

Adobe Systems Inc.

IDPF Technical Note

Simple content protection for embedded fonts in OCF-packaged OPF documents

OCF [1] is a technology which is well-suited to package OPF/OPS-compliant [2] electronic publications. Since OCF is fundamentally a zip file, commonly available zip tools can be used to extract any unencrypted content stream from the package. In fact, on some systems, the content of the zip file will look like the content of the file system folder. While ability to do this is quite useful, it can pose a problem for an author of the publication who wishes to use a third-party resource in his publication, in particular, a third-party font. Many commercial fonts allow embedding, but embedding a font implies making it an integral part of the publication, not sending it alongside. Even though it could be argued that including the font file in the zip package is embedding, the more realistic position would be that zipping several files together is not any different than placing them in the same folder. This uncertainty can undermine the otherwise very useful font embedding capability that OPF/OPS. Adobe Systems Inc. intends to support the simple content protection method described here to avoid this uncertainty and encourages other vendors to support it as well.

When simple content protection method is used, all embedded fonts in the publication should be “mangled” in the way that makes them only usable for the publication in which they are included. Technically, “mangling” is done by applying XOR operation byte-by-byte to the first 1024 bytes of the font stream and the bytes from the mangling key. Mangling key is a big-endian binary form (16 bytes) of the first UUID URN-based unique identifier [3] in the publication’s OPF file. It is an error if such identifier is missing in the OPF file. When the bytes in the key are exhausted, the process should start again reading key bytes from the beginning of the key.

Section 3.5.5 of the OCF specification describes an encryption.xml file that can be included in the package to describe encryption for the individual files in the container. Simple content protection method leverages this feature to specify font mangling as a form of encryption. (While other resources in the publication can be mangled using this method as well, this is not recommended). Each mangled resource in the publication must be listed in the encryption.xml file. The algorithm URI that must be used for each resource is “http://ns.adobe.com/pdf/enc#RC”. Since the key is implicit to the algorithm, no encryption key needs to be specified. Here is a sample encryption.xml file:

```
<encryption xmlns="urn:oasis:names:tc:opendocument:xmlns:container">
  <EncryptedData xmlns:enc="http://www.w3.org/2001/04/xmlenc#">
    <EncryptionMethod Algorithm="http://ns.adobe.com/pdf/enc#RC"/>
    <CipherData>
      <CipherReference URI="OEBPS/Fonts/BKANT.TTF"/>
    </CipherData>
  </EncryptedData>
</encryption>
```

</EncryptedData>
</encryption>

References

- [1] OCF <http://www.idpf.org/ocf/ocf1.0/index.htm>
- [2] OPF: latest draft is available at
http://www.idpf.org/doc_library/informationaldocs/OPS/OPF_2.0_0.7_draft.htm
- [3] RFC 4122, “A Universally Unique Identifier (UUID) URN Namespace” e.g.
<http://www.packetizer.com/rfc/rfc4122>