# IDOL
# Panopticon

Software Version 12.12

## Panopticon Java Programming Guide

MICRO FOCUS

Document Release Date: June 2022
Software Release Date: June 2022

## Legal notices



## Documentation updates

The title page of this document contains the following identifying information:

- Software Version number, which indicates the software version.
- Document Release Date, which changes each time the document is updated.
- Software Release Date, which indicates the release date of this version of the software.

To check for updated documentation, visit https://www.microfocus.com/support-and-services/documentation/.

## Support

Visit the MySupport portal to access contact information and details about the products, services, and support that Micro Focus offers.

This portal also provides customer self-solve capabilities. It gives you a fast and efficient way to access interactive technical support tools needed to manage your business. As a valued support customer, you can benefit by using the MySupport portal to:

- View information about all services that Support offers
- Submit and track service requests
- Contact customer support
- Search for knowledge documents of interest
- View software vulnerability alerts
- Enter into discussions with other software customers
- Download software patches
- Manage software licenses, downloads, and support contracts

Many areas of the portal require you to sign in. If you need an account, you can create one when prompted to sign in.

# Contents

Chapter 1: Introduction ..... 5
Overview ..... 5
Features ..... 5
Known Limitations ..... 5
Requirements ..... 6
Supported Platforms ..... 6
Supported Compilers ..... 6
Software Dependencies ..... 6
Windows Installation ..... 7
UNIX Installation ..... 8
Package Contents ..... 9

Chapter 2: Use Panopticon ..... 10
Decrypt Microsoft Azure RMS Protected Files ..... 10
Configure the Proxy for RMS ..... 10
Panopticon Sample Program ..... 11

Send documentation feedback ..... 13

# Chapter 1: Introduction

This guide is for developers who want to incorporate Micro Focus KeyView Panopticon into their applications using a Java development environment. It is intended for readers who are familiar with Java.


- Overview ..... 5
- Features ..... 5
- Known Limitations ..... 5
- Requirements ..... 6
- Windows Installation ..... 7
- UNIX Installation ..... 8
- Package Contents ..... 9


## Overview

Micro Focus Panopticon enables you to decrypt files that have been protected by Microsoft Azure Rights Management System (RMS), which is part of Azure Information Protection, allowing your workflow to operate on the original, unencrypted file. You can use Panopticon with existing workflows to allow complete access to protected data for which the service has permission.

Panopticon is part of the KeyView suite of products. KeyView provides high-speed text extraction, conversion to web-ready HTML and well-formed XML, and high-fidelity document viewing.

## Features

KeyView Panopticon enables the following features:

- Automatic detection of encryption type.
- Decryption of most file formats protected with RMS (text-only for PDF).

## Known Limitations

Panopticon decrypts most RMS encrypted documents, with the following known limitations:

- Text is decrypted from RMS protected PDFs, but not text formatting, images or subfiles.
- Email clients such as Microsoft Outlook can protect email messages as rights-managed email

messages. In these cases, it stores the contents of the original message as an encrypted rpmsg attachment. Panopticon does not support decryption of these encrypted attachments.

# Requirements

This section describes the supported platforms, compilers, and dependencies for Panopticon.

## Supported Platforms

Panopticon is supported on the following platforms:

**Microsoft Windows x86 64**

- Windows Server 2019
- Windows Server 2016
- Windows Server 2012
- Windows 10

**Linux x86 64**

The minimum supported versions of particular Linux distributions are:

- Red Hat Enterprise Linux (RHEL) 7
- CentOS 7
- SuSE Linux Enterprise Server (SLES) 12

## Supported Compilers

| Component | Compiler |
|---|---|
| Java components | Java 8 to 17 |

## Software Dependencies

To run Panopticon on Windows requires the Microsoft Visual C++ 2019 redistributables to be installed. The redistributables are provided in the `vcredist` folder of the Panopticon SDK but you can download the latest installers from Microsoft to get the latest security, reliability, and performance improvements.

To run Panopticon on 64-bit Linux requires `libstdc++.so.6` and `libgcc_s.so.1` from GCC 5.4. For your convenience, these are provided in the `redist` folder of your Panopticon installation.

**NOTE:** The `kvoop`, `servant`, and `WKOOP` executables must be able to link to `libstdc++.so.6` and `libgcc_s.so.1`.

- If these are installed in a system folder, like `/lib64`, KeyView will find them automatically.
- If you prefer you can add the path of the folder containing these libraries to the environment variable `LD_LIBRARY_PATH`.

Some components require specific third-party software:

- On Linux platforms you must install the following dependencies, which are required by the embedded browser (WKOOP):

| RHEL 7 / CentOS 7 | RHEL 8 | SLES 15 |
|---|---|---|
| libatomic<br>libX11<br>libXtst<br>libXScrnSaver<br>libXcomposite<br>atk<br>at-spi2-core<br>at-spi2-atk<br>cups<br>cairo<br>pango<br>alsa-lib-devel | libatomic<br>libX11<br>libX11-xcb<br>libXtst<br>libXScrnSaver<br>libXcomposite<br>atk<br>at-spi2-core<br>at-spi2-atk<br>cups<br>cairo<br>pango<br>alsa-lib-devel | libatomic1<br>libX11-6<br>libXtst6<br>libXss1<br>libXcomposite1<br>at-spi2-core<br>cups<br>libcairo2<br>libpci3 |

  For example, on CentOS 7:

```
sudo yum install libatomic libX11 libXtst libXScrnSaver libXcomposite atk
at-spi2-core at-spi2-atk cups cairo pango alsa-lib-devel
```

- Java Runtime Environment (JRE) or Java Development Kit (JDK) version 8 to 17 is required to use the Panopticon Java API.

# Windows Installation

To install the Panopticon SDK on Windows, use the following procedure.

**To install the Panopticon SDK**

1. Run the installation program, `Panopticon_`*`VersionNumber`*`_`*`Platform`*`.exe`, where *`VersionNumber`* is the product version number, and *`Platform`*is the operating system platform.

   For example:

   `Panopticon_12.12_Windows_X86_64.exe`

The installation wizard opens.

2. Read the instructions and click **Next**.

   The License Agreement page opens.

3. Read the agreement. If you agree to the terms, click I accept the agreement, and then click **Next**.

   The Installation Directory page opens.

4. Select the directory in which to install Panopticon. To specify a directory other than the default, click , and then specify another directory. After choosing where to install Panopticon, click **Next**.

   The Pre-Installation Summary opens.

5. Review the settings, and then click **Next**.

   The SDK is installed.

6. Click **Finish**.

# UNIX Installation

To install the Panopticon SDK, use one of the following procedures.

**To install the Panopticon SDK from the graphical interface**

- Run the installation program and follow the on-screen instructions.

**To install the Panopticon SDK from the console**

1. Run the installation program from the console as follows:

   `./Panopticon_`*`VersionNumber_Platform`*`.exe --mode text`

   where:

| | |
|---|---|
| *`VersionNumber`* | is the product version number. |
| *`Platform`* | is the name of the platform |

2. Read the welcome message and instructions and press `Enter`.

   The first page of the license agreement is displayed.

3. Read the license information, pressing `Enter` to continue through the text. After you finish reading the text, and if you accept the agreement, type **Y** and press `Enter`.

   You are asked to choose an installation folder.

4. Type an absolute path or press `Enter` to accept the default location.

   The Pre-Installation summary is displayed.

5. If you are satisfied with the information displayed in the summary, press `Enter`.

   The SDK is installed.

# Package Contents

The Panopticon installation contains:

- Libraries and executable files necessary for detecting the encryption type and decrypting files.
- The include files that define the functions and structures used by applications to establish an interface with Panopticon.
- A C sample program that demonstrates Panopticon functionality.
- (Windows only) Microsoft Visual C++ 2019 redistributable files.
- The Java API (the JAR package `Panopticon.jar`), Javadoc documentation, and a sample program written in Java.

# Chapter 2: Use Panopticon

This section describes how to perform some basic tasks using Panopticon.


- Decrypt Microsoft Azure RMS Protected Files ..... 10
- Configure the Proxy for RMS ..... 10
- Panopticon Sample Program ..... 11


# Decrypt Microsoft Azure RMS Protected Files

This section describes the steps required to use Panopticon to decrypt files protected with Microsoft Azure Rights Management System (RMS) with the Java API. For more information about the objects and methods, refer to the Javadoc documentation provided in the package.

**To decrypt protected files**

1. Construct a `PanopticonLicense` object, using the company name and 31-character license key that Micro Focus has provided.
2. Start a Panopticon session by constructing a `Panopticon` object. Pass to the constructor your `PanopticonLicense` object, the absolute path of the `bin` directory in the Panopticon installation, and the absolute path of a directory where Panopticon can create temporary files.
3. Determine the type and level of support for a particular file by calling the `getEncryptionInfo()` method of the `Panopticon` object with the absolute path of the file.
4. Configure Panopticon to use the RMS credentials for your application:
    a. Construct an `RMSCredentials` object with the credentials for your application.
    b. Call the `configureRMS()` method of the `Panopticon` object.
5. If decryption is supported, decrypt the file by calling the `decryptFile()` method. Use absolute paths to specify the input file and the output file to create.
6. When you have finished using Panopticon , call the `close()` method to end the session.

> **TIP:** To do this automatically, use the `Panopticon` object in a try-with-resources statement.

# Configure the Proxy for RMS

When Panopticon needs to access contents that are protected by RMS, it must make HTTP requests. By default, Panopticon uses the system proxy settings for these requests.

To use different proxy settings, you can configure them in the `[RMS]` section of the `cryptographyservices.cfg` configuration file. The following table describes the available options.

| Parameter | Description |
|---|---|
| `UseSystemProxy` | Whether to obtain details about your HTTP proxy from the system. By default, this parameter is set to **`TRUE`**, which means:<br>• On Microsoft Windows platforms, KeyView reads the proxy settings that are configured in the Windows Control Panel.<br>• On Linux, KeyView reads the proxy settings from environment variables such as `HTTP_PROXY` and `HTTPS_PROXY`.<br>You can use `UseSystemProxy` instead of setting the other proxy parameters (`ProxyHost`, `ProxyPort`, `ProxyUsername`, and `ProxyPassword`). When `UseSystemProxy` is set to **`TRUE`**, you must remove these other parameters from your configuration.<br>Set `UseSystemProxy` to **`FALSE`** to use different proxy settings. In this case you must set at least `ProxyHost` and `ProxyPort`. |
| `ProxyHost` | The host name or IP address of the proxy server. |
| `ProxyPassword` | The password to use to authenticate with the proxy server. |
| `ProxyPort` | The port of the proxy server to use to access the repository. This port must be greater than 0, and less than 65535. |
| `ProxyUsername` | The user name to use to authenticate with the proxy server. |

# Panopticon Sample Program

Panopticon includes a sample program, written in Java, which demonstrates how to use Panopticon through the Java API. The sample program determines the type of encryption used to encrypt a file and, if possible, decrypts the file.

The source code for the sample program, and a compiled `PanopticonSample.jar`, are located in *`install`*`\javaapi\sample`, where *`install`* is the path to your copy of the Panopticon SDK.

When you run the program using the provided shell scripts, the working directory must be the *`install`*`\javaapi\sample` folder.

**To run the sample program**

1. Open a command prompt in the `javaapi\sample` directory.
2. Run the program:
    - Directly

```
java -Djava.library.path="bin_path" -jar PanopticonSample.jar bin_path
license_org license_key tenant_id client_id client_secret input_file output_
file
```

- Using the Windows shell script

  ```
  PanopticonSample.bat bin_path license_org license_key tenant_id client_id
  client_secret input_file output_file
  ```

- Using the Linux shell script

  ```
  PanopticonSample.sh bin_path license_org license_key tenant_id client_id
  client_secret input_file output_file
  ```

The arguments are as follows.

| | |
|---|---|
| *bin_path* | The path to the Panopticon `bin` directory. |
| *license_org* | The organization name from your Panopticon license. |
| *license_key* | The key from your Panopticon license. |
| *tenant_id* | Your Microsoft Azure tenant ID. |
| *client_id* | The application client ID for authentication with Azure. |
| *client_secret* | The application client secret for authentication with Azure. |
| *input_file* | The path of the file to decrypt. |
| *output_file* | The path and file name to use for the decrypted file. |

# Send documentation feedback

If you have comments about this document, you can contact the documentation team by email. If an email client is configured on this system, click the link above and an email window opens with the following information in the subject line:

**Feedback on Micro Focus IDOL Panopticon 12.12 Panopticon Java Programming Guide**

Add your feedback to the email and click **Send**.

If no email client is available, copy the information above to a new message in a web mail client, and send your feedback to swpdl.idoldocsfeedback@microfocus.com.

We appreciate your feedback!