Restricted and confidential property of Oracle.
Solely for use by recipient under agreement forbidding disclosure.

Outside In Clean Content SDK
Technical Note

Encrypted Document Support

Version 2.0 – October 3, 2012

Restricted and confidential property of Oracle.
Solely for use by recipient under agreement forbidding disclosure.

Change History

Version 1.0 – December 28, 2010
Initial version

Version 2.0 – October 3, 2012
Updated to include information regarding latest support for PDF encryption

Overview

Authoring applications often provide a set of security related features designed to restrict the access to and usage of the contents of a file. There is a wide variety of security features in place across the file formats supported by Clean Content. Many of these features obfuscate the content of the file through the use of encryption or other techniques. Other features provide only a weak level of protection by controlling access through the application user interface while leaving the underlying data unencrypted. In some cases a user defined password is required to decrypt content while other features only use a pre-defined encryption password.

Versions of Clean Content prior to 2011.1 focused on detecting and reporting the use of encryption and specific weak protections and also included support for decrypting documents that used a pre-defined password. Clean Content version 2011.1 added extensive support of extracting and analyzing password protected encrypted documents through the use of an option that provides a list of passwords. If the document leverages a supported encryption algorithm and any password from the list is verified then analysis and extraction of the associated encrypted content is enabled. The use of an unsupported encryption algorithm or failure to verify a required password is detected and reported and analysis and extraction is then limited to any unencrypted portions of the document. Scrubbing of encrypted documents is explicitly disabled.

This document outlines the specific level of support provided by Clean Content related to the use of document encryption.

Clean Content Encryption API Support

The Clean Content API includes two options and one analysis target related to encrypted document support. These features allow encrypted documents to be detected and decrypted.

The SecureOptions.PasswordList option can be used to provide a list of passwords to Clean Content that is leveraged to generate and validate an encryption key for encrypted documents. The passwords are provided as Unicode strings in clear text. In the interest of security, this option is never persisted to disk by Clean Content. When Clean Content encounters a document that is encrypted using a supported method it will attempt to validate any password in this list along with any applicable default password.

The SecureOptions.DecryptionStatus option provides result information detailing the decryption status of a document after analysis and extraction. There are five possible values described below.

- SecureOptions.DecryptionStatusOption.NotEncrypted:
  The document contained no encryption.

- SecureOptions.DecryptionStatusOption.DecryptedWithDefaultPassword:
  Parts of the document were encrypted and they were decrypted with the default password.
- SecureOptions.DecryptionStatusOption.DecryptedWithPasswordList:
  Parts of the document were encrypted and they were decrypted with a password from the list provided.
- SecureOptions.DecryptionStatusOption.DecryptionFailed:
  Parts of the document were encrypted but neither the default nor the provided passwords were successful.
- SecureOptions.DecryptionStatusOption.DecryptionNotSupported:
  Parts of the document were encrypted using an unsupported method.

The SecureOptions.Encryption analysis target tracks whether the document being analyzed includes any form encryption.

The Clean Content logging feature also generates warning and information logs that provide textual descriptions that further detail the type of encryption and decryption status.

The Clean Content demo application includes a user interface option to set the password list (see the Security button under the Extract Options and/or Scrub Options). The demo application will leverage this list when analyzing a single or group of documents. The single document mode will also report the decryption status on documents that include encryption.

AES 256 Installation Requirement

Clean Content leverages the Java™ Cryptography Extensions (JCE) for AES decryption. Due to import control restrictions the version of the JCE policy files that are bundled with the JREs that ship with Clean Content allow 'strong' but limited cryptography. Office or PDF documents encrypted with AES 256 bit encryption can only be decrypted if the Java Runtime Environment leveraged by Clean Content is updated to include the Unlimited Strength Java™ Cryptography Policy Files. These files can be downloaded from Oracle Technology Network and installed into the appropriate JRE location as documented in the download.

File Format Support Details

A variety of encryption algorithms may be used across the supported file formats. Each encryption algorithm leverages additional parameters ranging from an associated hashing algorithm and encryption key size, to specific algorithm details like chaining mode, spin count, and salt size. Additionally, the file format may support encrypting the entire document or leaving portions unencrypted (i.e. properties and embeddings). Each combination of variables represents a specific use case. The table below represents the general list of file format and encryption algorithm combinations supported by Clean Content. A more detailed description for each file format is contained in later sections.

File Format	**Cipher Algorithm**
Microsoft Word 97-2003	XOR (Weak Encryption)
Microsoft Word 97-2003	RC4 (Office 97/2000 Compatible)

Restricted and confidential property of Oracle.
Solely for use by recipient under agreement forbidding disclosure.

Microsoft Word 97-2003	RC4 (Crytpo API)
Microsoft Excel 97-2003	XOR (Weak Encryption)
Microsoft Excel 97-2003	RC4 (Office 97/200 Compatible)
Microsoft Excel 97-2003	RC4 (Crytpo API)
Microsoft PowerPoint 2002-2003	RC4 (Crytpo API)
Adobe PDF	RC4
Adobe PDF	AES
Microsoft Office XML formats (2007-2010)	Standard AES
Microsoft Office XML formats (2007-2010)	Agile AES
Microsoft Office XML formats (2007-2010)	Agile DES
Microsoft Office XML formats (2007-2010)	Agile 3DES
Microsoft Office XML formats (2007-2010)	Agile RC2

Microsoft Word Binary File Formats (97-2003)

Assigning a password to open to a Word binary document will result in encryption of some or all of the files contents using one of three implementations. Early versions of Word leveraged an Office implementation of the RC4 algorithm. As of Office 2002 users were allowed to select from a very weak and easily circumvented XOR algorithm, the Office RC4 implementation, or a CryptoAPI compatible RC4 implementation provided by one of the installed CryptoAPI services. The CryptoAPI services also included RC4 implementations that allowed the encryption key length to be set as high as 128 bits.

There are three areas within a Word document that may be encrypted. The primary word document content is encrypted in all cases described above. The document properties may optionally be encrypted or left unencrypted based on a user option when using the new RC4 CryptoAPI. Both XOR and the Office RC4 implementations leave properties unencrypted. The encryption of OLE embeddings occurs only when one of the newer CryptoAPI services is selected.

Clean Content supports decrypting all of the forms described above when given the valid password to open.

Note that assigning a password to modify does not result in encrypting the underlying file contents. Instead it only provides a limited level of document protection by requiring the user to re-enter the password to enable modifications. Clean Content flags this using the Weak Protections target.

Microsoft Excel Binary File Format (97-2003)

Microsoft Excel also allows for three different encryption implementations. The choices include an Excel specific version of an XOR algorithm, the Office RC4 implementation, or the RC4 CryptoAPI implementation.

The XOR and the Office RC4 implementations are limited to encrypting the Workbook stream and therefore left properties, embeddings, and various other pieces of data, unencrypted. Excel 2002 added the RC4 CryptoAPI option and allowed for the underlying encryption of properties, embeddings, and several other streams of data including change tracking information.

Assigning workbook protection to an Excel document also results in the encryption of the workbook content by using the Office RC4 implementation. Properties and embeddings remain unencrypted in this case. Excel uses a pre-defined password of 'VelvetSweatshop' to encrypt spreadsheets that leverage workbook protection that do not also require a password to open. Note that sheet protection does not result in any encryption.

Clean Content supports decrypting all of the forms described above when given the valid password to open or when the default password applies.

Using a password to modify by itself does not result in encryption. Clean Content flags password to modify, worksheet protection, and workbook protection as weak protections because they only provide user interface level protection that can be circumvented either with no password or with a pre-defined password.

Microsoft PowerPoint Binary File Format (97-2003)

PowerPoint did not begin offering encryption until version 2002 and therefore only allows the CryptoAPI RC4 implementation. Similar to Word and Excel, PowerPoint implements encryption when a password to open is applied. Unlike Word and Excel, PowerPoint also implements encryption when a password to modify is assigned to a document. However, a default password of '/01Hannes Ruescher/01' is leveraged during this form of encryption. This allows the presentation to be decrypted for read-only purposes without having to provide a password.

PowerPoint encrypts the presentation content, including OLE embeddings, and separately encrypts each image inserted into the presentation. Document properties may be optionally encrypted based on a user option.

Clean Content supports decrypting all of the forms described above when given the valid password to open or when the default password applies.

Microsoft Office XML File Formats (2007and above)

With the release of Office 2007 and the XML based ECMA-376 file formats Microsoft standardized the encryption architecture across all of the Office XML based file formats. This category covers Office 2007 and above versions of PowerPoint (pptx pptm potx potm ppam ppsx ppsm), Excel (xlsx xlsb xlsm xltx xltm xlam xlsb), and Word (docx docm dotx dotm). Three forms of encryption may be used within these file formats referred to as Standard, Agile, and Extensible.

The Standard form uses AES encryption with 128, 192, or 256 bit keys and is the default used in Office 2007.

The Agile form allows for a wider selection of cipher algorithms including AES, RC2, DES, DESX, 3DES, and 3DES_112. This list may also be extended to cover future and custom encryption algorithms. Agile allows a wide selection of associated hashing algorithms to be used for encryption key generation and validation. These include SHA-1, SHA256, SHA384, SHA512, MD5, MD4, MD2, RIPEMD-128, RIPEMD-160, and WHIRLPOOL. Many additional attributes associated with

the encryption process are also configurable when using this form of encryption including the encryption key size and spin count. Office 2010 uses Agile 128-bit AES encryption with the SHA1 hashing algorithm by default but also allows any Office implementation to be modified to leverage different algorithms and parameters.

The Extensible form allows a completely custom encryption implementation to be leveraged.

Clean Content supports extraction and analysis of Office XML formats that leverage AES, RC2, DES, and 3DES cipher algorithms combined with the hashing algorithms SHA-1, SHA256, SHA384, and SHA512, and MD5. Any encryption key size is supported with the caveat that use of a 256 bit key size requires installation of the Unlimited Strength Java™ Cryptography Policy Files as outlined in the implementation section. This covers the vast majority of encrypted Office XML documents. Clean Content does not support decrypting Office XML documents that leverage the Extensible encryption form or an Agile encryption form that uses a cipher or hashing algorithm not supported by Clean Content.

Adobe Portable Document Format

The Adobe PDF format has gone through numerous changes related to encryption between version 1.1 and the most recent version of the specification. Encrypted PDF documents include the name of a Security Handler that identifies the implementation associated with the documents encryption. The Standard Security Handler supports several forms of encryption and is leveraged when applying encryption passwords to documents using Acrobat Pro and many other PDF generation tools. PDF also supports the use of Public-Key Security Handlers. Clean Content includes support for encrypted PDF documents that leverage the Standard Security Handler as outlined below. Clean Content does not support decrypting documents that use Public-Key or any other security handler.

PDF documents that leverage the Standard Security Handler define the form of encryption through an algorithm identifier and a revision level. The algorithm identifier (V) includes values from 0 through 5. Algorithms 0 and 3 were not published and are not supported by Clean Content. Algorithms 1 and 2 leverage RC4 encryption and are supported by Clean Content. Algorithms 4 and 5 leverage AES 128 and AES 256 respectively and are supported by Clean Content through revision 6 of the Standard Security Handler. The use of a 256 bit encryption key size is supported with the caveat that it requires installation of the Unlimited Strength Java™ Cryptography Policy Files as outlined in the implementation section. Adobe Acrobat Pro X supports the 6th revision of the Security Handler that leverages a modified implementation of algorithm 5 that was previously used by Acrobat 9. This revision of the Security Handler is supported by Clean Content in both forms.

Encryption may be applied to all strings and streams stored inside the PDF document. Later revisions of the Security Handler allowed document content, document properties, and file attachments to be selectively encrypted or not encrypted. Revision level 4 and above allows any stream of data in the document to be selectively encrypted or not. Clean Content honors the selective encryption as it applies to document content, properties, and file attachments.

PDF supports the concept of both an owner and a user password. Specific document protection features may be enabled or disabled within a PDF application based on the type of password that is validated. Clean Content attempts to authenticate a given password as either the owner or user password and will enable content extraction and analysis if either is validated. Many PDF documents also employ a blank encryption password that allows a document to be encrypted but

Restricted and confidential property of Oracle.
Solely for use by recipient under agreement forbidding disclosure.

easily decrypted. Clean Content will always attempt to authenticate against a blank password to support this requirement.