How to sign the already signed PDF document

Written by Apitron Documentation Team

Introduction

Digital signatures provide a reliable way to sign documents and verify the identity of the signing person(s). They are used nowadays in many areas of our everyday workflow, and, when it comes to PDF, we luckily have the support for them built-in into the standard.

We have already covered the PDF signing and validating topic in one of our earlier posts (you can read it here), but it was related to signing the document with one or several signatures *at once.*

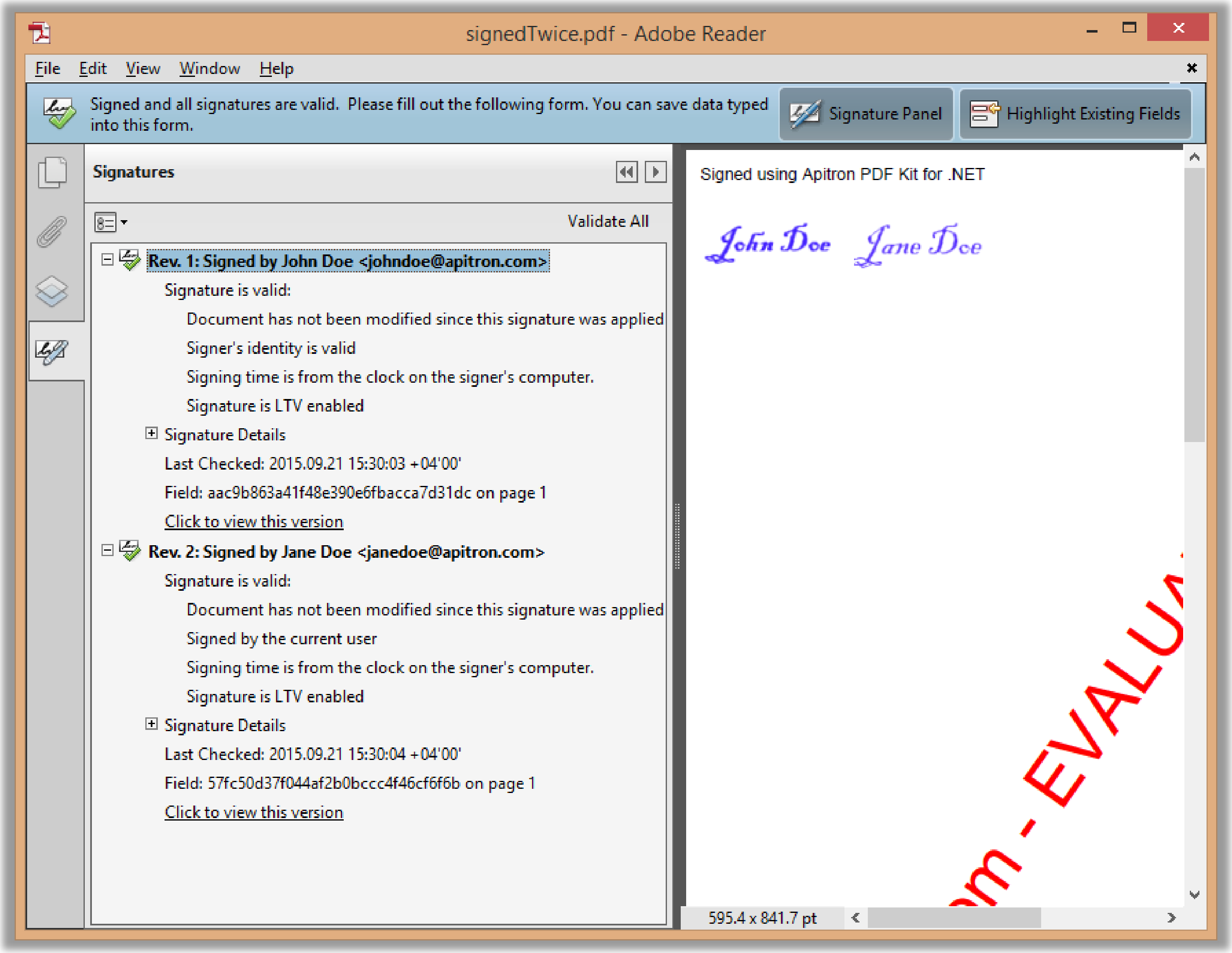
In this article we'll cover another interesting scenario where signing will be performed for previously signed document. "What is the difference?" - you may ask. Sequential signing requires different handling and implementation because as the document changes, previous signatures may become invalid. But the Apitron PDF Kit for .NET component can handle any signing task and we'll demonstrate it in code section.

Code sample

The code below adds two digital signatures to PDF document sequentially. This way you can add as many signatures as you want.

```
class Program
{
    private static void Sign(string pathToDocument, string pathToCertificate, string password,
       string pathToSignatureImage, Boundary signatureViewLocation)
    {
        // open existing document andd sign once
        using (Stream inputStream = new FileStream(pathToDocument,FileMode.Open, FileAccess.ReadWrite))
        {
            using (FixedDocument doc = new FixedDocument(inputStream))
            {
                string imageResourceId = Guid.NewGuid().ToString("N");
                string signatureFieldId = Guid.NewGuid().ToString("N");

                // register signature image resource
                doc.ResourceManager.RegisterResource(new Image(imageResourceId, pathToSignatureImage));

                // create first signature field and initialize it using a stored certificate
                SignatureField signatureField = new SignatureField(signatureFieldId);
                using (Stream signatureDataStream = File.OpenRead(pathToCertificate))
                {
                    signatureField.Signature = Signature.Create(
                       new Pkcs12Store(signatureDataStream,password));
                }

                // add signature fields to the document
                doc.AcroForm.Fields.Add(signatureField);

                // create first signature view using the image resource
                SignatureFieldView signatureView =
                     new SignatureFieldView(signatureField, signatureViewLocation);
                signatureView.ViewSettings.Graphic = Graphic.Image;
                signatureView.ViewSettings.GraphicResourceID = imageResourceId;
                signatureView.ViewSettings.Description = Description.None;

                // add views to page annotations collection
                doc.Pages[0].Annotations.Add(signatureView);

                // save as incremental update
                doc.Save();
            }
        }
    }
    static void Main(string[] args)
    {
        string fileName = "signedTwice.pdf";
        CreatePDFDocument(fileName);

        // save once and save
        Sign(fileName, "../../data/certs/JohnDoe.pfx", "password",
        "../../data/images/signatureImage.png", new Boundary(10, 750, 110, 800));

        // add second signature to the already signed doc and save
        Sign(fileName, "../../data/certs/JaneDoe.pfx",
        "password","../../data/images/signatureImageJane.png",new Boundary(120, 750, 220, 800));
    }

    // .. the code continues on next page
    private static void CreatePDFDocument(string fileName)
    {
        using (Stream stream = File.Create(fileName))
        {
            FlowDocument doc = new FlowDocument(){Margin = new Thickness(10)};
            doc.Add(new TextBlock("Signed using Apitron PDF Kit for .NET"));
            doc.Write(stream,new ResourceManager());
        }
    }
}
```

The resulting file looks as follows:



Signed and all signatures are valid. Please fill out the following form. You can save data typed into this form.

Rev. 1: Signed by John Doe <johndoe@apitron.com>
Signature is valid:
Document has not been modified since this signature was applied
Signer's identity is valid
Signing time is from the clock on the signer's computer.
Signature is LTV enabled
Signature Details
Last Checked: 2015.09.21 15:30:03 +04'00'
Field: aac9b863a41f48e390e6fbacca7d31dc on page 1
Click to view this version

Rev. 2: Signed by Jane Doe <janedoe@apitron.com>
Signature is valid:
Document has not been modified since this signature was applied
Signed by the current user
Signing time is from the clock on the signer's computer.
Signature is LTV enabled
Signature Details
Last Checked: 2015.09.21 15:30:04 +04'00'
Field: 57fc50d37f044af2b0bccc4f46cf6f6b on page 1
Click to view this version

Signed using Apitron PDF Kit for .NET

Pic. 1 Signed PDF document

Conclusion

Apitron PDF Kit handles multiple signing scenarios and can be used to create digitally signed documents in corporate, government and end user environments. You can download it from our website and try for yourself. Whenever you have a question you can read our free book, browse online API documentation and articles (see documentation page). Contact us if you need any help and we'll be happy to assist you.