are using Oracle IRM Wrapper on another operating system inspect the Windows, Mac OS X or Linux install files for guidance.

Windows

For your convenience Oracle IRM Wrapper comes with simple Windows installation scripts which are packaged into **wrapper.zip**. You will only need wrapper.zip as it contains all the required components. Perform the following steps to install:

1. Select a folder into which you wish to install, e.g. C:\Program Files\Wrapper.

2. Extract all the files from **wrapper.zip** into this folder.

3. Execute the **install.bat** script. This will install wrapped file types and associated icons into the Windows registry and generate application launch scripts **pre_wrap.bat** and **wrap.bat**. You may need to edit wrap.bat if you are using SSL to connect to the Oracle IRM server (recommended – see “Oracle IRM 11g and HTTPS”).

4. The configuration file **wrap.properties** should open in Notepad.

   Edit this file to match your Oracle IRM installation, i.e. your Oracle IRM server URL(s) and Oracle IRM account username.

Perform the following steps to uninstall:

1. Open your installation folder, e.g. C:\Program Files\Wrapper.

2. Execute the **uninstall.bat** script.

Mac OS X

Note: for older versions of Mac OS X (e.g. 10.4 and 10.5), which use Java 5, download and use the special disk image **wrapper_1_5_1.dmg** (in place of wrapper.dmg).

For your convenience Oracle IRM Wrapper comes with a simple Mac OS X installer packaged as a disk image **wrapper.dmg**. You will only need wrapper.dmg as it contains all the required components. Perform the following steps to install:

1. Download the **wrapper.dmg** disk image and mount it as a file system (by either double clicking it or using the OS X Disk Utility).

2. Drag the 'Oracle IRM Wrapper' application from the mounted disk image into your Applications folder. Doing this should cause the operating system to load the sealed and wrapped icons (from the application). You may need to restart Finder or log out and back in again to see the new file type icons.

3. Edit the **wrap.properties** configuration file, which is stored within the 'Oracle IRM Wrapper' application bundle:

    a. In Finder right click on the 'Oracle IRM Wrapper' application (in your Applications folder) and select Show Package Contents. This will open a new Finder window, showing the files and folders contained within the Oracle IRM Wrapper application bundle.

    b. Navigate into the Contents folder and then into the Resources folder.

    c. Right click on wrap.properties and select Open With > TextEdit.

    Edit **wrap.properties** to match your Oracle IRM installation, i.e. your Oracle IRM server URL(s) and username.

4. If you are using SSL to connect to the Oracle IRM server (recommended – see "Oracle IRM 11g and HTTPS") you may need to edit the Oracle IRM Wrapper launch script, which is stored within the 'Oracle IRM Wrapper' application bundle:

    a. In Finder right click on the 'Oracle IRM Wrapper' application (in your Applications folder) and select Show Package Contents. This will open a new Finder window, showing the files and folders contained within the Oracle IRM Wrapper application bundle.

    b. Navigate into the Contents/Resources/Scripts folder.

    c. Right click on **wrap.sh** and select Open With > TextEdit.

5. You can add the Oracle IRM Wrapper application to the Dock by dragging its icon from the Applications folder to the Dock.

6. You can create aliases for the Oracle IRM Wrapper application by right-clicking (or Control clicking) the Oracle IRM Wrapper icon in the Applications folder and selecting 'Make Alias'. Drag the resulting alias icon to its destination, e.g. the Desktop.

Perform the following steps to uninstall:

1. Drag the Oracle IRM Wrapper icon from the Applications folder to the Trash folder. This should remove the .irm file type icons (may require log out/in).

2. Delete all aliases for the Oracle IRM Wrapper application.

Linux

For your convenience Oracle IRM Wrapper comes with a simple Linux installer packaged as a compressed archive **wrapper.tar.gz**.

Note: These install scripts have only been tested on Oracle Enterprise Linux running the GNOME 2 desktop, but use cross-platform utilities – so they should work on other Linux environments, e.g. those using the KDE desktop.

Perform the following steps to install:

1. Download the **wrapper.tar,gz** archive and extract its contents into a folder in your user profile. It will create a wrapper subfolder, e.g. /home/fred/oracle/wrapper.

2. In the wrapper subfolder you will see the Oracle IRM Wrapper PDF documentation, the **wrapper.jar** Java archive, a sample **wrap.properties** configuration file and a set of install scripts.

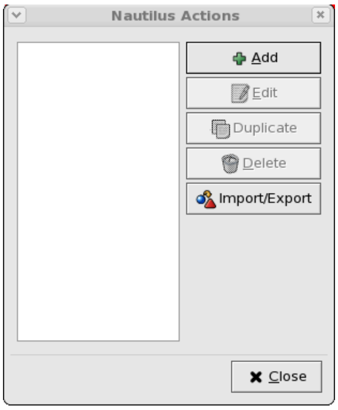
3. Open a terminal in the wrapper subfolder and execute the **install.sh** script. This will install the various wrapped MIME types and their associated icons, create an application launch script called **irm_wrap.sh** and an 'Oracle IRM Wrapper' launcher icon on the desktop. Do not run install.sh as root (e.g. using 'sudo').

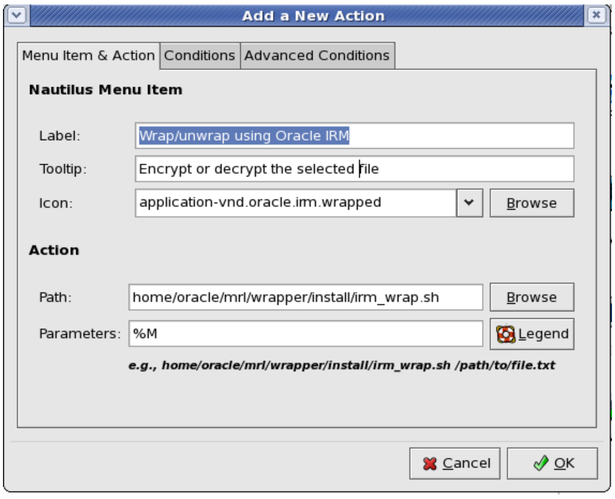
4. Edit **wrap.properties** to match your Oracle IRM installation, i.e. your Oracle IRM server URL(s) and username.

5. If you are using HTTPS you will need to edit the generated **irm_wrap.sh** launch script to provide a trusted key store (see "Oracle IRM 11g and HTTPS").

6. The installation thus far provides a Linux user with a drag target on the desktop onto which they can drop files to be wrapped or unwrapped (see "Running Oracle IRM Wrapper on Linux"). An additional, useful but manual installation step is to add wrapping/unwrapping as a right-click context menu to the Linux file manager (in this case Nautilus). This step assumes that the Nautilus Actions extension has been installed (search online for "Nautilus Actions") and that you can locate and launch the Nautilus Actions Configuration Tool.

   a. Open the Nautilus Actions Configuration Tool, click 'Add' and add an action pointing the **irm_wrap.sh** launch script in the wrapper sub-folder. Click OK once done.





Nautilus Menu Item

Label: Wrap/unwrap using Oracle IRM

Tooltip: Encrypt or decrypt the selected file

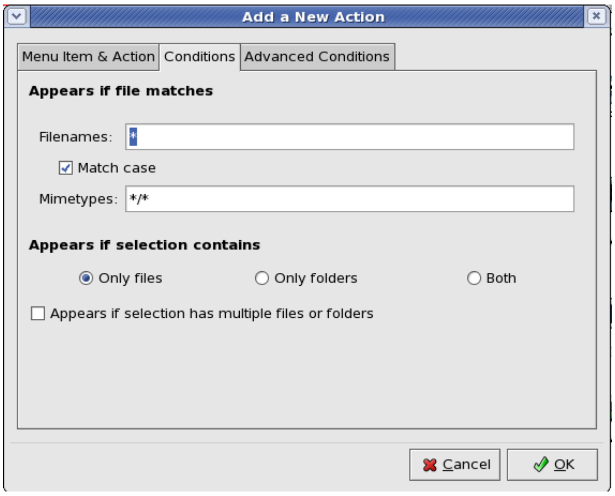
Icon: application-vnd.oracle.irm.wrapped

Action

Path: home/oracle/mrl/wrapper/install/irm_wrap.sh

Parameters: %M

e.g., home/oracle/mrl/wrapper/install/irm_wrap.sh /path/to/file.txt



Add a New Action

Menu Item & Action | Conditions | Advanced Conditions

Appears if file matches

Filenames: *

☑ Match case

Mimetypes: */*

Appears if selection contains

● Only files ○ Only folders ○ Both

□ Appears if selection has multiple files or folders

Perform the following steps to uninstall:

1. Open a terminal in your installation folder, e.g. /home/fred/oracle/wrapper.
2. Execute the **uninstall.sh** script.

CONFIGURING ORACLE IRM WRAPPER

Configuring the Oracle IRM Wrapper application

Oracle IRM Wrapper takes its configuration settings from the file **wrap.properties**, which it reads in at startup. This file uses the standard formatting of Java properties files (see Java documentation on java.sun.com or Google for “java property files”). A sample wrap.properties file is included in wrapper.jar, which contains detailed inline comments. To get Oracle IRM Wrapper working you will only need to edit the first three settings in the BASIC section of wrap.properties, but feel free to experiment with the settings in the ADVANCED section (after first making a copy of wrap.properties).

IMPORTANT - Edit the BASIC settings to match your environment!

wrap.properties

```
#
# Oracle IRM Wrapper configuration file
#
# BEFORE YOU BEGIN WRAPPING FILES YOU MUST EDIT THIS FILE TO POINT TO
# *YOUR* ORACLE IRM SERVER (see your IT Admins for help). THE URLs
# GIVEN BELOW ARE JUST FOR ILLUSTRATION PURPOSES. SEE THE ORACLE IRM
# WRAPPER DOCUMENTATION FOR MORE DETAILS ON EACH SETTING.
#
# Note: leading or trailing whitespace characters (in property values) are ignored.
#

# -----------------------------------------------------------------------------------------------
# BASIC SETTINGS - YOU *WILL* NEED TO EDIT THESE SETTINGS
# -----------------------------------------------------------------------------------------------

#
# The URL of your Oracle IRM server
#
oracle.irm.wrapper.irm_server=https://irm.oracle.demo

#
# The URL of your Oracle IRM 'sealing' server
#   Note: the Oracle IRM web services can be configured to run on a different host (for improved
#        performance, load balancing, etc.) known as the 'sealing' server.
#   Note: if not specified, defaults to the value of oracle.irm.wrapper.irm_server
#
#oracle.irm.wrapper.seal_server=http://irm.oracle.demo

#
# You can cache your Oracle IRM account username here
#   Note: if not specified, you will be prompted for the username
#
oracle.irm.wrapper.username=w_user

# -----------------------------------------------------------------------------------------------------------------------
# ADVANCED SETTINGS - YOU WILL GENERALLY *NOT* NEED TO EDIT THESE SETTINGS
# -----------------------------------------------------------------------------------------------------------------------

#
# 1 = delete original file after it has been wrapped/unwrapped
# 0 = preserve original file
# defaults to 1 (if not specified)
#
oracle.irm.wrapper.delete_original=1

#
# 1 = list classifications in which user has BOTH "Seal" and "Unrestricted Export" rights
#     (required to "wrap" AND "unwrap")
# 0 = list classifications in which user has "Seal" rights, warn if no "Unrestricted Export" rights
#     (required to "wrap")
# defaults to 1 (if not specified)
#
oracle.irm.wrapper.require_unrestricted_export=1
#
# 1 = use sealed file extensions for file types supported by the Oracle IRM desktop
#     (e.g. a.doc <-> a.sdoc, a.zip <-> a.zip.irm)
# 0 = use wrapped file extentions for all file types
#     (e.g. a.doc <-> a.doc.irm, a.zip <-> a.zip.irm)
# defaults to 1 (if not specified)
#
oracle.irm.wrapper.use_sealed_extensions=1

#
# 1 = select unsealed file extension (when multiple options, e.g. a.sjpg -> a.jpg or a.jpeg)
# 0 = use default unsealed file extension
# defaults to 1 (and only used when use_sealed_extensions=1)
#
oracle.irm.wrapper.select_unsealed_extension=1

#
# 1 = verbose logging (useful when debugging)
# 0 = concise logging
# defaults to 0 (if not specified)
#
oracle.irm.wrapper.verbose=0
```

Configuring the IRM server(s) – 11g

Oracle IRM 11g servers providing the IRM web services need a certain amount of configuration to support Oracle IRM Wrapper:

1. Oracle IRM 11g web services are enabled by default and listen on the URL and port(s) configured for Oracle IRM via the Oracle Enterprise Manager Fusion Middleware Control Console. By default the Oracle IRM sealing service runs on the same WebLogic managed server as the Oracle IRM server, but it can be run on a separate managed server (see next section).

    a. Example: if the URL sealed into content is https://irm.oracle.demo/irm_desktop then the IRM server URL you configure in **wrap.properties** should be https://irm.oracle.demo. You can discover the IRM server sealed into content by right clicking on a sealed file in Windows Explorer and inspecting its properties.

b. To confirm network connectivity to the Oracle IRM web services you can open your version of the URL https://irm.oracle.demo/irm_desktop in a browser.

2. Your Oracle IRM user account need to be have been assigned the appropriate rights to seal and reseal documents. For how to create user accounts and assign rights see the Oracle IRM documentation. Note: that the server-side Oracle IRM sealing service will check out rights on behalf of this user account, just as if it were an IRM desktop, so you need to ensure that this user is not competing with a desktop user for rights (you may need to use a special non-desktop user account).

3. The table below lists the document rights required by the user account calling the Oracle IRM 11g web services. These rights must be configured on the IRM server implementing the IRM web services. See the Oracle IRM 11g documentation for more detailed information.

Required Oracle IRM 11g document rights (per-context)	
Seal	The user must have the 'seal' right in the target context (to 'wrap' files to this context). This often corresponds to assigning the user a role with a name like 'Contributor'.
Exporting Content – Allow with no restrictions	The user must have the unrestricted export right in the target context (to 'unwrap' files from this context). This often corresponds to assigning the user a role with a name like 'Contributor with Export') which has both 'seal' and unrestricted export rights. Note: this right is configured on the IRM server as a role constraint ("Exporting Content – Allow with no restrictions"). Roles with this constraint are typically not available in contexts created from the Standard context template. You will need to create your wrapping contexts from an export-capable context template (i.e. a context containing export-capable roles).

Oracle IRM servers and Oracle IRM sealing servers – 11g

For more sophisticated deployments it is worth understanding the 11g IRM web services architecture. A typical 11g IRM server installation actually includes two WebLogic managed applications: the IRM server and the IRM sealing server. The IRM server stores and serves the encryption keys and access rights that power the whole IRM application, but it does not itself seal (encrypt) content. The IRM sealing server does the server-side sealing and exposes a set of web services for that purpose. The IRM sealing server and the IRM desktop agent are both clients of the core IRM server (one resides on the desktop while the other resides in the middle tier) and Oracle IRM Wrapper is a client of the IRM sealing service, not the core IRM server.

The advantage of this architecture is that the IRM sealing service can easily be deployed to other hosts (so that it is not consuming CPU cycles from the core IRM server) and multiple IRM sealing server instances on multiple hosts can be configured to seal content against the same core IRM server. Oracle IRM Wrapper uses the **irm_server** configuration parameter to point to the core IRM server and **seal_server** to point to the related IRM sealing service. If they are collocated (the default) then **seal_server** need not be specified. When Oracle IRM

Wrapper sends a file to the sealing service (to be sealed) it passes the value of **irm_server** to the sealing service so that it can request the relevant keys and rights from the core IRM server (IRM sealing service applications are not tied to a specific core IRM server).

This architecture, combined with the scalability of the core IRM server means that multiple IRM sealing server instances can be deployed across multiple hosts running on low cost hardware to automatically seal large volumes of content.

Oracle IRM 11g and HTTPS

Oracle IRM 11g supports web services over HTTP and HTTPS (SSL). HTTPS is more secure because it encrypts the network channel over which unsealed files are sent to the Oracle IRM sealing service, but it introduces the administration overhead of managing SSL certificates and making them available to Oracle IRM Wrapper, the core Oracle IRM server and to the Oracle IRM sealing server (see “Oracle IRM servers and Oracle IRM sealing servers – 11g***Error! Reference source not found.***”). The core Oracle IRM server hosts the SSL certificate while Oracle IRM Wrapper and the Oracle IRM sealing server must be configured to trust the SSL certificate.

For details on obtaining and installing SSL certificates on the core Oracle IRM server and Oracle IRM sealing server see the documentation for Oracle WebLogic or Oracle Content Management Suite (of which Oracle IRM 11g is currently a part). This requires a moderate understanding of the concepts behind SSL certificates: self-signed certificates, root certificates, trusted certificate chains, etc. – which is beyond the scope of this document.

Note: The Oracle IRM sealing server (which may not always be running on the same Oracle WebLogic managed server) communicates with the core Oracle IRM server using web services, so it is especially important that it be configured to trust the core IRM server’s SSL certificate. Failure to establish trust will result in server-side SSL exceptions. If you are using the Java API this step is unnecessary as it does not use the IRM sealing server.

Assuming that an SSL certificate has been obtained and installed on the server-side, perform the following steps to make it available to Oracle IRM Wrapper:

1. Check that the SSL certificate installed on the Oracle IRM 11g server is functional by using a web browser to browse to https:<host:port>/irm_desktop (inserting the host and port of your IRM server). If you are using real SSL certificates issued by a trusted certificate authority your browser should silently accept the server certificate and show you a lock icon via which you can inspect the SSL certificate (just like browsing to a trusted retail website). If you are using a self-signed test certificate you should be prompted to make an exception and import the self-signed certificate into the browser’s trusted certificate store, after which you can again inspect the certificate from the lock icon.

2. Note that the SSL certificate must have the following properties:
   a. Subject CN: should match hostname of web service URL
   b. Basic constraints: Subject Type=CA
   c. Key Usage: Non-critical, Key Encipherment and Certificate Signing

3. If the SSL certificate has been signed by a trusted certificate authority, and the root certificate for that authority is already in the trust store of your Java VM, you need do nothing more. The SSL stack of the Java VM running Oracle IRM Wrapper should automatically trust the IRM server’s SSL certificate.

4. If the SSL certificate is self-signed, or signed by some intermediate certificate authority whose root certificate is not in your Java VM’s trust store, then you need to export the SSL certificate (containing the SSL public key) into a keystore that can be made available to Oracle IRM Wrapper via the command line. You can do this either by exporting the certificate from the browser and importing it into a JKS or JCEKS

keystore, using the Java keytool utility (search online for documentation) or finding the keystore containing the certificate on the IRM server (see the IRM server documentation for SSL). Copy the keystore containing the SSL certificate to the computer running Oracle IRM Wrapper and modify its command line to:

```
set TRUST_W=-Djavax.net.ssl.trustStore=c:\temp\TrustMyOwnSelf.jks
java %TRUST_W% -jar %IRM_W%\wrapper.jar %IRM_W%\wrap.properties
```

Modify the keystore path to the correct value for your keystore. This argument tells the Java Virtual Machine to trust SSL certificates stored in that keystore. The example shown above is for Windows. You will need to locate the wrapper launch script or batch file to edit it (see the per-platform installation instructions).

5. **IMPORTANT**: In some versions of WebLogic the JAX-WS web services stack used by the Oracle IRM sealing server has a bug in its handling of hostname verification (comparing the hostname of the requested web service with the hostname in the certificate) so you may need to restart the Oracle IRM managed service with client-side hostname verification disabled, for example:

   ```
   set JAVA_OPTIONS=-Dweblogic.security.SSL.ignoreHostnameVerification=true
   D:\Oracle\Middleware\user_projects\domains\irm_domain\bin\startManagedWebLogic.cmd
   IRM_server1
   ```

   See WebLogic documentation for how to set this Java property in your system. Only try this as a last resort.

Language support

Oracle IRM Wrapper is internationalized but not fully localized. Within **wrapper.jar** is a properties file, **wrapperbundle.properties**, which is the English-language default Java resource bundle for all Wrapper UI and logging messages. This file can be copied, translated and renamed to support other locales, without rebuilding the application – simply insert the new resource bundles into wrapper.jar, with the correct filename for the desired locale. A hideous French translation is included for illustrative purposes.

For example, to add a Russian resource bundle:

1. Using a ZIP utility or JAR tool extract a copy of the file **wrapperbundle.properties** from the wrapper.jar archive.
2. Rename the copy of **wrapperbundle.properties** to have the appropriate locale suffix, e.g. **wrapperbundle_ru.properties**.
3. Using a text editor translate all the strings in **wrapperbundle_ru.properties**, on the right hand side of the = signs, to their Russian equivalent. For example **warn.bad_username=invalid or missing username** could become **warn.bad_username=недействительными или пропавших без вести Имя пользователя** (I have no idea whether this is a good translation).
4. Ideally you should add the translated **wrapperbundle_ru.properties** file back into wrapper.jar (alongside **wrapperbundle.properties**) but you could just add **wrapperbundle_ru.properties** to your CLASSPATH.
5. The UI and logging should now be in Russian!

Note: if a local language resource bundle is not found Oracle IRM Wrapper will revert to using the English (EN) strings found in **wrapperbundle.properties**.

USING ORACLE IRM WRAPPER

Sealed and wrapped file extensions

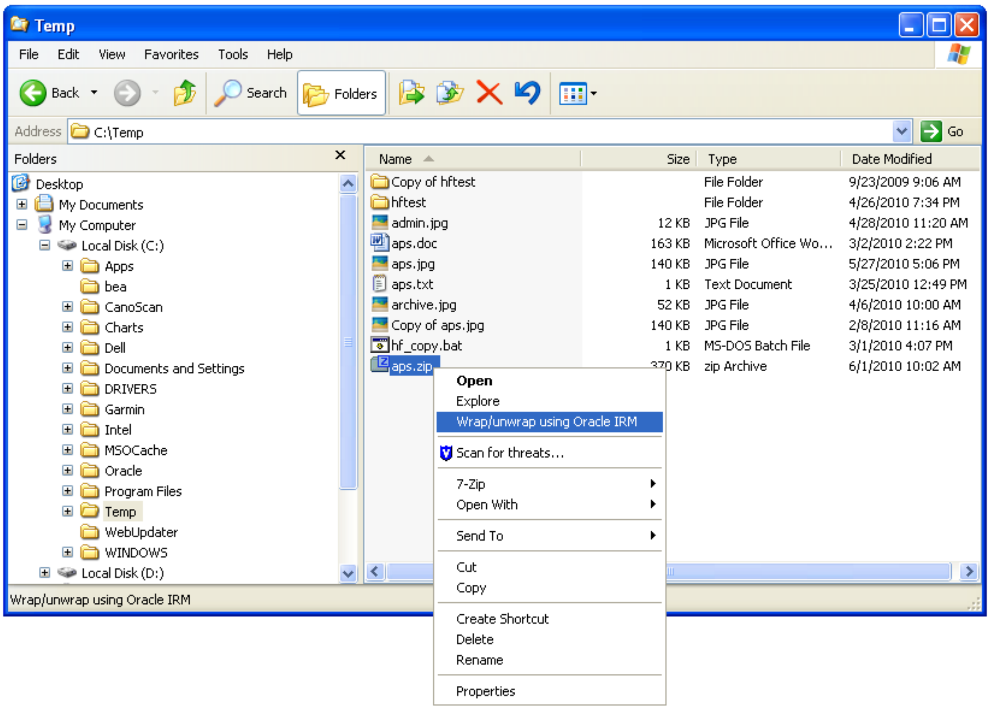
Oracle IRM Wrapper supports two types of file extensions:

- Sealed file extensions are used by the core IRM product to denote encrypted files, and there are about 30 such extensions, e.g. .sdoc, .sppt. When a file is sealed its existing extension is *replaced* by the sealed extension, e.g. a.doc⇒a.sdoc. File types for which there are sealed file extensions are known as ‘supported’ file types.
- The wrapped file extension (.irm) is used by Oracle IRM Wrapper to denote encrypted files. When a file is wrapped .irm is *appended* to the filename, e.g. a.zip⇒a.zip.irm. Files for which there are no sealed file extensions (i.e. they are not supported by the core Oracle IRM desktop product) are known as ‘unsupported’ file types.

By default Oracle IRM Wrapper will wrap/unwrap ‘supported’ file types to/from their sealed file extensions (mirroring the behaviour of the Oracle IRM desktop) and will wrap/unwrap ‘unsupported’ file types to/from the .irm file extension. This default behaviour can be overridden by changing the setting oracle.irm.wrapper.use_sealed_extensions to 0 in **wrap.properties**, in which case Oracle IRM Wrapper will wrap/unwrap both ‘supported’ and ‘unsupported’ file types to/from the .irm file extension.

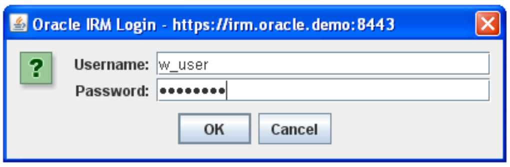
Wrapping a file - Windows

1. Right-click on any file and select “Wrap/unwrap using Oracle IRM".



Name	Size	Type	Date Modified
Copy of hftest		File Folder	9/23/2009 9:06 AM
hftest		File Folder	4/26/2010 7:34 PM
admin.jpg	12 KB	JPG File	4/28/2010 11:20 AM
aps.doc	163 KB	Microsoft Office Wo...	3/2/2010 2:22 PM
aps.jpg	140 KB	JPG File	5/27/2010 5:06 PM
aps.txt	1 KB	Text Document	3/25/2010 12:49 PM
archive.jpg	52 KB	JPG File	4/6/2010 10:00 AM
Copy of aps.jpg	140 KB	JPG File	2/8/2010 11:16 AM
hf_copy.bat	1 KB	MS-DOS Batch File	3/1/2010 4:07 PM
aps.zip	370 KB	zip Archive	6/1/2010 10:02 AM

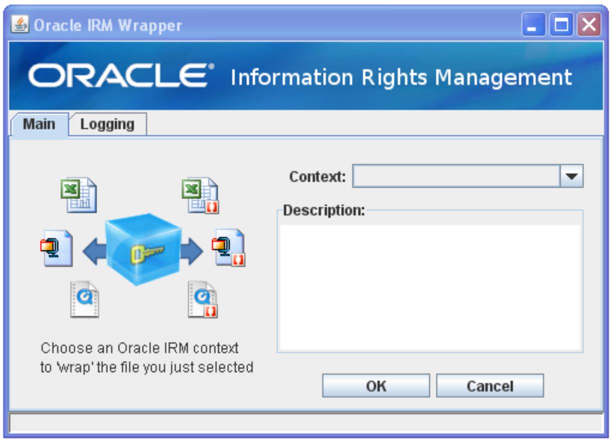
2. You should see the Oracle IRM Wrapper application and a login credentials dialog. Enter your Oracle IRM username and password and press OK.



Oracle IRM Login - https://irm.oracle.demo:8443

Username: w_user

Password: ••••••••



Oracle IRM Wrapper

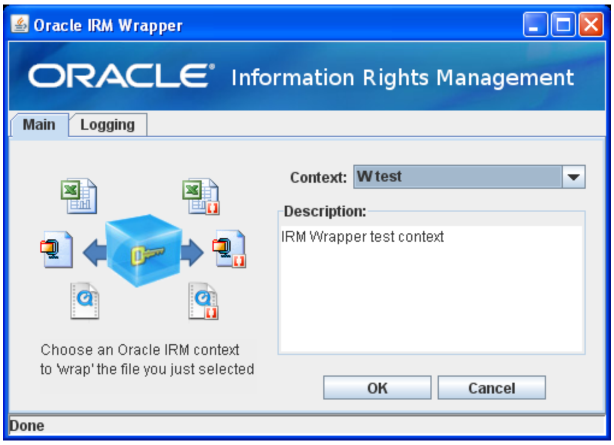
ORACLE Information Rights Management

Context:

Description:

Choose an Oracle IRM context to 'wrap' the file you just selected

3. Oracle IRM Wrapper communicates with the remote Oracle IRM server (via web services) and retrieves the list of classifications (also known as contexts) to which your user account has access.



Oracle IRM Wrapper

ORACLE Information Rights Management

Context: W test

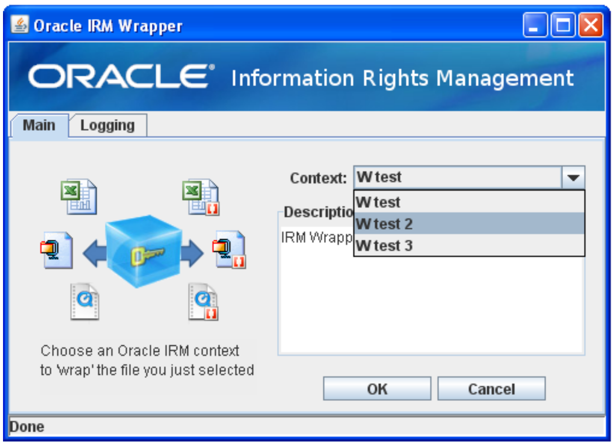
Description:

IRM Wrapper test context

Choose an Oracle IRM context to 'wrap' the file you just selected

Done

4. You can select contexts and review their descriptions.



Oracle IRM Wrapper

ORACLE Information Rights Management

Context: W test

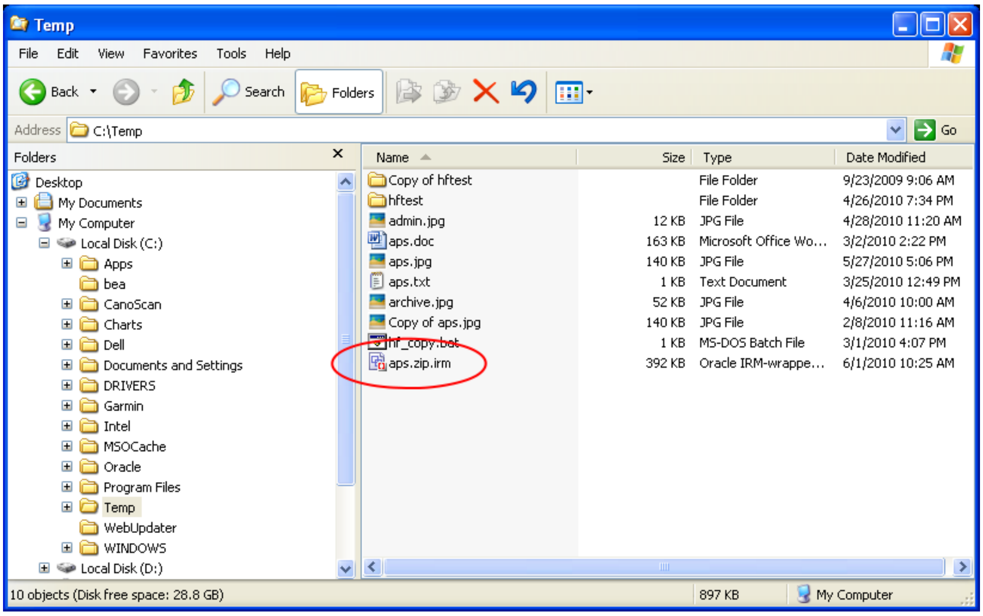
Descriptio

IRM Wrapp

Choose an Oracle IRM context to 'wrap' the file you just selected

Done

5. When you have selected a context press OK to wrap the file you originally selected. It should show up in the Windows Explorer as a wrapped file (with a .irm suffix added and a wrapped file icon).

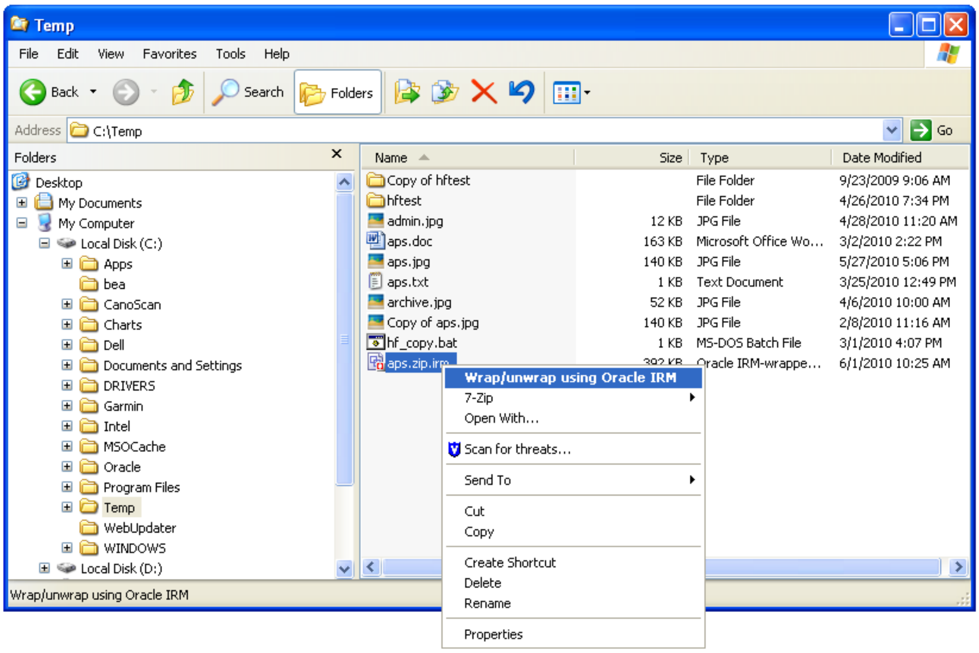


Name	Size	Type	Date Modified
Copy of hftest		File Folder	9/23/2009 9:06 AM
hftest		File Folder	4/26/2010 7:34 PM
admin.jpg	12 KB	JPG File	4/28/2010 11:20 AM
aps.doc	163 KB	Microsoft Office Wo...	3/2/2010 2:22 PM
aps.jpg	140 KB	JPG File	5/27/2010 5:06 PM
aps.txt	1 KB	Text Document	3/25/2010 12:49 PM
archive.jpg	52 KB	JPG File	4/6/2010 10:00 AM
Copy of aps.jpg	140 KB	JPG File	2/8/2010 11:16 AM
hf_copy.bat	1 KB	MS-DOS Batch File	3/1/2010 4:07 PM
aps.zip.irm	392 KB	Oracle IRM-wrappe...	6/1/2010 10:25 AM

10 objects (Disk free space: 28.8 GB) 897 KB My Computer

Unwrapping a file – Windows

1. Right-click on a wrapped file and select “Wrap/unwrap using Oracle IRM". Alternatively just double-click the wrapped file (unwrapping is the default action for *.irm files).

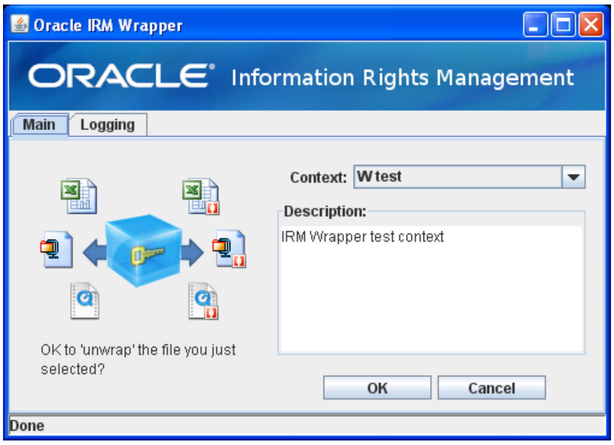


Name	Size	Type	Date Modified
Copy of hftest		File Folder	9/23/2009 9:06 AM
hftest		File Folder	4/26/2010 7:34 PM
admin.jpg	12 KB	JPG File	4/28/2010 11:20 AM
aps.doc	163 KB	Microsoft Office Wo...	3/2/2010 2:22 PM
aps.jpg	140 KB	JPG File	5/27/2010 5:06 PM
aps.txt	1 KB	Text Document	3/25/2010 12:49 PM
archive.jpg	52 KB	JPG File	4/6/2010 10:00 AM
Copy of aps.jpg	140 KB	JPG File	2/8/2010 11:16 AM
hf_copy.bat	1 KB	MS-DOS Batch File	3/1/2010 4:07 PM
aps.zip.irm	392 KB	Oracle IRM-wrappe...	6/1/2010 10:25 AM

Wrap/unwrap using Oracle IRM

2. As with wrapping files you will be challenged for your Oracle IRM username and password. Enter them and press OK.

3. The Oracle IRM Wrapper application sends the file to the remote Oracle IRM web server for scanning (often referred to as “peeking”) and displays the context to which it was originally wrapped. If you wish to unwrap the file press OK.



Context: W test

Description: IRM Wrapper test context

OK to 'unwrap' the file you just selected?

4. The file should be restored to its original unwrapped state.

Drag & Drop Wrapping – Windows

Oracle IRM Wrapper supports drag & drop wrapping. Just drag a file to a desktop icon and it will be automatically wrapped or unwrapped. You can have multiple icons (drop targets) and specify a different context for each drop target. So instead of selecting contexts from a pull-down list you can just drag your file to the appropriate icon.

To set up drag and drop:

- Open your Oracle IRM Wrapper installation folder and locate the file **pre_wrap.bat** (created during the installation process).
- In Windows Explorer Right-click on pre_wrap.bat and select “Select > Desktop (create shortcut)” from the context menu. This will create a shortcut to pre_wrap.bat on your desktop called **Shortcut to pre_wrap.bat**. By right-clicking on this shortcut and selecting “Properties” you can change its name and icon. Give it a memorable name such as “Oracle IRM Wrapper”.
- If you drag files from other folders and drop them onto the desktop shortcut it will launch the Oracle IRM Wrapper just as if you right-clicked on the file and selected “Wrap/unwrap using Oracle IRM".

This becomes useful when you have a small number of contexts to which you regularly wrap files. If you want to have a drop target icon for each context:

- In your installation folder make a copy of **pre_wrap.bat** and name it after a context, e.g. **pre_wrap_contextA.bat**.
- Follow the previous procedure to create a corresponding shortcut on your desktop and give it a memorable name such as “Context A – Oracle IRM”.
- In your installation folder edit pre_wrap_contextA.bat to include a line

```
set IRM_W_PARAMS=%1 "Context A"
```

- This appends “Context A” to the parameters passed to the Oracle IRM Wrapper application, resulting in it being automatically selected. A sample line is already included in pre_wrap.bat.

Selecting a default classification - Windows

If you find yourself wrapping files to one classification (context) more often than others you can configure Oracle IRM Wrapper to automatically select this classification (if present) from the pulldown list in the main user interface. To configure a default classification:

- In your installation folder edit **pre_wrap.bat** to include a line

  ```
  set IRM_W_PARAMS=%1 "Context A"
  ```

  where “*Context A*” is the name of your default classification (quotes are required if the classification includes whitespace characters).

- This appends “Context A” to the parameters passed to the Oracle IRM Wrapper application, resulting in it being automatically selected. A sample line is already included in pre_wrap.bat.

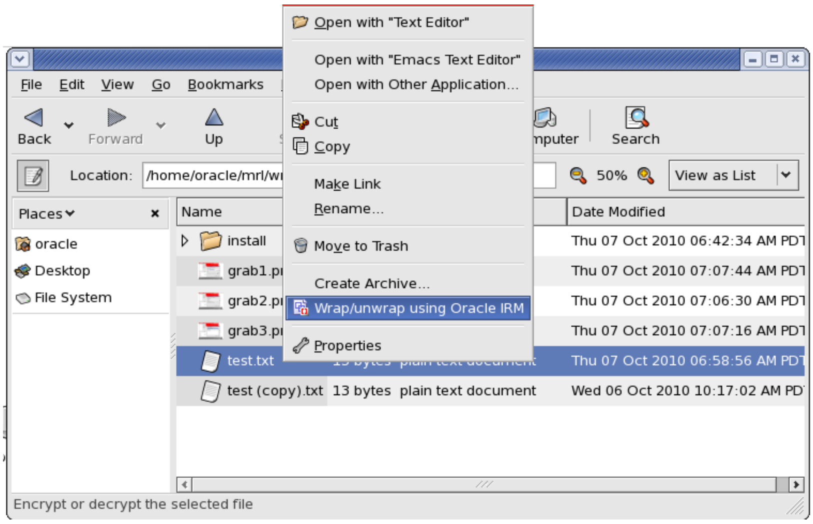
- Note how a similar configuration is used for drag & drop configuration.

Running Oracle IRM Wrapper on Mac OS X

Since Oracle IRM Wrapper is a pure Java application its user interface on Mac OS X is the same as on Windows (see previous sections). Launching Oracle IRM wrapper on Mac OS X is slightly different. If you have followed the instructions in the “Mac OS X” section of “INSTALLING ORACLE IRM WRAPPER” you should have Oracle IRM Wrapper icons in your Applications folder and possibly on the Dock and on your Desktop. To wrap/unwrap a file, select it in the Finder and drag it onto one of these icons. If you double-click an already wrapped file it will launch Oracle IRM Wrapper in unwrapping mode.

Running Oracle IRM Wrapper on Linux

Since Oracle IRM Wrapper is a pure Java application its user interface on Linux is the same as on Windows (see previous sections). Launching Oracle IRM wrapper on Linux is also very similar. If you have followed the instructions in the “Linux” section of “INSTALLING ORACLE IRM WRAPPER” you should have Oracle IRM Wrapper icon on your Desktop and (possibly) a right-click context menu in the Nautilus file manager. To wrap/unwrap a file, select it in the file manager and drag it to the ‘Oracle IRM Wrapper’ desktop icon, or right-click and select the ‘Wrap/unwrap using Oracle IRM’ context menu.

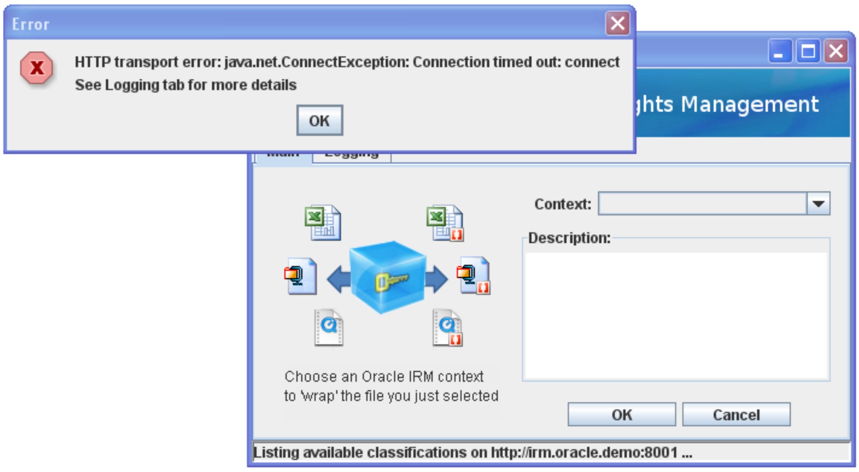

File Edit View Go Bookmarks
Back
Forward
Up
mputer
Search
Location: /home/oracle/mrl/wı
50%
View as List
Places
oracle
Desktop
File System
Name
Date Modified
install
grab1.pı
grab2.pı
grab3.pı
test.txt
test (copy).txt 13 bytes plain text document
Thu 07 Oct 2010 06:42:34 AM PDT
Thu 07 Oct 2010 07:07:44 AM PDT
Thu 07 Oct 2010 07:06:30 AM PDT
Thu 07 Oct 2010 07:07:16 AM PDT
Thu 07 Oct 2010 06:58:56 AM PDT
Wed 06 Oct 2010 10:17:02 AM PD
Open with "Text Editor"
Open with "Emacs Text Editor"
Open with Other Application...
Cut
Copy
Make Link
Rename...
Move to Trash
Create Archive...
Wrap/unwrap using Oracle IRM
Properties

Encrypt or decrypt the selected file

If you double-click an already wrapped file it will launch the Oracle IRM Wrapper in unwrapping mode.

Error handling

If the Oracle IRM Wrapper application experiences an error (for example the remote Oracle IRM server is inaccessible) you may be shown an error dialog.

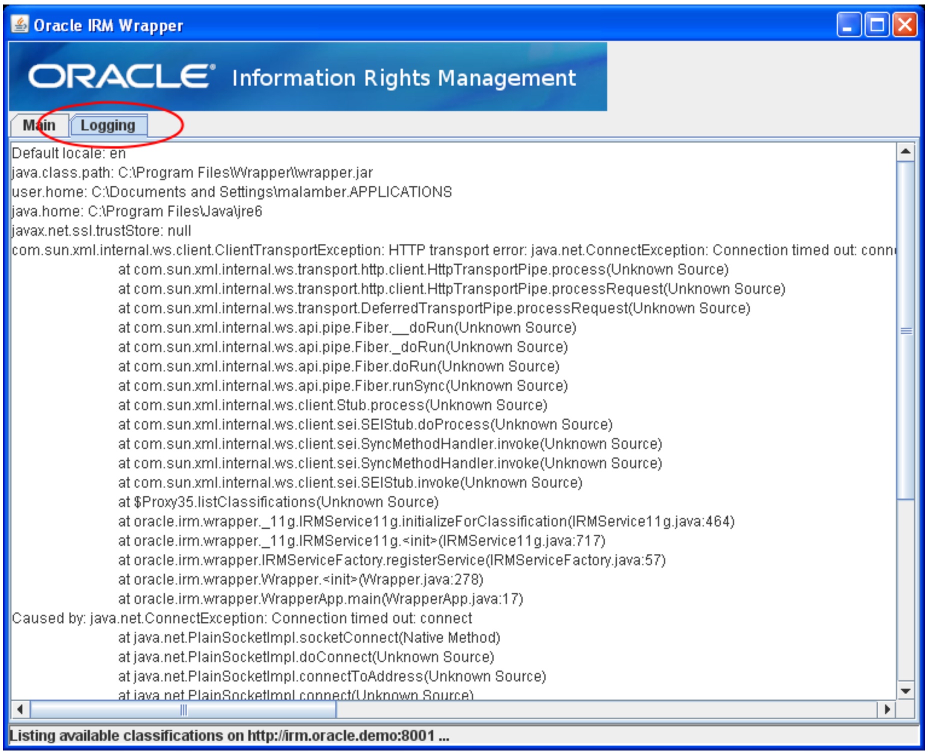


HTTP transport error: java.net.ConnectException: Connection timed out: connect
See Logging tab for more details

Choose an Oracle IRM context to 'wrap' the file you just selected

Listing available classifications on http://irm.oracle.demo:8001 ...

Further diagnostic information can be obtained from the logging tab on the main Oracle IRM Wrapper application window. You can resize this window to view the logging details.



```
Default locale: en
java.class.path: C:\Program Files\Wrapper\\wrapper.jar
user.home: C:\Documents and Settings\malamber.APPLICATIONS
java.home: C:\Program Files\Java\jre6
javax.net.ssl.trustStore: null
com.sun.xml.internal.ws.client.ClientTransportException: HTTP transport error: java.net.ConnectException: Connection timed out: conn
            at com.sun.xml.internal.ws.transport.http.client.HttpTransportPipe.process(Unknown Source)
            at com.sun.xml.internal.ws.transport.http.client.HttpTransportPipe.processRequest(Unknown Source)
            at com.sun.xml.internal.ws.transport.DeferredTransportPipe.processRequest(Unknown Source)
            at com.sun.xml.internal.ws.api.pipe.Fiber.__doRun(Unknown Source)
            at com.sun.xml.internal.ws.api.pipe.Fiber._doRun(Unknown Source)
            at com.sun.xml.internal.ws.api.pipe.Fiber.doRun(Unknown Source)
            at com.sun.xml.internal.ws.api.pipe.Fiber.runSync(Unknown Source)
            at com.sun.xml.internal.ws.client.Stub.process(Unknown Source)
            at com.sun.xml.internal.ws.client.sei.SEIStub.doProcess(Unknown Source)
            at com.sun.xml.internal.ws.client.sei.SyncMethodHandler.invoke(Unknown Source)
            at com.sun.xml.internal.ws.client.sei.SyncMethodHandler.invoke(Unknown Source)
            at com.sun.xml.internal.ws.client.sei.SEIStub.invoke(Unknown Source)
            at $Proxy35.listClassifications(Unknown Source)
            at oracle.irm.wrapper._11g.IRMService11g.initializeForClassification(IRMService11g.java:464)
            at oracle.irm.wrapper._11g.IRMService11g.<init>(IRMService11g.java:717)
            at oracle.irm.wrapper.IRMServiceFactory.registerService(IRMServiceFactory.java:57)
            at oracle.irm.wrapper.Wrapper.<init>(Wrapper.java:278)
            at oracle.irm.wrapper.WrapperApp.main(WrapperApp.java:17)
Caused by: java.net.ConnectException: Connection timed out: connect
            at java.net.PlainSocketImpl.socketConnect(Native Method)
            at java.net.PlainSocketImpl.doConnect(Unknown Source)
            at java.net.PlainSocketImpl.connectToAddress(Unknown Source)
            at java.net.PlainSocketImpl.connect(Unknown Source)
```

Listing available classifications on http://irm.oracle.demo:8001 ...

RELATED INFORMATION

To learn more about Oracle Information Rights Management (IRM), Oracle JDeveloper and web services in general, refer to:

- Oracle Information Rights Management on OTN