Offline Catalogs – Publisher's Guide

for the Drumlin Digital Rights Management (DRM) Service

Overview

There are three main ways of downloading a secured PDF (files in Drumlin Security's drmz encrypted format) from an external web server:

(i)   the publisher provides the end user with a web link for the file, the user clicks on the link (Windows and OSX users) and using the download facility of their web browser they SAVE the file, e.g. to their DOCUMENTS folder, and then open it with a Javelin secure PDF reader – this link can be to a drmz, drmx or zip file that contains one or more secured PDF files

(ii)   the user selects the download icon on the toolbar in the app and types in the web address provided, and Javelin will download the file for the user; or

(iii)   the user selects publisher's catalog they require. Then selects the file they want and it will download and be saved automatically

The option (iii) above is the simplest and most robust mechanism, and is the subject of this Publisher's Guide.

A catalog as used in the Drumlin DRM service is a specially formatted text file that lists each downloadable file with its name (title of the document) and where it can be found (its unique URL). The file format is described below and conforms to the xml (extended markup language) convention. Catalogs are quick and simple to create and may be updated almost instantly to reflect changes in the set of documents to be provided. Catalogs may also be "nested", so a top-level catalog can have sub-catalogs within it, and in turn these may have further levels of catalogs and documents.

A typical catalog entry might be:

```
<Document>
        <Name>Alice in Wonderland - Test Code: abcabcabc</Name>
        <URL>http://www.drumlinsecurity.co.uk/ipad/alice.drmz</URL>
        <ThumbURL>http://www.drumlinsecurity.co.uk/ipad/alice.png</ThumbURL>
</Document>
```

The beginning and end of any entry must start with a descriptor in angle brackets <descriptor> and end with the matching closing descriptor of the form </descriptor>. This is illustrated in the highlighted text above. These lines illustrate the minimum entry for a document (a PDF or drmz file): a Document outer descriptor, within which are descriptors for the document NAME, the URL (web location) where the file can be found; and ThumbURL, which can be blank, and provides the location of a cover image for that entry. Other optional fields are described below.

The catalog illustrated below is a sample default catalog included within the Javelin3 for Windows reader software (v3.0.71 or later). It includes an icon for refreshing (re-downloading) the catalog, which will update the available entries, plus a mix of downloaded files (e.g. the Alice in Wonderland example), pure PDF files (like this guide), and other Catalogs that are specific to individual publishers – for example, LaSota Racing, as shown in this screenshot. When any of these sub-Catalogs is clicked it will download the entries for that publisher's catalog, which may in turn include additional sub-Catalogs (e.g. Science books, Art books, Course1, Course2, etc).

The top section of the catalog provides icons for: Going back, Closing the catalog facility, refreshing the current catalog and displaying the catalog as a list or grid (toggle icon). Each icon responds to a single click and has a pop-up text indicating its function.
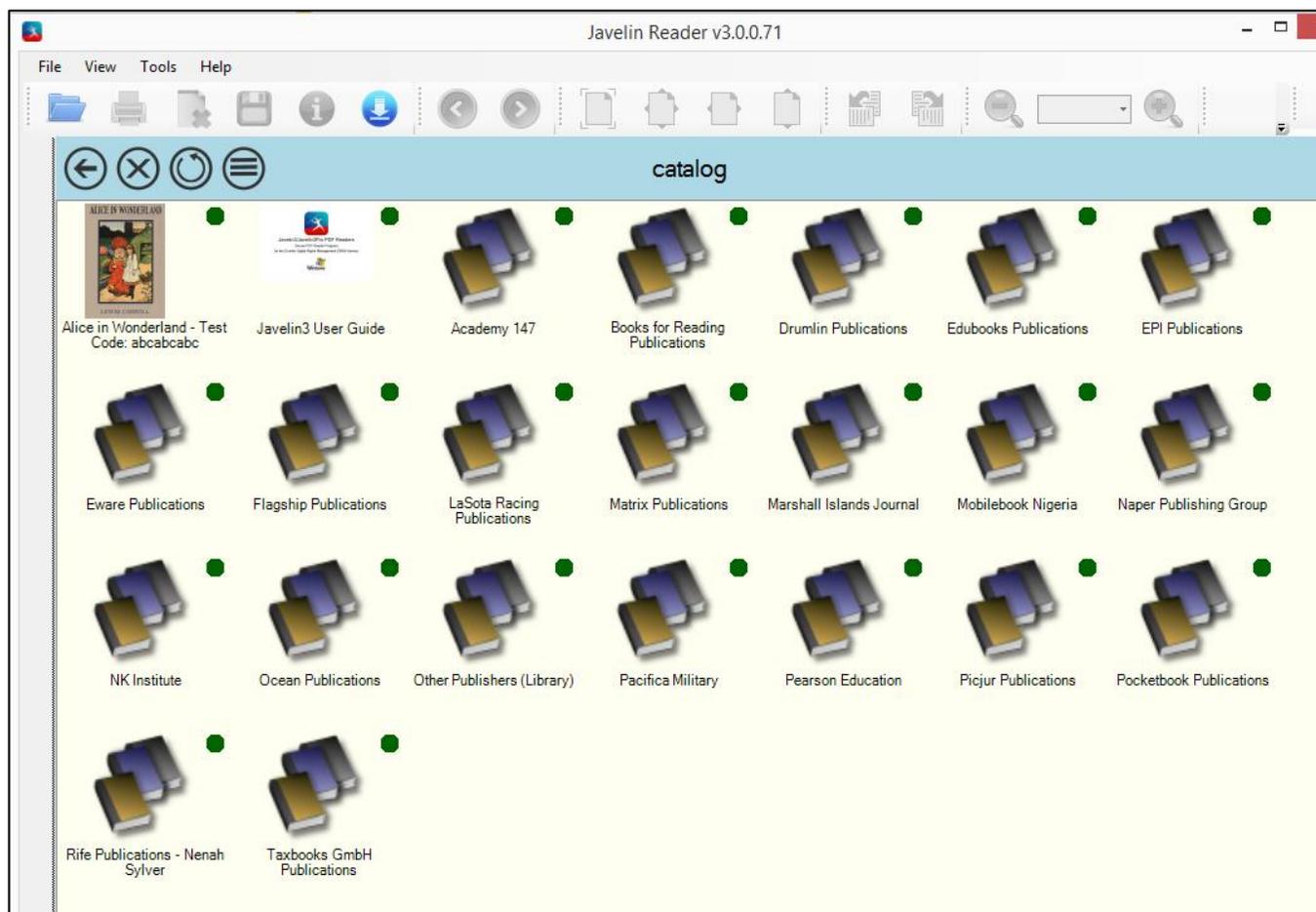
Right clicking on a catalog in Javelin3 for Windows will provide options to refresh the catalog, open the folder (in the user's AppData directory) where the Catalogs and downloaded files are stored, and optionally to view the catalog itself as a text file. Right-clicking on a downloaded and authorized file will provide the following facilities: Delete (deletes the downloaded file, not the catalog entry), Info (Information about the file), and Refresh (re-downloads the file and its cover image if provided by the publisher).

IMPORTANT NOTE: Catalog files are best created using the free Notepad++ text editor and set with Encoding type: UTF8. Because some devices and hosting service are case sensitive, we recommend that as far as possible lower case names for catalogs, zip files, and if necessary, URLs, are used. iOS devices, which are essentially based on Linux/Unix, are case sensitive. Avoiding special characters and spaces in URLs is desirable, although not always essential. Also note that drmz filenames can be changed without affecting their operation.

Downloading files and catalogs

For Javelin3 and Javelin3Pro for Windows, PDF and DRMZ files and XML and ZIP format catalogs can be downloaded directly using the Download icon on the main toolbar, as illustrated below.
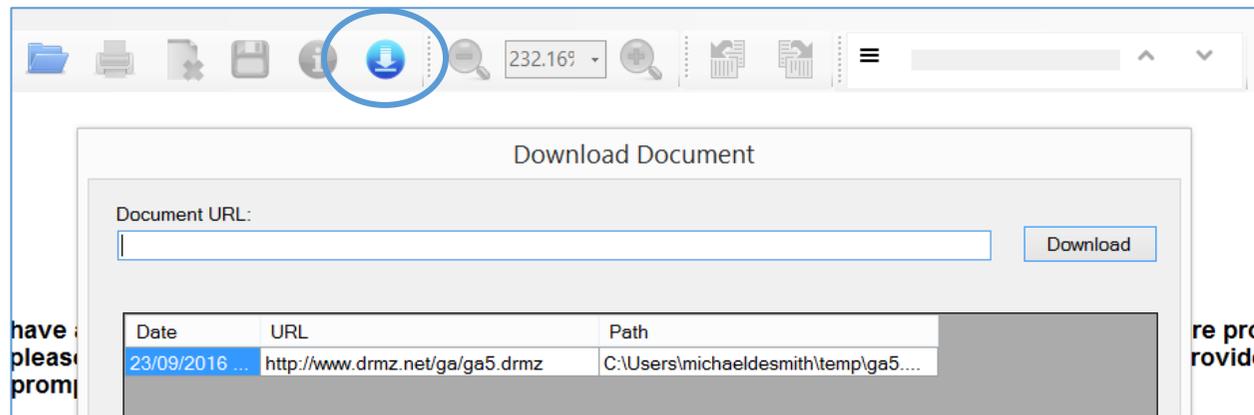
The document URL must be a valid web address and thus must not include spaces or a number of other special characters such as &; underline and dash characters are fine – short names without spaces are the most suitable. It is always a good idea to test a download link by entering it into a web browser's address bar, and then if it is accepted, copy the address bar contents as it will have automatically converted the URL to an acceptable form (e.g. spaces are replaced with %20). When files are downloaded via the above facility they are stored in a location specified by the user.

For mobile versions of Javelin (Javelin for iOS and Javelin for Android) similar catalogs are provided built-in to the apps or downloadable via the facility on the right-hand end of their home page toolbars. Examples include:

Javelin for iOS home page toolbar

Javelin for Android home page toolbar is similar. Files and catalogs downloaded via these downloading facilities are saved on the app's Home page.

Creating Catalogs

As noted above, catalogs are simply a special type of text file. However, because they need to be compliant with xml and encoded like web pages using UTF-8 encoding, we strongly recommend they are created using the free text editor for Windows called Notepad++ - it can be obtained from here: https://notepad-plus-plus.org/

If another text editor is used, for example on a Mac platform, then users should ensure that the encoding is UTF-8 and the format of the xml file created is correct. A typical (single entry) xml catalog is shown below:

```
<?xml version="1.0" encoding="UTF-8"?>
<JavelinCatalog>
        <CatalogURL>http://www.pdfz.org/sample.xml</CatalogURL>
        <Document>
                <Name>Alice in Wonderland - Test Code: abcabcabc</Name>
                <URL>http://www.drumlinsecurity.co.uk/ipad/alice.drmz</URL>
                <ThumbURL>http://www.drumlinsecurity.co.uk/ipad/alice.png</ThumbURL>
                <Subtitle>Sample secure PDF document - Cover image courtesy the Gutenberg Project, Images section</Subtitle>
                <ISBN>-</ISBN>
                <PublisherName>Macmillan (in 1866)</PublisherName>
                <PublisherURL></PublisherURL>
                <Authors>Lewis Carroll (Charles Lutwidge Dodgson)</Authors>
                <AuthorURL></AuthorURL>
                <Language>English</Language>
                <Edition>1</Edition>
                <Description>The famous 19th century tale of Alice's adventures underground. Illustrated by John Tenniel</Description>
                <Review></Review>
                <PrintLength>78</PrintLength>
                <PublicationDate>1865-01-01</PublicationDate>
                <Price>Free</Price>
                <CurrencyCode></CurrencyCode>
        </Document>
</JavelinCatalog>
```

Most of the fields in the above <Document> structure are optional, but as can be seen in the above example the details may be included with the appropriate entries. If the descriptors are provided with no content they will be accepted. Many of these additional fields are displayed as part of the Information (INFO) facilities associated with documents downloaded via a catalog (i.e. metadata).

If a catalog xml file has another catalog within it, that second catalog is a separate xml file, which also needs to be created and stored on a website somewhere. For example, if you open the catalog illustrated in the previous section using Notepad++ you will find the following entry which refers to a second level catalog, lasota.xml:

```
<Document>
        <Name>LaSota Racing</Name>
        <URL>http://www.pdfz.org/lasota.xml</URL>
        ….
</Document>
```

Catalog files may also be specified using a standard compressed (zip) file URL – for example:

```
<URL>http://www.drumlinsecurity.co.uk/verlag/2017.zip</URL>
```

This format is specifically designed to allow a catalog to contain both a list of the publications AND a set of matching cover images that are automatically extracted and displayed by Javelin when the file is downloaded by the user AND optionally pdf and drmz files. If a standard xml catalog is used then an icon rather than a cover image will be shown for each document, underneath which will be the Title (NAME) of the document. After downloading and authorizing such documents the first page of the document (which might be a cover or a text page) is used to replace the icon.

One benefit of using sub-catalogs is that the full catalog name will be used (rather than the filename) after downloading.

Let us suppose we want to create a catalog for a university, with sub-catalogs/folders for each Faculty and within these, sub-catalogs for each Department and maybe within these Departments, course-related catalogs. To create your xml file, you would need a top level catalog, with entries that refer to (i.e. link to the URLs for) catalogs called faculty1.xml, faculty2.xml etc., and then within the faculty1.xml file you would have a department1.xml etc., and then within department1.xml you would have your department1-100.xml file department1-200.xml etc., and within these catalogs would be the links to the pdf files and drmz files (although such files can be included at higher levels in the 'tree', for example to provide information relating to that Faculty or Department). The customer would only need to download the top-level catalog, because having done so, each sub-catalog would automatically download when the catalog item was touched or clicked, and then

its elements would be available via the sub-catalog graphic. New items can be added to such catalogs simply by editing the version of the server where the catalog is held, and then touching or clicking the refresh icon (in the corner of each catalog) to re-download the latest version.

There is no specific limit to the number of entries in a catalog, or to the number of levels (sub-catalogs). However, we would recommend that no more than 100 entries are included in any one catalog, and ideally no more than 3 or 4 levels of catalogs are provided. Catalog URLs should ideally be standard http or https web addresses, but some other URLs are supported, notably Dropbox links (dl=0 or dl=1 links). The order of entries matter – catalog entries are displayed in the order in which they appear in the file and are not otherwise sortable or searchable (at least, not currently). End users have the option of displaying entries in the default format (a grid) or as a list (the latter allows for the display of longer document name entries and info details).

Within catalogs there are often cover images. The format of these is flexible, but a good rule of thumb is an image about 170×240 (width, height) – i.e. quite small and portrait orientation if possible, but 240×170 will work also or anything close to this arrangement – check you are happy with the results before issuing a catalog. Several points should be noted about cover images (or 'jackets'):

- cover image files should be in .png format (jpg is also supported)

- catalog files are best with short names. We can host xml catalogs for you if you wish on our pdfz.org website, so if the catalog is for manual download rather than via a higher level catalog, users simply type: pdfz.org/catalogname.xml to download your catalog (the http://www. bit is not required for this to work)

- as with the document files themselves, cover images can be refreshed/downloaded by long-touching the cover of the item of interest (or right-clicking for the Windows version), waiting for the pop-up menu to appear, and then asking for the image to be refreshed

- because mobile devices are poor at multi-threading, downloading images and catalog data at the same time is a problem. On Android the Javelin software attempts to retrieve all the cover images for a catalog and these are downloaded as per files using the built-in Android downloader. On iOS, which is not so good at performing this kind of function, cover downloading is manual, as described above, or you can used zipped catalogs (see further below)

- if no cover image is provided, or is available but not downloaded, the Javelin software will generate a cover using the first page of the document after it is downloaded and opened

- Javelin for mobile devices and Javelin3 for Windows recognize zipped xml files, which can contain cover images files, and even document files etc., within the zipped catalog. An example file of this type with 100 covers can be downloaded from http://www.pdfz.org/bhc.zip

The bhc.zip file contains two items:

bhc.catdir
bhc.xml

The first item is a folder (sub-directory) that contains copies of all the thumbnail images (cover images) for the documents listed in the second item, the catalog file bhc.xml. In this particular case there are lots of image files in the catdir folder. Note that both items are in lower case, and the matching items on the server should be in the same case.

Validation and testing

Having created a catalog, then next step is to make sure it works exactly as you expect. The best way to do this is as follows:

(i)   give the top level catalog a very short/simple name, using lower case letters, e.g. ucla.xml and then place it on a web server with a short name – if you would like us to host it on our pdfz.org or drmz.net servers then let us know and we can do that for you. All sub-catalogs should also be available online, with their URLs correctly specified as per the lasota.xml example above

(ii)   it is a good idea to check that the structure of your xml file is valid – you can do this using a range of tools, or the free online validator at: http://www.w3schools.com/xml/xml_validator.asp

(iii)   Then, once you are happy it is correct, download the catalog to Javelin3 for Windows and then Javelin for iOS and Javelin for Android using the Download icons provided in Javelin. Then test each level of the catalog, downloading each sub-catalog and opening it to check the next level, including downloading PDF or drmz files contained within the various catalog levels and checking the images

(iv)   You can ask us to check your catalogs if you are having problems, or we can create and optionally host, catalogs for you

Updating catalogs

Catalogs can be updated at any time, by simply editing the xml file offline and then uploading it to your server in the same location as was previously used. Customers can then see the updated information by refreshing the catalog using the Refresh icon in the top right corner of the catalog, or by downloading the new catalog. This is particularly useful when adding new titles.

If you remove a title from a catalog and the user refreshes it, that title will no longer by visible within it, even if the user has downloaded and authorized the file in question.

Updating of files WITHIN catalogs, e.g. PDF or drmz files, is simply a matter of updating the source files and loading them on to the relevant server from which they are retrieved. Users can then select the cover of the title in question on their mobile device, holding their finger on the cover for a couple of seconds until a pop-up menu appears (mobile apps) or right clicking (Windows version), and the option **Refresh file** will appear allowing them automatically to re-download the file and/or cover image in question.

To summarize: All current versions of Javelin support built-in and downloadable catalog files. These files are xml files that include links to the location of drmz files and cover image files. A zipped version of the file download can be provided that includes both the xml file and a sub-directory containing the cover images. Now these special zip format files can be refreshed with the zip file as the file to re-download rather than just the xml catalog file. This means that catalogs AND the cover images can be refreshed in one go, so that new entries and their cover images are downloaded when user select to refresh a catalog. An example is provided here: http://www.pdfz.org/ziptest.zip - the catalog refresh line for the xml catalog file itself includes the following entry:

<CatalogURL>http://www.pdfz.org/ziptest.zip</CatalogURL>

Thus the zipped version is re-downloaded after a refresh rather than just the xml file, resulting in the cover image updates to appear automatically as part of the refresh operation