IBM Systems MEDIA

Produce Secure PDFs With Signed Secure PDF Transform

In a digital environment, it’s important to establish trust when using electronic communication to share documents.

10/01/2018

In a digital environment, it’s important to establish trust when using electronic communication to share documents. To build that level of trust, you should use a digital signature to sign PDF documents. Digital signatures guarantee the authenticity of a document’s content and the authenticity of the signer. They also guarantee that documents have not been modified or altered since being signed.

IBM Signed Secure PDF Transform

The IBM Signed Secure PDF Transform feature enables users to sign PDF documents produced by IBM Print Transform from AFP to PDF for Infoprint Server for z/OS. It builds on the base of the AFP to PDF product to produce a digitally signed PDF by applying a digital certificate to the PDF content. It also provides additional security to prevent changes to sensitive transactional documents and to enable tamper detection.

With the Signed Secure PDF Transform feature, you can:

- Use industry-standard digital certificates to provide best-in-class protection by encrypting the hash of a document's contents
- Use SHA2 algorithms provided by z/OS Cryptographic Services
- Be assured that both the visual appearance and contents of a document remain unchanged

The Signed Secure PDF Transform feature uses the Certificate Manager Services API, which is part of the IBM Cryptographic Services System SSL, to process certificates located in z/OS repositories. When sharing signed PDFs with the public at large, a digital certificate from a member of the Adobe Approved Trust List must be obtained for use with the Signed Secure PDF Transform.

Applying Digital Signatures to Standard PDFs

After a digital certificate is obtained (this step is required), the Signed Secure PDF Transform feature goes through several internal steps to produce a digitally signed PDF:

1. Start with the PDF output from the base of the AFP to PDF product
2. Hash the base PDF document contents
3. Encrypt the resulting hash digest with the digital certificate's private key and store it in the document along with the digital certificate
4. Later, when the document is viewed using a conforming PDF viewer, hash the document contents again and compare with the hash value that was previously stored in the document. If the comparison of hash values fails, the PDF viewer alerts the user that the document may have been altered.
5. The Signed Secure PDF Transform feature embeds a detached public key cryptography standards signature object in the PDF document.

For more information and for a list of Adobe-approved providers of digital certificates, see the Adobe Approved Trust List (https://helpx.adobe.com/acrobat/kb/approved-trust-list2.html).

Conforming PDF Viewers

The term "conforming PDF viewer" refers to software that can process PDF 1.7 (ISO 32000-1) documents. Conforming PDF viewers allow a user to obtain additional details about the digital signature used to sign a document and the status of the certificate used to alert the user if any issues are detected with the certificate or the document contents.

To view certificate details for a signed PDF document using Adobe Acrobat:

1. Open the "Signature Panel" (the pen icon)
2. Click "Certified By <name of the certificate >"
3. Go to "View Signature Details" and then "View Certificate Details"
4. Clicking this option will open the "Certificate Viewer" dialog box that enables you to view details of the certificate used to sign the PDF document

Signed Secure PDF Transform Use Cases

There are three common use cases where the Signed Secure PDF Transform feature is typically used:

1. Sharing documents within an organization
2. Sharing documents between organizations
3. Sharing documents with the public at large

For the first two use cases, you can use RACF to create internal certificates and then share them with users to establish trust. The end users don't need to import the certificate if they want to view the PDF or to detect tampering. Users can choose to add the certificate that is used to sign a PDF document by adding the certificate to their trusted identities to establish trust for future documents signed with the same certificate.

When you need to share signed PDF documents with the public at large, you must use a certificate from a provider on the Adobe Approved Trust List. Adobe products are designed to periodically download a list of trusted digital certificates. Any digital certificate that can establish a relationship back to a trustworthy certificate will be displayed as a trusted certificate.
You will need to work with your system security team to set up the certificate repository, to create or install the required certificates and to set up appropriate access for use of the certificate/repository by the Signed Secure PDF Transform.

Creating Certificates

The JCL job in Figure 1 demonstrates how you can use RACF commands to create two certificates (a root certificate and an end user certificate) and then use them with the Signed Secure PDF Transform feature. For more information about the RACF commands and the sample JCL job, see "z/OS Security Server RACF Command Language Reference." (https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zOSV2R3SA232292/$file/icha400_v2r3.pdf)

```
//ADDCRTS JOB (SAMPLE01)
//STEP1       EXEC  PGM=IKJEFT01,DYNAMNBR=20
//SYSTSPRT    DD    SYSOUT=A
//SYSTSIN     DD    *
RACDCERT CERTAUTH GENCERT +
SUBJECTSDN(CN('Signed PDF Sample Certificate Name spdfca') +
           T('Signed PDF Sample Certificate spdfca' ) +
           OU('Sample Organization Unit') +
           O('Sample Organization') +
           L('sCity') SP('sState') C('US'))
+
SIZE(4096) RSA  +
WITHLABEL('Signed PDF Sample spdfca') +
KEYUSAGE(CERTSIGN)

RACDCERT ID(AUSER1) GENCERT +
SUBJECTSDN(CN('Signed PDF Sample Certificate Name spdf') +
           T('Signed PDF Sample Certificate spdf' ) +
           OU('Sample Organization Unit') +
           O('Sample Organization') +
           L('sCity') SP('sState') C('US'))
+
SIZE(4096)  RSA  +
WITHLABEL('Signed PDF Sample') +
KEYUSAGE(HANDSHAKE DOCSIGN) +
SIGNWITH(CERTAUTH LABEL('Signed PDF Sample spdfca'))

RACDCERT ADDRING(MYRING101) ID(AUSER1)
RACDCERT ID(AUSER1) CONNECT( CERTAUTH +
         LABEL('Signed PDF Sample spdfca') +
         USAGE(CERTAUTH)
RING(MYRING101))
RACDCERT ID(AUSER1) CONNECT(  +
         LABEL('Signed PDF Sample') RING(MYRING101))
/*
```

Figure 1: Sample JCL job to create certificates

The JCL example in Figure 1 creates a root certificate and an end user certificate that is signed by the root certificate. Next, it creates a key ring and connects the two certificates to the key ring. Once the certificates are connected to the key ring, user AUSER1 can use them when signing a PDF document using the Signed Secure PDF Transform.

Use the following entries in your AFPXPDF transform class named “signed” (the transform class is in the AOPXFD.CONF file):

```
transform afpxpdf_signed

  start-command = afpxpdfd
    min-active = 0
    max-active = 2
    maximum-idle-time = 300 # 5 minutes
    environment = {
       AOP_FONTLIB ->    " sys1.sfntilib "
       AOP_FORMDEFLIB -> " sys1.fdeflib  "
       AOP_PAGEDEFLIB -> " sys1.pdeflib  "
       AOP_PAGESEGLIB -> " sys1.pseglib  "
       AOP_OVERLAYLIB -> " sys1.overlib  "
       AOP_DIGITAL_CERTIFICATE_LOCATION -> 'AUSER1/MyRing101'
       AOP_SIGNATURE_DIGESTALG -> 'SHA512'
       AOP_SIGNATURE_PSIZE     ->  65535
       _BPX_JOBNAME -> AFPXPDFD
     }
;
```

An example transform command that uses the certificate that was just created is:

```
afpxpdf -o spdf.pdf -c signed -j " pdf-digital-certificate-name='Signed PDF
Sample'" signed_sample.afp
```

The “pdf-digital-certificate-name” attribute must be included to invoke the signed PDF transform processing.

Producing a Signed PDF: UNIX and AOPBATCH

The following UNIX command line example uses sample files that are shipped with the Signed Secure PDF Transform feature in the /usr/lpp/IBM/PrintXform/V1R2/samples directory:

```
afpxpdf -o signed_sample.pdf -c signed -j " pdf-digital-certificate-
name='samplesha256ss'" /usr/lpp/IBM/PrintXform/V1R2/samples/signed_sample.afp
```

In AOPBATCH, the input file "SAMPLE.INPUT.BAXTERBB" is first converted to a PDF and then signed using transform class “signed” and special job attributes to produce a signed PDF output “SAMPLE.PDFSGN02.” The resulting PDF file is then emailed to a recipient by using a printer definition designed to send emails with Infoprint Server email support. For a sample AOPBATCH job, see Figure 2.

```
//PDFSGN02 JOB (SAMPLE02)
//STEP1   EXEC PGM=AOPBATCH,
// PARM='/afpxpdf -c signed -jatt=//DD:ATTR -o//DD:OUT //DD:IN'
//IN    DD DSN=SAMPLE.INPUT.BAXTERBB,DISP=SHR
//OUT   DD DSN=SAMPLE.PDFSGN02,DISP=(NEW,CATLG,DELETE),
//    DCB=(RECFM=VB,LRECL=1024,BLKSIZE=1028),SPACE=(CYL,(5,5))
//ATTR   DD *
pdf-digital-certificate-name='samplesha256ss'
pdf-title='This is a sample PDF Title text - Signed'
pdf-signature-name='Sample Signature'
pdf-signature-reason='Sample Reason'
pdf-signature-contact='Sample Contact'
pdf-signature-location='Sample Location'
/*
//STDOUT DD SYSOUT=*
//STDERR DD SYSOUT=*
//STDENV DD *
AOP_MVS_RETURN_CODES=YES
/*
//STEP2   EXEC PGM=IEBGENER
//OUT1 OUTPUT  CLASS=C,DEST=PXEMAIL,
//             PRTATTRS='document-format=pdf',
//             MAILTO='userid@us.ibm.com',
//             TITLE='SAMPLE_SIGNED_PDF',
//             MAILFILE='SAMPLE.PDFSGN02.PDF'
//*
//SYSUT1   DD DSN=SAMPLE.PDFSGN02,DISP=SHR
//SYSUT2   DD SYSOUT=(,),OUTPUT=(*.OUT1)
//SYSPRINT DD SYSOUT=*
//SYSIN    DD DUMMY
//*
```

Figure 2: Sample AOPBATCH job to transform AFP input to PDF and sign it

To use the UNIX command line and AOPBATCH examples, users will need to install and set up the Signed Secure PDF Transform feature and add a special transform class “signed” in the /etc/Printsrv/aopxfd.conf file. See “Infoprint Server User’s Guide” (https://www-01.ibm.com/servers/resourcelink/svc00100.nsf/pages/zOSV2R3SA380695/$file/aopu000_v2r3.pdf) for details on configuring and sending PDF documents as email attachments.

You can also configure Infoprint Server printer definitions to receive print jobs from JES spool, transform them to PDF format and apply the digital certificate to produce a signed PDF document that can be emailed to end users using the sendmail protocol, all in one request.

Job Attributes and Environmental Variables

The following Infoprint Server job attributes have been added to support the Signed Secure PDF Transform feature:

- pdf-digital-certificate-name—Specifies the digital certificate name used for signing the PDF document.
- pdf-signature-name—Specifies the name of the person or authority signing the PDF document
- pdf-signature-reason—Specifies the reason for signing the pdf document
- pdf-signature-contact—Specifies the contact information about the signer or signing authority
- pdf-signature-location—Specifies either the server name or the physical location of signing

The following aopxfd.conf class environment variables have been added to support the Signed Secure PDF Transform feature:

- AOP_DIGITAL_CERTIFICATE_LOCATION—Specifies the repository name that contains the digital certificate to be used
- AOP_SIGNATURE_NAME—Specifies the name of the person or authority signing the PDF document
- AOP_SIGNATURE_REASON—Specifies the reason for signing the PDF document
- AOP_SIGNATURE_CONTACT—Specifies the contact information about the signer or signing authority
- AOP_SIGNATURE_LOCATION—Specifies either the server name or the physical location of signing
- AOP_SIGNATURE_DIGESTALG—Specifies the cryptographic hash algorithm to be used for generating the signature digest
- AOP_SIGNATURE_PSIZE—Specifies the amount of reserved space needed for the signature content key in the signature dictionary.
- AOP_GSKKYMAN_DB_PWORD—Specify this value if the digital certificate being used is stored in a gskkyman database
- AOP_GSK_FIPS_STATE—Specifies the FIPS mode setting the transform uses when calling CMS API services

More detailed information about these job attributes and environment variables can be found in “IBM Print Transforms from AFP for Infoprint Server for V1.2.2.” (https://www.ibm.com/servers/resourcelink/svc00100.nsf/pages/zOSV2R3G3252634/$file/aokfa00_v2r3.pdf)