Protection Profile

For

# Redaction Tools

14 May 2014

Version 1.0

# 1 PP Introduction

## 1.1 PP Reference Identification

PP Reference: Protection Profile for Redaction Tools

PP Version: 1.0

PP Date: 14 May 2014

## 1.2 CC Conformance

As defined by the references [CC1], [CC2] and [CC3], this PP conforms to the requirements of Common Criteria v3.1, Revision 4.

## 1.3 TOE Overview

Redaction is the process of selectively removing and replacing information from a document or other logical container of data for release to an audience not intended to view that information. Redacted information is not limited to classified material; other examples include privacy data, proprietary information, trade secrets, and legal strategy. Instances of redaction include replacing classified text with a black box to release a document to an unclassified environment, replacing privacy-related data such as telephone numbers with all Xs to release a database to a contractor, converting a proprietary format document to Portable Document Format (PDF) to release a what-you-see-is-what-you-get (WYSIWYG) document. The risk from improper or incomplete redaction is the inadvertent disclosure of classified or sensitive data.

Redaction has become a more complex problem due to electronic file formats, including documents, spreadsheets, databases or others that act as a logical container for data. These formats include the visual content as well as hidden data such as document properties, comments, and executable content. Users cannot easily access or examine some of the hidden data making it a substantial avenue for inadvertent disclosure. Redaction tools are designed to assist the user with review and removal of both visible and hidden information.

In the past, the typical redaction method for electronic documents was to print the document, black out the necessary information, then make a photocopy releasing a physical document. This paper method has the advantage of assurance but the disadvantage of loss of usability. The physical printed copy is exactly what is visible, a true WYSIWYG representation of the information. The photocopy could be scanned back into the computer for release as an electronic document, but the usability is greatly diminished. This method of redaction is no longer satisfactory in situations where the released document must retain the usability of the original source, or in situations where the object is not a document but a database table or other unprintable container.

Redaction is not sanitization. In the sanitization process, information is removed with no indication that the sanitization process took place. In the redaction process, selected visible information is removed and replaced with something innocuous (e.g. black box or text) so that the reader knows redaction took place. This replacement is a critical part of the process not shared with sanitization.

### 1.3.1 Types of Data - Terminology for Requirements

An electronic document is a logical container that is the product of an application such as a word processor or other text-based editor. Documents contain data in the form of text, images, video, audio, or other embedded logical containers such as spreadsheets (e.g. Microsoft Object Linking and Embedding (OLE)).

Data that users may need to redact from documents falls into two broad types: visible data and hidden data. The same element of a document can have both a visual representation as well as a different hidden representation. For example, the image displayed to the user may be a smaller size and resolution from the image stored in the internal format of the document file. The following terms for types of visible and hidden data are established for use in the requirements section of the protection profile:

Visible Data:

- Visible contents – the visible contents of the file; the visual representation of text, images and complex objects;
- Obscured visible data – content that could be visible but is obscured in some way such as content that runs off an edge of the container, text in a black font on black background (or any color of font on a similar color background), very small fonts, cropped or clipped graphics or images, hidden layers, portions of an embedded object (e.g. Microsoft OLE) that are outside the view container.

Hidden Data:

- Static data or metadata – file properties such as author or creation date, stored form field data, undo cache or any data kept to revert to a prior version of an element or the document itself, incremental updates, collaboration data such as comments, tracked changes, workflow data, embedded search indexes, bookmarks, document info added by 3$^{rd}$-party apps, accessibility data such as alternate text, etc.
- Structural data - data that is part of the file format structure, such as a file header or fonts, and is necessary to interpret the file properly for display or print
- Functional data – forms, scripts, link Uniform Resource Locators (URLs), workflow data, action buttons, formulas in a spreadsheet, macros or any type of executable content
- Remnant data – artifacts of the original application or source file format such as remnant or unreferenced data from fast saves, unreferenced or unused elements, malformed elements that cannot be fixed, garbage data in the file structure
- Images – the actual image data stored in the file as opposed to what is visible; the visible image can be cropped or resized but the full image could still be retained in the file format and may or may not match the visible image; some image formats can have their own metadata, such as Joint Photographic Experts Group (JPG) and Tagged Image File Format (TIFF)
- Complex Objects – objects that may have their own static or functional metadata and may differ between the stored and visible form, such as images, attachments, Microsoft OLE objects, Microsoft ActiveX controls, and temporal objects
- Temporal Objects – a particular type of complex object whose representation extends through a time interval, such as video, audio, flash animation, slide shows, etc. References to “complex objects” in the requirements section of this paper include temporal objects
- Metadata of objects or embedded objects – such as EXIF data of images; images themselves can contain other images and their own metadata
- Attachments – an electronic document or data file that is part of the main file but is logically distinct and separable from the main electronic document

### 1.3.2 Other Terminology

The following concepts are used in the assurance activities.

Simplify (or simplified) – replace a complex object or element with a single layer element that does not contain hidden or obscured data. The original representation of the object and all of the original hidden data must be removed from the document and the visible space must be replaced with the simplified element. For example, a document could contain an embedded spreadsheet on a page where only a small portion of the spreadsheet is visible within the view container on the page while content in the rest of the spreadsheet is in the document but hidden from the viewer. To simplify such an object, the TOE could create a single layer image (with no metadata of its own) from just the portion of the spreadsheet that is visible, place that image on the page and remove the embedded spreadsheet. This is just one solution. The intent of simplifying an object is to remove something complex that could contain hidden data and replace it with something simple that does not contain hidden data.

Examine a test file or document - the evaluator uses a hex editor or similar tool to view the raw binary (or hex) data of the file and the format structure. This allows the evaluator to view the contents of the file directly rather than through the TOE's interpretation of those contents. The examination can include the use of tools to extract, decode or decompress certain types of elements from the format, such as text or images. Care must be taken so that the extraction process does not change the element's size, resolution, or content. The Technical Community will identify appropriate tools.

Apply the TOE - follow the multi-step process where the user selects or marks areas or items in a document for redaction and then instructs the TOE to redact the marked areas and hidden information from the file.

Test files - to test the TOE, the evaluator will have to use test documents that have content expected to produce a certain result. The evaluator will apply the redaction tool to the test files and examine the output to determine if it is as expected. The same test documents can be used for each TOE that produces the same format as those test documents. For example, a set of PDF test files can be used for all PDF redaction tools. Test files will usually contain only one testable item at a time to make it easier to identify that item in the structure of the document, but some test files should contain multiple testable items. The Technical Community will create a set of test files for the assurance activities for the most common formats.

## 1.4 Use Case

A user needs to release an electronic document to an audience outside of some boundary. Some of the information in the document must not be available to that audience. The user must first redact this information from the document before moving the document across the boundary. The user manually selects the data to be redacted with an interactive software application. The format of the source and destination documents can be the same or different.

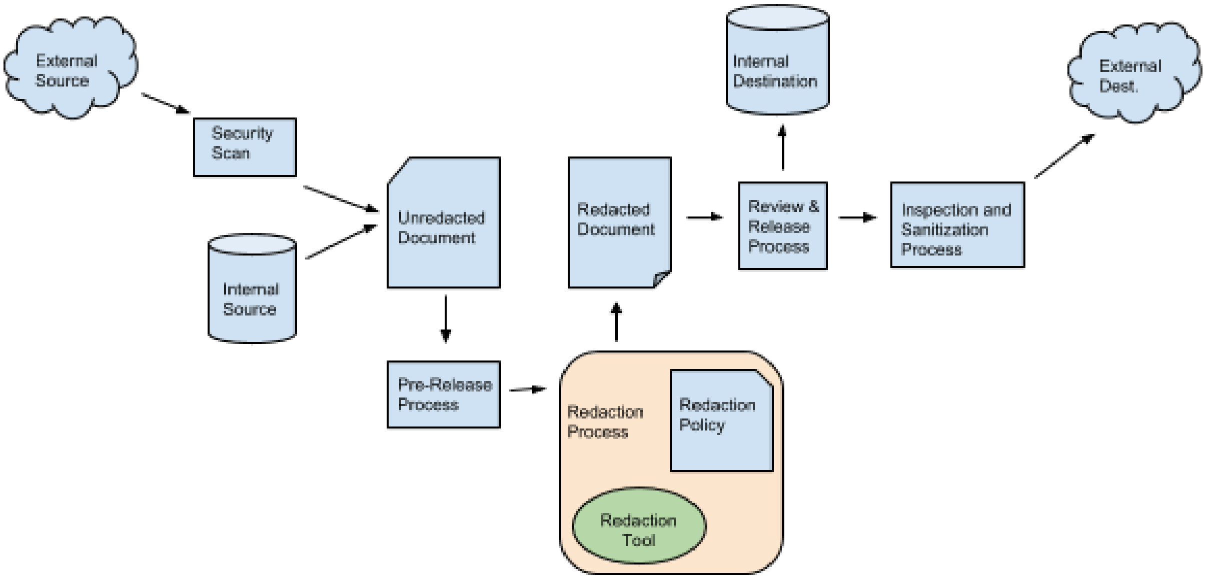

Figure 1: One possible workflow of an electronic document through the redaction process.

Figure 1 shows the typical workflow of a document from source to destination and through the redaction process. The original source of the document can be either internal or external to the system which hosts the redaction tool. If the source is external, the document will first go through the system's security policy for importing documents, e.g. virus scanning. A document from an internal source was either created on the system or has already gone through security scans. When a document enters the redaction process, it should have already been checked for malicious executable content.

During the pre-release process, the user or another authority in the workflow determines what to redact using the organization's information policies or declassification guidelines. The authority makes a determination about the release of images, audio, video or other complex objects that are in the document that may require other pre-processing before release. For example, the document will be examined for potential steganography attacks or a decision will be made to redact those portions that may contain such issues. For encrypted or password-protected files, the user will remove passwords and decrypt files during this phase, which assumes the host system is secured to a level appropriate to contain the decrypted data.

There are several different types of interactive redaction tools: stand-alone, plug-in, or multi-purpose. A stand-alone tool is self-contained and performs only redaction-related functions. A plug-in performs redaction-related functions as part of another application when installed in that application. A multi-purpose tool includes other functionality along with built-in redaction functions, such as a word processor that includes built-in redaction functionality or a redaction tool that also performs encryption and digital signature functions. This protection profile applies to the redaction functionality in all of these types of interactive tools.

During the redaction process, the user will manually mark visible data or select hidden data. The redaction tool will remove this data from the document in such a way that reverse engineering the document will not allow recovery of the redacted data. The organization should have a redaction policy that specifies which features of the redaction tool to use, such as whether to remove all hidden data automatically or to review and select elements individually, whether to replace redacted text with other text or with solid blocks of color, what color to use, etc. While this policy is part of the redaction process, it is separate from the redaction tool.

The organization will apply review and final approval policies during the review & release process. There may also be post-processing requirements for the document such as inserting special metadata, adding passwords, document encryption or digital signature. The redaction tool may include the functionality to perform these post-processing requirements but they are not within the scope of the redaction functionality of the tool.

Once approved for release, the organization may have additional inspection and sanitization requirements that depend on the source and destination. For example, documents on a classified system may have to go through additional processing before being moved to an unclassified destination.

Other workflows are possible. For example, decryption and re-encryption of data may be performed by the redaction tool without the data ever being saved as an intermediate file. Again, the host system is assumed to be secured to the level necessary for the decrypted data. Software vendors have the flexibility to devise their own workflow solutions for their target consumer. However, in any workflow, this protection profile applies only to the part of the workflow that is performed by the redaction tool and only to the redaction functionality in that tool. Other functionality in the redaction tool, other tools used in the workflow, the organization's redaction policy as well as security requirements and security policies that apply to other parts of the workflow is beyond the scope of this protection profile.

## 1.5 TOE Scope

This PP is limited to the redaction of electronic documents defined in standards such as the series International Organization for Standards (ISO)/International Electrotechnical Commission (IEC)-29500 (Office OpeneXtensible Markup Language (XML), e.g. Microsoft Word, PowerPoint, and Excel documents) and ISO/IEC-32000 (PDF), or the definitive standard for a format. This PP applies to an interactive tool requiring the user to selectively review and redact information one document at a time. Mail guards, filters, and batch redaction tools are beyond the scope of this PP.

Requirements that apply to features such as administrative control over particular redaction settings, multi-person review prior to release, etc., are outside the scope of this PP. The TOE may have those features but is not required to have them and their use and enforcement is governed by the organization's redaction policy.

This PP covers the software functionality of the redaction process; it does not include requirements for how users should decide what to redact or other policy issues. Analysis of documents for covert data transfer is part of the decision-making process for what to redact and so occurs prior to the redaction itself. The requirements in this document are independent of requirements levied on document release by statute or the judiciary.

Data execution risks inherent in some file formats are beyond the scope of this PP. This PP assumes that scanning for such risks occurs prior to the document entering the redaction functionality of the TOE.

Documents to be redacted may contain objects that are vulnerable to steganography, such as images or video. Functional data such as scripts can contain strings or images that may not be accessible to the redaction tool. Analysis of such objects for attacks or covert data transfer will occur outside of the redaction process. An organization's security policy will determine whether such objects are released or redacted in their entirety.

# 2 Security Problem Definition

## 2.1 Threat

The security problem to be addressed by compliant TOEs is described by threats that might be targeted by the specific functionality of a Redaction Tool. Annex A: Supporting Tables presents the Security Problem Description (SPD) in a more “traditional” form. The following sections detail the problems that compliant TOEs will address; references to the “traditional” statements in Annex A are included.

**T.Clues_to_Original_Data**

A user or application may release redacted documents, where, the text or graphics placed in the redacted area by the TOE may contain clues to the nature of the original redacted information.

**T.Expose_Process_Data**

A user or application may release documents containing hidden elements that may contain information on process details unnecessary for release.

**T.Implementation_Flaws**

The TOE software may contain flaws that result in improper TOE operation.

**T.Inaccessible_Data**

Inaccessible files and/or files that are 'unhealthy' to open or modify can prevent proper inspection of metadata and elements that are selected for redaction.

**T.Obscured_Data**

A user or application may release documents containing data that is obscured via some well-known or proprietary process. Because the original data is still present, there is a risk that an adversary can reverse the process to reveal data deemed unreleasable for dissemination.

**T.Unredacted_Visible_Data**

A user or application may release documents containing visible objects or text that may contain information deemed unreleasable for dissemination outside of the data owner's organization.

**T.Unredacted_Hidden_Data**

A user or application may release documents containing hidden, extraneous, unrecognized or noninterpretable elements that may contain information deemed unreleasable for dissemination outside of the data owner's organization.

**T.User_Error**

Unintentional or accidental user errors can occur due to certain human factors (fatigue, drowsiness, illness) as well as certain environmental aspects that can distract a user (noise, excessive workload, coworkers).

**T.Unredacted_Release**

The TOE may fail to redact selected or hidden data elements that may contain information deemed unreleasable for dissemination outside of the data owner's organization.

## 2.2 Assumptions

This section of the security problem definition shows the assumptions that are made on the operational environment in order to be able to provide security functionality. If the TOE is placed in an operational environment that does not meet these assumptions, the TOE may not be able to provide all of its security functionality. Assumptions can be for physical, personnel and connectivity aspects of the operational environment.

**A.Platform**

The TOE relies upon a trustworthy computing platform for its execution. This includes the underlying operating system and the runtime environment it provides to the TOE.

**A.Proper User**

The user of the application software is not willfully negligent or hostile, and uses the software within compliance of a reasonable Enterprise security policy.

**A.Proper Admin**

The administrator of the application software is not careless, willfully negligent or hostile, and administers the software within compliance of a reasonable Enterprise security policy.

**A.Enterprise**

The configuration of the application software must adhere to the Enterprise security policy.

**A.Knowledge User**

The user is knowledgeable concerning document management and has appropriate training with the redaction tool. Part of this knowledge and training includes how to prepare a document for the redaction tool, e.g. resolve and turn off tracked changes prior to redaction, work with a copy of the document and preserve the original file, remove passwords and decrypt files, etc.

**A.Information Release Policy**

There is a redaction or information release policy in place for the organization which the user follows.

**A.Preservation of Document Layout**

The TOE will preserve the layout of the document.

# 3 Security Objectives

Compliant TOEs will provide security functionality that address threats to the TOE. The following sections provide a description of this functionality in light of the threats previously discussed that motivate its inclusion in compliant TOEs. The security functionality provided includes generation of reports of redacted items, validation of files designated for redaction, and removal of data and elements designated for redaction.

**O.Deep_Inspection**

The TOE will analyze the file content for metadata and elements, to include any that are purposely hidden or not immediately visible to the naked eye. This metadata and elements includes, but is not limited to those that are obstructed from view such as shapes on top of text, hidden objects (manual direct formatting or programmatically hidden), and text that is positioned off the margins, and/or is located in header and footer sections of the file.

**O.Failure**

The user must have assurance that the redaction is complete to avoid releasing the document prior to proper redaction. The TOE can fail redaction for several reasons. Failing in a secure state will assure the user that a partially redacted file will not be released.

**O.Hidden Data**

The TOE will provide the capability to find and remove hidden data elements in the document. The TOE will include the capability to automatically remove certain hidden data elements that are unnecessary for the functionality, display or printing of the document.

**O.Locate_and_Remove**

Complex file formats may store multiple instances of data in different locations within the file. The TOE must locate all instances of data to be redacted and completely remove this data from the file.

**O.No_New_Hidden_Data**

The TOE may not add hidden data to the redacted file without informing the user. This hidden data may include metadata, document management system tracking information, or other process related information.

**O.Remove**

The TOE will provide the capability to completely remove any data selected for redaction.

**O.Remove_References**

Complex file formats may contain multiple internal references to data to be redacted. The TOE will remove all such internal references.

**O.Replace**

During the redaction process the TOE will place text or graphics in the area of the redacted information to signify data has been redacted. This newly placed information cannot convey any information which may offer clues as to the nature of the original redacted information.

**O.Report**

As users employ redaction tools to remove elements from documents or other types of logical containers, confirmation that the elements selected by the user and the TOE were removed is necessary. Report generation is one means of providing users with feedback about what was redacted. This capability allows the user to call up a report upon request.

**O.Selected_Elements**

The TOE will provide the capability to select any type of document object in whole or in part for redaction.

**O.Validation**

The TOE will provide the capability to parse through the structure of the document and extraneous or unnecessary data without affecting the overall structure or functionality (e.g., display, print) of the document. The TOE will also provide the capability to parse different complex objects based on the format and simplify or remove those that it cannot interpret. In doing so, the TOE will produce a clean document for redaction without involving the user.

# 4 Security Functional Requirements

Some of the Security Functional Requirements included in this section are derived from Part 2 of the *Common Criteria for Information Technology Security Evaluation, Version 3.1, Revision 4*, with additional extended functional components.

Conventions

The CC defines operations on Security Functional Requirements: assignments, selections, assignments within selections and refinements. This document uses the following font conventions to identify the operations defined by the CC:

- Assignment: Indicated with *italicized* text;
- Refinement made by PP author: Indicated by the word "Refinement" in **bold text** after the element number with additional text in **bold** text and deletions with strikethroughs, if necessary;
- Selection: Indicated with underlined text;
- Assignment within a Selection: Indicated with *italicized and underlined* text;
- Iteration: Indicated by appending the iteration number in parenthesis, e.g., (1), (2), (3).

Explicitly stated SFRs are identified by having a label 'EXT' after the requirement name for TOE SFRs.

## 4.1 Report Generation and Review

**REP_GEN_EXT.1 Report Generation**

Hierarchical to: No other components.

Dependencies: None

REP_GEN_EXT.1.1 The TOE must be able to generate a report entry that contains metadata about each element that was redacted, including at least the following: the type of the element that was removed, the location if it was a visible element, and whether the element was selected by the user or removed automatically.

***Application Note:*** *The report can be a configurable feature that is only generated on user request. Location can be a page number, a cell number for a spreadsheet, or some other indication that allows the user to easily locate the visible element.*

***Assurance Activity:*** *The evaluator shall examine the TSS to ensure it describes the TOE's reporting feature and the metadata that is included for each report entry. The evaluator shall examine the operational guidance to ensure it contains instructions for the configuration of the reporting feature in accordance with this requirement. The evaluator shall create test files with specific elements to redact, apply the TOE to the test files, and observe that there is a report for each element expected to be redacted. This assurance activity can be done in conjunction with REP_RVW_EXT.1.*

**REP_RVW_EXT.1 Report Review**

Hierarchical to: No other components.

Dependencies: REP_GEN.1

REP_RVW_EXT.1.1 The TOE must allow the user to access a report of the data that was redacted.

***Application Note****: This can be satisfied with a dialog box or other simple list of items that were redacted. The report can be a configurable feature that is only generated on user request.*

***Assurance Activity****: The evaluator shall create test files with specific elements to redact, apply the TOE to the test files, and observe that there is a report entry for each element expected to be redacted.*

## 4.2 Validation Stage

**VAL_REM_EXT.1 Validation of Data**

Hierarchical to: No other components.

Dependencies: None

VAL_REM_EXT.1.1 The TOE must remove unrecognized data, unexpected data, and extraneous structural data.

***Application Note****: Structural data is extraneous if it is unnecessary for the printing or display of the document contents, or unnecessary for the functionality of the document.*

*Example - many formats include comments, e.g. PDF allows file format comments which are preceded by %. When these comments are unnecessary, unrelated to the printing or display of the content of the document, or provide no functionality whatsoever they must be removed.*

*Example – some formats expect a header structure starting at the first byte of a file, but a tool may be able to interpret a file where the header starts at a later byte by ignoring the data that precedes the header structure. In this case, the preceding data must be removed since it is unexpected.*

***Assurance Activity****: The evaluator shall create or acquire test files that contain unrecognized data, unexpected data, and extraneous structural data. The evaluator shall examine the files prior to redaction to identify the data. The evaluator shall apply the TOE but make no visible redactions and save the output files. The evaluator shall examine the output files, comparing it to the originals, to verify that the data has been removed.*

VAL_REM_EXT.1.2 The TOE must [selection: simplify, remove] any element which it cannot completely interpret.

***Application Note****: For example, if the tool cannot recurse through a stream with embedded OLE objects, it must convert the stream to a single layer image with no metadata or remove it. If the redaction tool cannot interpret or process temporal objects, it must remove the temporal object and replace it with a simplified object or other placeholder. If a stream of data is compressed, encoded or encrypted and the redaction tool cannot uncompress, decode or decrypt the data, the tool must delete the stream.*

***Assurance Activity****: The evaluator shall examine the TSS to ensure that it describes how the TOE handles data that it cannot completely interpret. The evaluator shall create or acquire test files with data that the TOE should not be able to completely interpret, apply the TOE and examine the output to verify that the TOE handled the data according to the requirement.*

## 4.3 Redaction Stage

### RED_SEL_EXT.1 Selected Redaction

Hierarchical to: No other components.

Dependencies: None

RED_SEL_EXT.1.1 The TOE must [selection: simplify, remove] any complex object, embedded object or graphic image which is selected for redaction.

***Application Note:*** *The selection may be of either the whole element or only part of the element. If part of an element is selected, only that part must be simplified or removed.*

***Assurance Activity****: The evaluator shall examine the TSS to ensure it describes in detail which complex objects can be simplified by the TOE and how they are simplified (e.g. whether the object or the whole page is converted to another format and what that format is). The TSS shall also list those complex objects or images that cannot be simplified and will be removed. The evaluator shall create or acquire test documents that contain complex objects and examine the documents to identify where those objects are in the format. The evaluator shall then apply the TOE and examine the output to verify that the objects have been simplified or removed. The evaluator shall test all objects that can be simplified as well as all objects that should be removed according to the TSS.*

*The evaluator shall also create or acquire test documents with complex objects that are not documented in the TSS, apply the TOE, and verify that those objects are removed from the document.*

**RED_DIN_EXT.1 Deep Inspection**

Hierarchical to: No other components.

Dependencies: None

RED_DIN_EXT.1.1 For each element of the file format that can contain its own metadata, other elements, or hidden data, the TOE must [selection: recurse through the element chain and apply the PP to each layer, simplify the element, redact the element].

***Application Note****: For example, JPG images can contain metadata called exif data. Some image formats can contain the same image in another format, such as raw which can contain a complete jpg version of the image. A complex object can contain other complex objects (e.g. Microsoft OLE). The tool must apply the requirements to each layer of every element and identify hidden/metadata not just at the top layer of the document but in each element and in* *all layers within that element. If the TOE cannot recurse through the layers, it must simplify the element at the top level.*

***Assurance Activity****: The evaluator shall examine the TSS to ensure it lists and describes the methods used to replace redacted elements that contain metadata, other elements, or hidden data. The evaluator shall ensure that the TSS' description complies with the requirement that each element is handled by either recursing through the element chain and applying the TOE to each layer, simplifying the element, or redacting the element. The evaluator shall create or acquire test files that contain elements that themselves contain other elements and hidden data. The evaluator shall examine the document to identify these elements in the structure, apply the TOE, and examine the output to verify that the elements were handled properly via either redaction or simplification in accordance with the requirement.*

### RED_RPL_EXT.1 Visible Space Replace

Hierarchical to: No other components.

Dependencies: None

RED_RPL_EXT.1.1 The TOE must replace the visible space of redacted content in such a way that the visible space conveys no information about the previous contents.

***Application Note****: A vendor may use several different methods to replace content, such as opaque blocks, text, whitespace or some other vendor-defined method. These methods must not convey information about the content being replaced. For example, if text is replaced with text, the replacement text must not indicate length of component words. Blocks of color used to replace parts of images must not show variations in intensity that could convey information about the image content.*

***Assurance Activity****: The evaluator shall examine the TSS to ensure it lists and describes the content used to replace redacted elements. The evaluator shall ensure that the TSS' description complies with the requirement to convey no information about the previous contents. The evaluator shall create or acquire a test file with an image, mark part of the image for redaction and apply the TOE, and examine the image in the output to verify that the visual appearance does not provide any indication of the content that was redacted. If the TOE allows text content to be replaced with text, the evaluator shall create or acquire a test file with some text as content, apply the TOE, and verify that the replacement text does not preserve word length or other identifying information that could allow recovery of the original content.*

### RED_REM_EXT.1 Removal of Redacted Data

Hierarchical to: No other components.

Dependencies: None

RED_REM_EXT.1.1 All data that is either selected by the user for redaction or identified by the TOE for redaction must be removed from the document.

***Application Note****: Selected content must be removed, not obscured by encryption, encoding, conversion to a proprietary format, or any other method.*

***Assurance Activity****: The evaluator shall examine the TSS to ensure it describes the removal of all data selected for redaction and verify that no encryption, encoding or proprietary process is used to obscure selected data. The evaluator shall ensure that the TSS' description complies with the requirement to remove all data selected by the user or identified by the TOE for redaction. The evaluator shall acquire or create test files that contain text, images and other elements. The evaluator shall examine the test files to locate the content in the format. The evaluator shall apply the TOE, marking some of the content for redaction, and examine the output to verify that the marked content was removed and not obscured through encryption, encoding, or conversion to a proprietary format.*

**RED_LOC_EXT.1 Redact Content from Every Location**

Hierarchical to: No other components.

Dependencies: None

RED_LOC_EXT.1.1 The TOE must remove redacted content from every location in the file format where it is stored.

***Assurance Activity****: The evaluator shall create or acquire test files that contain content in multiple places and examine the files to locate the content. The evaluator shall apply the TOE and examine the output to verify that it has been removed from every location.*

**RED_NND_EXT.1 No New Data Introduced by TOE**

Hierarchical to: No other components.

Dependencies: None

RED_NND_EXT.1 .1 The TOE itself must not introduce new hidden data that was not requested by the user without warning the user of the addition.

***Application Note****: If the redaction process changes the format of an object, such as converting a complex object to an image, the conversion must not introduce new metadata.*

*The TOE can modify or add structural data, including fonts, without alerting the user if the modification is necessary for the proper display or print of the file.*

***Assurance Activity****: The evaluator shall examine the TSS to ensure it describes the actions taken by the TOE when removing, simplifying, or redacting an element. If structural data is added, the TSS shall specify what structural data is added and the purpose of the structural data. If non-structural hidden data is added, the TSS shall detail the added hidden data and describe how the user is notified of the addition. The evaluator shall ensure that the TSS' description complies with the requirement to not introduce new hidden data, other than structural data, without warning the user. The evaluator shall create or acquire test files with complex objects or other elements and examine the files to locate those items in the structure. The examiner shall apply the TOE and examine the output to verify that no new hidden/metadata was introduced.*

**RED_OBJ_EXT.1 Removal of Objects and Corresponding References**

Hierarchical to: No other components.

Dependencies: None

RED_OBJ_EXT.1.1 The TOE must remove all references and indicators in the structural data to objects that are completely redacted by the TOE.

***Application Note:*** *In some formats, there are references in the structural data to objects, such as a name dictionary in PDF. If an object in a PDF document, such as an image, is completely redacted (i.e. the user has selected the entire image to be redacted), then not only must the image data be removed, but references to it in a name dictionary as well as all structural references to the image must be removed. If only part of the object is selected for redaction, then the references necessarily remain in the file since the object remains in the file.*

***Assurance Activity****: The evaluator shall examine the TSS to ensure its description of the removal of redacted objects includes the removal of all references and indicators to the redacted objects in conformance with the requirement. The evaluator shall create or acquire test files that contain objects and examine the files to locate these objects in the file format and all references to them in the structural data. The evaluator shall apply the TOE and select elements for complete redaction. The evaluator shall examine the output files to verify that the objects and all references to them have been redacted.*

**RED_RIP_EXT.1 Residual Information Removal**

Hierarchical to: No other components.

Dependencies: None

RED_RIP_EXT.1.1 The TOE must automatically remove all remnant data, undo buffers, tracked changes buffers, multiple versions of the same object, and any buffer or cache type container of data.

***Application Note****: The user does not have to select this data for removal.*

***Assurance Activity****: The evaluator shall examine the TSS to ensure it specifies the residual data and objects (e.g., remnant data, undo buffers, tracked changes buffers, multiple versions of the same object, and any buffer or cache type data container) that the TOE will remove from files without any user interaction. The evaluator shall ensure that the TSS' description complies with the requirement to automatically remove all such data. The evaluator shall create or acquire test files that contain the types of data described in the requirement and examine the files to locate the data. The evaluator shall apply the TOE and not select anything for redaction, and examine the files to verify that this data has been removed automatically.*

**FPT_FLS.1 Failure with Preservation of Secure State**

Hierarchical to: No other components.

Dependencies: None

FPT_FLS.1.1 The TSF shall preserve a secure state when the following types of failures occur: [*assignment: any failure*].

***Application Note****: If the redaction functionality fails for any reason, the TOE must not produce a partially redacted file.*

***Assurance Activity****: The evaluator shall examine the TSS to ensure it describes what actions the TOE performs upon any failure. The evaluator shall ensure that the TSS' description complies with the requirement to not produce a partially redacted file. The evaluator shall create or acquire test files that cause the TOE to fail and observe that the TOE fails and does not produce partially redacted files.*

## 4.4 User Experience

**RED_ID_EXT.1 Identification of Data**

Hierarchical to: No other components.

Dependencies:

RED_ID_EXT.1.1 The TOE must identify all hidden data in the document, except remnant data and undo or tracked change buffers, and allow the user to review and select each hidden data element individually for redaction.

***Application Note**: Remnant data and undo or tracked change buffers are removed automatically according to RED_RIP_EXT.1. If the file or part of the file is encrypted, the TOE will have to reject the file or decrypt it so that the tool and the user can review the hidden data.*

***Assurance Activity**: The evaluator shall examine the TSS to ensure it specifies the hidden data that it identifies and allows the user to select for redaction. The evaluator shall ensure that the TSS' description complies with the requirement for the TOE to identify all hidden data and allow the user to review and select each hidden data element for redaction. The evaluator shall create test documents with various types of hidden data apply the TOE, and verify that it identifies each expected element and allows the user to select and redact each.*

RED_ID_EXT.1.2 The TOE must identify all obscured data and must [selection: remove the obscured data automatically, allow the user to redact the obscured data].

***Application Note**: Obscured data is anything that could be visible but is obscured in some way, such as the cropped portion of an image or graphic. While the user sees only the portion of the graphic in the view container, the document contains the data in the cropped area. The tool must either remove the obscured data automatically or give the user the choice to remove or retain the obscured area.*

***Assurance Activity**: The evaluator shall examine the TSS to ensure it describes how the TOE handles all obscured data. The evaluator shall ensure that the TSS' description complies with the requirement that all obscured data is identified and either removed automatically or redacted by the user. The evaluator shall create test documents with various forms of obscured data, apply the TOE, and verify that the tool identifies the obscured data and either removes the obscured data automatically or gives the user the choice to remove or retain the obscured data.*

RED_ID_EXT.1.3 The TOE must identify images where the visible representation is reduced in size or resolution from the representation stored in the file format and must [selection: replace the stored data with the visible representation automatically, allow the user to [selection: replace the stored data with the visible representation, leave the image unaltered]].

***Assurance Activity**: The evaluator shall create a test document with an image that is stored in a larger size and resolution than the visible image and apply the TOE without selecting the image for redaction. The evaluator shall verify that the TOE either*

- *gives the user a choice to retain the image unaltered or replace the stored data with the visible data. For the choice to replace the image, the evaluator shall examine the output file to locate the image and verify its size and resolution, or extract the image with an appropriate tool for that format and compare the size and resolution of the extracted image to the visible image to determine if they match.*
- *resizes the stored image. To determine this, the evaluator shall examine the output file to locate the image and verify its size and resolution, or extract the image with an appropriate tool for that format and compare the size and resolution of the extracted image to the visible image to determine if they match.*

### RED_RVW_EXT.1 Element Review

Hierarchical to: No other components.

Dependencies: None

RED_RVW_EXT.1.1 The TOE must allow the user to review and select each element of visible data in whole or in part for redaction.

***Application Note**: If the file or part of the file is encrypted, the TOE will have to reject the file or decrypt it so that the user can review the data.*

***Assurance Activity**: The evaluator shall create test documents that contain images, text, and complex objects, apply the TOE and verify that each element is selectable for redaction in whole or in part.*

### RED_ALR_EXT.1 Redaction Failure Notification

Hierarchical to: No other components.

Dependencies: None

RED_ALR_EXT.1.1 The TOE must make the user aware when redaction fails for any reason.

***Assurance Activity**: The evaluator shall examine the TSS to ensure it describes how the TOE notifies the user when redaction fails. The evaluator shall ensure that the TSS' description complies with the requirement that the user is notified when redaction fails for any reason. The* *evaluator shall acquire or create test files that should fail the redaction, apply the TOE and verify that the TOE alerts the user that the redaction failed.*

# 5 Security Assurance Requirements

The Security Objectives for the TOE in Section 3 were constructed to address threats identified in Section 2. The Security Functional Requirements (SFRs) in Section 4.2 are a formal instantiation of the Security Objectives. The PP draws from the CC Security Assurance Requirements (SARs) to frame the extent to which the evaluator assesses the documentation applicable for the evaluation and performs independent testing.

While this section contains the complete set of SARs from the CC, the Assurance Activities to be performed by an evaluator are detailed both in Section 4.2 as well as in this section.

The general model for evaluating TOEs against STs written to conform to this PP is as follows:

After the ST has been approved for evaluation, the Common Criteria Testing Laboratory (CCTL) will obtain the TOE, supporting IT environment, and the administrative guides for the TOE. The Assurance Activities listed in the ST (which will be refined by the CCTL to be TOE-specific, either within the ST or in a separate document) will then be performed by the CCTL. The results of these activities will be documented and presented (along with the administrative guidance used) for validation.

For each assurance family, "Developer Notes" are provided on the developer action elements to clarify what, if any, additional documentation/activity needs to be provided by the developer. For the content/presentation and evaluator activity elements, additional assurance activities are described as a whole for the family, rather than for each element. Additionally, the assurance activities described in this section are complementary to those specified in Section 4.2.

The TOE security assurance requirements, summarized in Table 2, identify the management and evaluative activities required to address the threats identified in Section 2 of this PP.

**Table 2: TOE Security Assurance Requirements**

| **Assurance Class** | Assurance Components | Assurance Components Description |
|---|---|---|
| Development | ADV_FSP.1 | Basic Functional Specification |
| Guidance Documents | AGD_OPE.1 | Operational user guidance |
| | AGD_PRE.1 | Preparative User guidance |

| | | |
|---|---|---|
| Tests | ATE_IND.1 | Independent testing - conformance |
| Vulnerability Assessment | AVA_VAN.1 | Vulnerability analysis |
| Life Cycle Support | ALC_CMC.1 | Labeling of the TOE |
| | ALC_CMS.1 | TOE CM coverage |

### 5.1.1 Class ADV: Development

The information about the TOE is contained in the guidance documentation available to the end user as well as the TOE Summary Specification (TSS) portion of the ST. While it is not required that the TOE developer write the TSS, the TOE developer must concur with the description of the product that is contained in the TSS as it relates to the functional requirements. The Assurance Activities contained in Section 4.2, should provide the ST authors with sufficient information to determine the appropriate content for the TSS section.

#### *5.1.1.1 ADV_FSP.1 Basic Functional Specification*

The functional specification describes the Target Security Functions Interfaces (TSFIs). It is not necessary to have a formal or complete specification of these interfaces. Additionally, because TOEs conforming to this PP will necessarily have interfaces to the Operational Environment that are not directly invokable by TOE users, there is little point specifying that such interfaces be described in and of themselves since only indirect testing of such interfaces may be possible. For this PP, the activities for this family should focus on understanding the interfaces presented in the TSS in response to the functional requirements and the interfaces presented in the AGD documentation. No additional "functional specification" documentation is necessary to satisfy the assurance activities specified.

The interfaces that need to be evaluated are characterized through the information needed to perform the assurance activities listed, rather than as an independent, abstract list.

**Developer action elements:**

ADV_FSP.1.1D The developer shall provide a functional specification.

ADV_FSP.1.2D The developer shall provide a tracing from the functional specification to the SFRs.

Developer Note: As indicated in the introduction to this section, the functional specification is comprised of the information contained in the AGD_OPE and AGD_PRE documentation, coupled with the information provided in the TSS of the ST. The assurance activities in the functional requirements point to evidence that should exist in the documentation and TSS section; since these are directly associated with the SFRs, the tracing in element ADV_FSP.1.2D is implicitly already done and no additional documentation is necessary.

**Content and presentation elements:**

ADV_FSP.1.1C The functional specification shall describe the purpose and method of use for each SFR-enforcing and SFR-supporting TSFI.

ADV_FSP.1.2C The functional specification shall identify all parameters associated with each SFR-enforcing and SFR-supporting TSFI.

ADV_FSP.1.3C The functional specification shall provide rationale for the implicit categorization of interfaces as SFR-non-interfering.

ADV_FSP.1.4C The tracing shall demonstrate that the SFRs trace to TSFIs in the functional specification.

**Evaluator action elements:**

ADV_ FSP.1.1E The evaluator *shall confirm* that the information provided meets all requirements for content and presentation of evidence.

ADV_ FSP.1.2E The evaluator shall determine that the functional specification is an accurate and complete instantiation of the SFRs.

***Assurance Activities:***

*There are no specific assurance activities associated with these SARs. The functional specification documentation is provided to support the evaluation activities described in Section 4.2, and other activities described for AGD, ATE, and AVA SARs. The requirements on the content of the functional specification information is implicitly assessed by virtue of the other assurance activities being performed; if the evaluator is unable to perform an activity because the there is insufficient interface information, then an adequate functional specification has not been provided.*

### 5.1.2 Class AGD: Guidance Documents

The guidance documents will be provided with the developer's security target. Guidance must include a description of how the authorized user verifies that the Operational Environment can fulfill its role for the security functionality. The documentation should be in an informal style and readable by an authorized user.

Guidance must be provided for every operational environment that the product supports as claimed in the ST. This guidance includes

- instructions to successfully install the TOE in that environment; and
- instructions to manage the security of the TOE as a product and as a component of the larger operational environment.

Guidance pertaining to particular security functionality is also provided; specific requirements on such guidance are contained in the assurance activities specified in Section 4.2.

#### *5.1.2.1 AGD_OPE.1 Operational User Guidance*

**Developer action elements:**

AGD_OPE.1.1D The developer shall provide operational user guidance.

Developer Note: Rather than repeat information here, the developer should review the assurance activities for this component to ascertain the specifics of the guidance that the evaluator will be checking for. This will provide the necessary information for the preparation of acceptable guidance.

**Content and presentation elements:**

AGD_OPE.1.1C The operational user guidance shall describe, for each user role, the user-accessible functions and privileges that should be controlled in a secure processing environment, including appropriate warnings.

AGD_OPE.1.2C The operational user guidance shall describe, for each user role, how to use the available interfaces provided by the TOE in a secure manner.

AGD_OPE.1.3C The operational user guidance shall describe, for each user role, the available functions and interfaces, in particular all security parameters under the control of the user, indicating secure values as appropriate.

AGD_OPE.1.4C The operational user guidance shall, for each user role, clearly present each type of security-relevant event relative to the user-accessible functions that need to be performed, including changing the security characteristics of entities under the control of the TSF.

AGD_OPE.1.5C The operational user guidance shall identify all possible modes of operation of the TOE (including operation following failure or operational error), their consequences, and implications for maintaining secure operation.

AGD_OPE.1.6C The operational user guidance shall, for each user role, describe the security measures to be followed in order to fulfill the security objectives for the operational environment as described in the ST.

AGD_OPE.1.7C The operational user guidance shall be clear and reasonable.

**Evaluator action elements:**

AGD_OPE.1.1E The evaluator *shall confirm* that the information provided meets all requirements for content and presentation of evidence.

***Assurance Activities:***

*Some of the contents of the operational guidance will be verified by the assurance activities in Section 4.2 and evaluation of the TOE according to the CEM. The following additional information is also required.*

*The operational guidance shall at a minimum list the processes that comprise the TOE in its evaluated configuration. For each process listed, the administrative guidance will contain a short (e.g., one- or two-line) description of the process' function, and the privilege with which the process runs. "Privilege" includes the hardware privilege level (e.g., ring 0, ring 1), any* *software privileges specifically associated with the process, and the privileges associated with the user role the process runs as or under.*

### *5.1.2.2 AGD_PRE.1 Preparative Procedures*

**Developer action elements:**

AGD_PRE.1.1D The developer shall provide the TOE, including its preparative procedures.

Developer Note: As with the operational guidance, the developer should look to the assurance activities to determine the required content with respect to preparative procedures.

**Content and presentation elements:**

AGD_ PRE.1.1C The preparative procedures shall describe all the steps necessary for secure acceptance of the delivered TOE in accordance with the developer's delivery procedures.

AGD_ PRE.1.2C The preparative procedures shall describe all the steps necessary for secure installation of the TOE and for the secure preparation of the operational environment in accordance with the security objectives for the operational environment as described in the ST.

**Evaluator action elements:**

AGD_ PRE.1.1E The evaluator *shall confirm* that the information provided meets all requirements for content and presentation of evidence.

AGD_ PRE.1.2E The evaluator *shall apply* the preparative procedures to confirm that the TOE can be prepared securely for operation.

***Assurance Activities:***

*As indicated in the introduction above, there are significant expectations with respect to the documentation—especially when configuring the operational environment to support TOE* *functional requirements. The evaluator shall check to ensure that the guidance provided for the TOE adequately addresses all platforms claimed for the TOE in the ST.*

### 5.1.3 Class ATE: Tests

Testing is specified for functional aspects of the system as well as aspects that take advantage of design or implementation weaknesses. The former is done through the ATE_IND family, while the latter is through the AVA_VAN family. At the assurance level specified in this PP, testing is based on advertised functionality and interfaces with dependency on the availability of design information. One of the primary outputs of the evaluation process is the test report as specified in the following requirements.

#### *5.1.3.1 ATE_IND.1 Independent Testing - Conformance*

Testing is performed to confirm the functionality described in the TSS as well as the administrative (including configuration and operational) documentation provided. The focus of the testing is to confirm that the requirements specified in Section 4.2 are being met, although some additional testing is specified for SARs in Section 4.3. The Assurance Activities identify the additional testing activities associated with these components. The evaluator produces a test report documenting the plan for and results of testing, as well as coverage arguments focused on the platform/TOE combinations that are claiming conformance to this PP.

**Developer action elements:**

ATE_IND.1.1D The developer shall provide the TOE for testing.

**Content and presentation elements:**

ATE_IND.1.1C The TOE shall be suitable for testing.

**Evaluator action elements:**

ATE_IND.1.1E The evaluator *shall confirm* that the information provided meets all requirements for content and presentation of evidence.

ATE_IND.1.2E The evaluator *shall test* a subset of the TSF to confirm that the TSF operates as specified.

***Assurance Activities:***

*The evaluator shall prepare a test plan and report documenting the testing aspects of the system. The test plan covers all of the testing actions contained in the CEM and the body of this PP's Assurance Activities. While it is not necessary to have one test case per test listed in an Assurance Activity, the evaluator must document in the test plan that each applicable testing requirement in the ST is covered.*

*The test plan identifies the platforms to be tested, and for those platforms not included in the test plan but included in the ST, the test plan provides a justification for not testing the platforms. This justification must address the differences between the tested platforms and the untested platforms, and make an argument that the differences do not affect the testing to be performed. It is not sufficient to merely assert that the differences have no affect; rationale must be provided. If all platforms claimed in the ST are tested, then no rationale is necessary.*

*The test plan describes the composition of each platform to be tested, and any setup that is necessary beyond what is contained in the AGD documentation. It should be noted that the evaluator is expected to follow the AGD documentation for installation and setup of each platform either as part of a test or as a standard pre-test condition. This may include special test drivers or tools. For each driver or tool, an argument (not just an assertion) should be provided that the driver or tool will not adversely affect the performance of the functionality by the TOE and its platform.*

*The test plan identifies high-level test objectives as well as the test procedures to be followed to achieve those objectives. These procedures include expected results. The test report (which could just be an annotated version of the test plan) details the activities that took place when the test procedures were executed, and includes the actual results of the tests. This shall be a cumulative account, so if there was a test run that resulted in a failure; a fix installed; and then a successful re-run of the test, the report would show a "fail" and "pass" result (and the supporting details), and not just the "pass" result.*

### 5.1.4 Class AVA: Vulnerability Assessment

For the first generation of this protection profile, the evaluation lab is expected to survey open sources to discover what vulnerabilities have been discovered in these types of products. In most cases, these vulnerabilities will require sophistication beyond that of a basic attacker. Until penetration tools are created and uniformly distributed to the evaluation labs, the evaluator will not be expected to test for these vulnerabilities in the TOE. The labs will be expected to comment on the likelihood of these vulnerabilities given the documentation provided by the vendor. This information will be used in the development of penetration testing tools and for the development of future protection profiles.

#### *5.1.4.1 AVA_VAN.1 Vulnerability Survey*

**Developer action elements:**

AVA_VAN.1.1D The developer shall provide the TOE for testing.

**Content and presentation elements:**

AVA_VAN.1.1C The TOE shall be suitable for testing.

**Evaluator action elements:**

AVA_VAN.1.1E The evaluator *shall confirm* that the information provided meets all requirements for content and presentation of evidence.

AVA_VAN.1.2E The evaluator *shall perform* a search of public domain sources to identify potential vulnerabilities in the TOE.

AVA_VAN.1.3E The evaluator *shall conduct* penetration testing, based on the identified potential vulnerabilities, to determine that the TOE is resistant to attacks performed by an attacker possessing Basic attack potential.

***Assurance Activities:***

*As with ATE_IND, the evaluator shall generate a report to document their findings with respect to this requirement. This report could physically be part of the overall test report mentioned in ATE_IND, or a separate document. The evaluator performs a search of public information to determine the vulnerabilities that have been found in redaction tools in general, as well as those that pertain to the particular TOE. The evaluator documents the sources consulted and the vulnerabilities found in the report. For each vulnerability found, the evaluator either provides a rationale with respect to its non-applicability, or the evaluator formulates a test (using the guidelines provided in ATE_IND) to confirm the vulnerability, if suitable. Suitability is determined by assessing the attack vector needed to take advantage of the vulnerability. For example, if the vulnerability can be detected by pressing a key combination on boot-up, a test would be suitable at the assurance level of this PP. If exploiting the vulnerability requires expert* *skills and an electron microscope, for instance, then a test would not be suitable and an appropriate justification would be formulated.*

### 5.1.5 Class ALC: Life-cycle Support

At the assurance level provided for TOEs conformant to this PP, life-cycle support is limited to end-user-visible aspects of the life-cycle, rather than an examination of the TOE vendor's development and configuration management process. This is not meant to diminish the critical role that a developer's practices play in contributing to the overall trustworthiness of a product; rather, it's a reflection on the information to be made available for evaluation at this assurance level.

#### *5.1.5.1 ALC_CMC.1 Labeling of the TOE*

This component is targeted at identifying the TOE such that it can be distinguished from other products or versions from the same vendor and can be easily specified when being procured by an end user.

**Developer action elements:**

ALC_CMC.1.1D The developer shall provide the TOE and a reference for the TOE.

**Content and presentation elements:**

ALC_CMC.1.1C The TOE shall be labeled with its unique reference.

**Evaluator action elements:**

ALC_CMC.2.1E The evaluator *shall confirm* that the information provided meets all requirements for content and presentation of evidence.

***Assurance Activities:***

*The evaluator shall check the ST to ensure that it contains an identifier (such as a product name/version number) that specifically identifies the version that meets the requirements of the ST. Further, the evaluator shall check the AGD guidance and TOE samples received for testing to ensure that the version number is consistent with that in the ST. If the vendor maintains a web* *site advertising the TOE, the evaluator shall examine the information on the web site to ensure that the information in the ST is sufficient to distinguish the product.*

#### *5.1.5.2 ALC_CMS.1 TOE CM Coverage*

Given the scope of the TOE and its associated evaluation evidence requirements, this component's assurance activities are covered by the assurance activities listed for ALC_CMC.1.

**Developer action elements:**

ALC_CMS.2.1D The developer shall provide a configuration list for the TOE.

**Content and presentation elements:**

ALC_CMS.2.1C The configuration list shall include the following: the TOE itself; and the evaluation evidence required by the SARs.

ALC_CMS.2.2C The configuration list shall uniquely identify the configuration items.

**Evaluator action elements:**

ALC_CMS.2.1E The evaluator *shall confirm* that the information provided meets all requirements for content and presentation of evidence.

***Assurance Activities:***

*The "evaluation evidence required by the SARs" in this PP is limited to the information in the ST coupled with the guidance provided to administrators and users under the AGD requirements. By ensuring that the TOE is specifically identified and that this identification is consistent in the ST and in the AGD guidance (as done in the assurance activity for ALC_CMC.1), the evaluator implicitly confirms the information required by this component.*

# RATIONALE

The rationale tracing the threats to the objectives and the objectives to the requirements is contained in the prose in Sections 2.0 and 3.0. The only outstanding mappings are those for the Assumptions and Organizational Security Policies; those are contained in Annex A below.

### Annex A: Supporting Tables

In this Protection Profile, the focus in the initial sections of the document is to use a narrative presentation in an attempt to increase the overall understandability of the threats to Redaction Tools; the methods used to mitigate those threats; and the extent of the mitigation achieved by compliant TOEs. This presentation style does not readily lend itself to a formalized evaluation activity, so this Annex contains the tabular artifacts that can be used for the evaluation activities associated with this document.

### Assumptions

The specific conditions listed in the following subsections are assumed to exist in the TOE's Operational Environment. These assumptions include both practical realities in the development of the TOE security requirements and the essential environmental conditions on the use of the TOE.

**Table 1: TOE Assumptions**

| Assumption Name | Assumption Definition |
|---|---|
| A.Platform | The TOE relies upon a trustworthy computing platform for its execution. This includes the underlying operating system and the runtime environment it provides to the TOE. |
| A.Proper_User | The user of the application software is not willfully negligent or hostile, and uses the software within compliance of a reasonable Enterprise security policy. |
| A.Proper_Admin | The administrator of the application software is not careless, willfully negligent or hostile, and administers the software within compliance of a reasonable Enterprise security policy. |
| A.Enterprise | The configuration of the application software must adhere to the Enterprise security policy. |
| A.Knowledge_User | The user is knowledgeable concerning document management and has appropriate training with the redaction tool. Part of this knowledge and training |

| | |
|---|---|
| | includes how to prepare a document for the redaction tool, e.g. resolve and turn off tracked changes prior to redaction, work with a copy of the document and preserve the original file, remove passwords and decrypt files, etc. |
| A.Information_Release_Policy | There is a redaction or information release policy in place for the organization which the user follows. |
| A.Preservation_of_Document_Layout | The TOE will preserve the layout of the document. |

**Threats**

The following threats should be integrated into the threats that are specific to the technology by the PP authors when including the requirements described in this document. Modifications, omissions, and additions to the requirements may impact this list, so the PP author should modify or delete these threats as appropriate.

**Table 2: Threats**

| Threat Name | Threat Definition |
|---|---|
| T.Clues_to_Original_Data | A user or application may release redacted documents; where, the text or graphics placed in the redacted area by the TOE may contain clues to the nature of the original redacted information. |
| T.Expose_Process_Data | A user or application may release documents containing hidden elements that may contain information on process details unnecessary for release. |
| T.Implementation_Flaws | The TOE may contain flaws that result in improper TOE operation |
| T.Inaccessible_Data | Inaccessible files and/or files that are 'unhealthy' to open or modify can prevent proper inspection of metadata and elements that are selected for redaction. |
| T.Obscured_Data | A user or application may release documents containing data that is obscured via some well-known or proprietary process. Because the original data is still present, there is a risk that an adversary can reverse the process to reveal data deemed unreleasable for dissemination. |
| T.Unredacted_Visible_Data | A user or application may release documents containing visible objects or text that may contain information deemed unreleasable for dissemination outside of the data owner’s organization. |

| | |
|---|---|
| T.Unredacted_Hidden_Data | A user or application may release documents containing hidden elements that may contain information deemed unreleasable for dissemination outside of the data owner’s organization. |
| T.User_Error | Unintentional or accidental user errors can occur due to certain human factors (fatigue, drowsiness, illness) as well as certain environmental aspects that can distract a user (noise, excessive workload, coworkers). |
| T.Unredacted_Release | The TOE may fail to redact selected or hidden data elements that may contain information deemed unreleasable for dissemination outside of the data owner’s organization. |

### Security Objectives for the TOE

**Table 3: Security Objectives for the TOE**

| TOE Security Objective | TOE Objective Definition |
|---|---|
| O.Deep_Inspection | The TOE will analyze the file content for metadata and elements, to include any that are purposely hidden or not immediately visible to the naked eye. This metadata and elements includes, but is not limited to those that are obstructed from view such as shapes on top of text, hidden objects (manual direct formatting or programmatically hidden), and text that is positioned off the margins, and/or is located in header and footer sections of the file. |
| O.Failure | The user must have assurance that the redaction is complete to avoid releasing the document prior to proper redaction. The TOE can fail redaction for several reasons. Failing in a secure state will assure the user that a partially redacted file will not be released. |
| O.Hidden Data | The TOE will provide the capability to find and remove hidden data elements in the document. The TOE will include the capability to automatically remove certain hidden data elements that are unnecessary for the functionality, display or printing of the document. |
| O.Locate_and_Remove | Complex file formats may store multiple instances of data in different locations within the file. The TOE must locate all instances of data to be redacted and completely remove this data from the file. |

| | |
|---|---|
| O.No_New_Hidden_Data | The TOE may not add hidden data to the redacted file without informing the user.  This hidden data may include metadata, document management system tracking information, or other process related information. |
| O.Remove | The TOE will provide the capability to completely remove any data selected for redaction. |
| O.Remove_References | Complex file formats may contain multiple internal references to data to be redacted.  The TOE will remove all such internal references. |
| O.Replace | During the redaction process the TOE will place text or graphics in the area of the redacted information to signify data has been redacted. This newly placed information cannot convey any information which may offer clues as to the nature of the original redacted information. |
| O.Report | As users employ redaction tools to remove elements from documents or other types of logical containers, confirmation that the elements selected by the user and the TOE were removed is necessary. Report generation is one means of providing users with feedback about what was redacted. This capability allows the user to call up a report upon request. |
| O.Selected_Elements | The TOE will provide the capability to select any type of document object in whole or in part for redaction. |
| O.Validation | The TOE will provide the capability to parse through the structure of the document and extraneous or unnecessary data without affecting the overall structure or functionality (e.g., display, print) of the document. The TOE will also provide the capability to parse different complex objects based on the format and simplify or remove those that it cannot interpret. In doing so, the TOE will produce a clean document for redaction without involving the user. |

### Security Threats to Security Objectives

The following table contains a mapping of Security Threats to Objectives for the TOE.

**Table 4: Security Threats to Objectives Mapping**

| Threat | Security Objectives |
|---|---|
| T.Clues_to_Original_Data | O.Replace;<br>O.Remove_References |
| T.Expose_Process_Data | O.No_New_Hidden_Data |
| T.Implementation_Flaws | O.Report |
| T.Inaccessible_Data | O.Deep_Inspection |
| T.Obscured_Data | O.Remove |
| T.Unredacted_Visible_Data | O.Failure;<br>O.Selected_Elements |
| T.Unredacted_Hidden_Data | O.Locate_and_Remove;<br>O.Selected_Elements;<br>O.Deep_Inspection<br>O.Hidden Data<br>O.Failure<br>O.Validation |
| T.User_Error | O.Report |
| T.Unredacted Release | O.Failure |

Security Objectives to Security Requirements

The following table contains a mapping of Security Objectives for the TOE to Security Functional Requirements.

**Table 5: Security Objectives to Requirements**

| TOE Security Objective | Security Functional Requirements |
|---|---|
| O.Deep_Inspection | RED_DIN_EXT.1 |
| O.Failure | FPT_FLS_EXT.1;RED_ALR_EXT.1 |
| O.Hidden_Data | RED_RIP_EXT.1.1; RED_ID_EXT.1.1; RED_ID_EXT.1.2;<br>RED_ID_EXT.1.4 |

| | |
|---|---|
| O.Locate_and_Remove | RED_LOC_EXT.1.1 |
| O.No_New_Hidden_Data | RED_NND_EXT.1 .1 |
| O.Remove | RED_REM_EXT.1 |
| O.Remove_References | RED_OBJ_EXT.1.1 |
| O.Replace | RED_RPL_EXT.1.1 |
| O.Report | REP_GEN_EXT.1.1; |
| O.Selected_Elements | RED_SEL_EXT.1.1; RED_RVW_EXT.1.1 |
| O.Validation | VAL_PAR_EXT.1.1 |

## Annex B: Optional Requirements

As indicated in the introduction to this PP, the baseline requirements (those that must be performed by the TOE or its underlying platform) are contained in the body of this PP. Additionally, there are three other types of requirements specified in Annexes B, C, and D.

The first type (in this Annex) is requirements that can be included in the ST, but do not have to be in order for a TOE to claim conformance to this PP. The second type (in Annex C) is requirements based on selections in the body of the PP: if certain selections are made, then additional requirements in that annex will need to be included. The third type (in Annex D) is components that are not required in order to conform to this PP, but will be included in the baseline requirements in future versions of this PP, so adoption by redaction tool vendors is encouraged. Note that the ST author is responsible for ensuring that requirements that may be associated with those in Annex B, Annex C, and/or Annex D but are not listed (e.g., FMT-type requirements) are also included in the ST.

*No optional requirements have been identified at this time.*

## Annex C: Selection-Based Requirements

In some cases when certain selections are made in SFRs, additional requirements in may need to be included.

Note that minor adjustments to the narrative information in the beginning of the ST may be required depending on the selections performed. Additionally, depending on the requirements selected, the appropriate information from Section C.2 *Auditable Events* will need to be added to the auditable events table in the ST.

*No Selection-Based items have been identified at this time.*

## Annex D: Objective Requirements

As indicated in the introduction to this PP, the baseline requirements (those that must be performed by the TOE or its underlying platform) are contained in the body of this PP. There are additional requirements that specify security functionality that is desirable and these requirements are contained in this Annex. It is expected that these requirements will transition from objective requirements to baseline requirements in future versions of this PP.

At any time these may be included in the ST such that the TOE is still conformant to this PP.

*No objective items have been identified at this time.*

## Annex E: Glossary and Acronyms

### Technical Definitions used in This Document

| Term | Meaning |
|---|---|
| EXIF | Exchangeable Image File Format |
| IEC | International Electrotechnical Commission |
| ISO | International Organization of Standards |
| JPG (JPEG) | Joint Photographic Experts Group |
| OLE | Object Linked and Embedding |
| PDF | Portable Document Format |
| TIFF | Tagged Image File Format |
| URL | Uniform Resource Locator |
| WYSIWYG | What You See is What You Get |
| XML | eXtensible Markup Language |

### Common Criteria Definitions

| Term | Meaning |
|---|---|
| Assurance | Grounds for confidence that a TOE meets the SFRs [CC1]. |
| CC | Common Criteria |
| PP | Protection Profile |
| SAR | Security Assurance Requirement |
| SFR | Security Functional Requirement |
| Security target (ST) | Implementation-dependent statement of security needs for a specific identified TOE. |
| Target of | A set of software, firmware and/or hardware possibly accompanied |

| | |
|---|---|
| evaluation (TOE) | by guidance. [CC1] |
| TOE security functionality (TSF) | Combined functionality of all hardware, software, and firmware of a TOE that must be relied upon for the correct enforcement of the SFRs. |
| TOE summary specification (TSS) | Documentation which provides evaluators with a description of the implementation of SFRs in the TOE. |