On the Availability and Effectiveness of Open Source Software for Digital Signing of PDF Documents

Jonas Gamalielsson, Fredrik Jakobsson, Björn Lundell, Jonas Feist, Tomas Gustavsson, Fredric Landqvist

On the availability and effectiveness of open source software for digital signing of PDF documents

Jonas Gamalielsson[1], Fredrik Jakobsson[1], Björn Lundell[1], Jonas Feist[2], Tomas Gustavsson[3] and Fredric Landqvist[4]

[1]University of Skövde, Skövde, Sweden,
{jonas.gamalielsson,bjorn.lundell}@his.se
fredrik-jakobsson@live.se
[2]RedBridge AB, Kista, Sweden,
jfeist@redbridge.se
[3]PrimeKey Solutions AB, Solna, Sweden,
tomas@primekey.se
[4]Findwise AB, Göteborg, Sweden,
fredric.landqvist@findwise.com

Abstract. Digital signatures are important in order to ensure the integrity and authenticity of information communicated over the Internet involving different stakeholders within and beyond the borders of different nations. The topic has gained increased interest in the European context and there is legislation and project initiatives aiming to facilitate use and standardisation of digital signatures. Open standards and open source implementations of open standards are important means for the interoperability and long-term maintenance of software systems implementing digital signatures. In this paper we report from a study aiming to establish the availability and effectiveness of software provided under an open source license for digital signing and validation of PDF documents. Specifically, we characterise the use of digital signatures in Swedish Governmental agencies, report on the interoperability of open source and proprietary licensed software for digital signatures in PDF documents, and establish the effectiveness of software provided under an open source license for validation of digital signatures in PDF documents.

1 Introduction

With increased communication over the Internet involving information exchanged between different stakeholders within and beyond the borders of different nations, there is an increasing need to ensure the integrity and authenticity of such information (Kaur and Kaur, 2012; Roy and Karforma, 2012). Digital signatures (also known as cryptographic signatures) are important means to ensure that information received is authentic and that it has not been altered in transit. Application areas are for example E-commerce, E-governance, and E-learning. In the European context the EU has issued a directive for the establishment of a community

framework that forms the basis for legal recognition of electronic signatures[1] (EC, 1999). This directive has been adopted in legislation in order to facilitate the use of digital signatures in a number of EU countries including Sweden (SFS, 2000). There is also a Commission decision on the publication of generally recognised standards for electronic signatures (EC, 2003), which impacts on the standardisation in this field in Europe. More recently, the eIDAs regulation on electronic identification and trust services for electronic transactions was adopted by co-legislators to increase the support for cross-border interactions between businesses, citizens and public authorities (EC, 2014).

Challenges related to digital signatures include *interoperability* between software systems for digital signing of documents and validation of digital signatures in documents. It is of vital importance that different (both open source and proprietary licensed) software systems can interoperate effectively utilising open standards, protocols, and algorithms (Bird, 1998; Ghosh, 2005; UK, 2012). In fact, interoperability supports systems heterogeneity, thereby increasing options for organisations (Ghosh, 2005). Another challenge concerns *long-term maintainable* systems with effective support for digital signatures, since it is of fundamental importance that old digitally signed documents can be validated in contemporary software. Further, any software system needs to be maintained beyond the life-cycle of any specific provider through use of open standards and open source licensed software (Lundell, 2012; Lundell et al. 2011). Especially, there is increased risk for lack of long-term availability of both software and digital assets (e.g. documents) "if the commercial vendor of adopted proprietary software leaves the market" (Lundell et al., 2011).

The ***overarching goal*** of the study is to establish the availability and effectiveness of software provided under an open source license for digital signing and validation of PDF documents. We make three principal contributions. ***First***, we establish a characterisation of the use of digital signatures in Swedish Governmental agencies. ***Second***, we report on interoperability of open source and proprietary licensed software for digital signatures in PDF documents. ***Third***, we establish the effectiveness of software provided under an open source license for validation of digital signatures in PDF documents.

We focus on documents in PDF format since it is one of the most commonly used document formats and is widely deployed in both open source and proprietary licensed software. There is limited knowledge on the state of practice concerning use of digital signatures in Swedish public sector organisations. Further, knowledge is limited concerning details regarding availability, interoperability and effectiveness of open source licensed tools for digital signing and validation of PDF documents. To the best of our knowledge, this study is the first to report on such details.

For the first contribution, an investigation in Swedish Governmental agencies is of particular relevance given that Sweden is amongst the most IT-intensive countries in the EU (WEF, 2014). For the second and third contribution, the open source

[1] Electronic signature in this context is a broader term that includes digital (cryptographic) signatures

licensed tools iText (Itextpdf.com, 2015) and PDFBox (Apache.org, 2015) were used. Both software tools were used in combination with the Bouncy Castle Crypto API (Bouncycastle.org, 2015), which is also provided under an open source license. Those tools were selected since they have been identified as mature and are amongst the most widely adopted (in organisations in practice) and deployed open source libraries for creation, signing and validation of PDF files. Adobe Acrobat Professional XI Pro (Adobe.com, 2015) was selected and used since it is one of the most adopted and deployed proprietary tools for PDF processing, and is provided by the company that initially developed the PDF format. Further, for the second contribution LibreOffice Writer and Microsoft Word were chosen for generation of test documents with the motivation that they are both representative examples of software tools that are widely adopted, deployed, and provided under an open source and proprietary license, respectively.

The rest of this paper is organised as follows. We present a background on digital signatures and software support for digital signatures (section 2). Thereafter we present research approach (section 3), results (section 4), analysis (section 5), followed by discussion and conclusion (section 6).

2 On digital signatures and software support

A digital signature is an implementation of an asymmetric cryptography to ensure the integrity and authenticity of a document. A digital signature has the same purpose as the traditional physical signature, which is to prove the origin of the document, so that the recipient does not have any doubt that it was actually created by the person who sent it, and has not been tampered (with meaning or accidentally) along the way. The scheme for creation of a digital signature generally consists of three algorithms: 1) a key generation algorithm, which selects and outputs a private key and a corresponding public key; 2) a signature algorithm, which produces a signature given a private key and a message; and 3) verification algorithm, which accepts or rejects the authenticity claim of a message given a message, public key and a signature (Kaur and Kaur, 2012; Lowagie, 2013; Roy and Karforma, 2012). Central parameter choices for the signature algorithm include: encryption algorithm, e.g. RSA and DSA; standard for cryptographically protected messages, e.g. CMS (Cryptographic Message Syntax) and CAdES (CMS Advanced Electronic Signatures); and cryptographic hash (or “message digest”) function, e.g. the SHA (Secure Hash Algorithm) function family, MD5, and the RIPEMD (RACE Integrity Primitives Evaluation Message Digest) function family.

There are European project initiatives aiming to facilitate use and standardisation of digital signatures in the European context. One such example is the SD-DSS project (Joinup.eu, 2011) in which the European Commission has commissioned development of open source software for use by Member States and associated service providers to be able to complete the procedures and formalities that are necessary for conduction of activities with Member States' administrations within

and across borders. Another example is the E-signatures standards initiative whose mission is to create a rationalised framework for electronic signature standardisation in the European context (E-signatures-standards.eu, 2013). Specific standards addressed include CAdES, XAdES (XML Advanced Electronic Signatures), and PAdES (PDF Advanced Electronic Signature Profiles). This effort will support the realisation of one of the items of the EC Action Plan related to eSignatures, and is a collaboration between CEN (European Committee for Standardisation), EC (European Commission), ETSI (European Telecommunications Standards Institute), and the AFNOR Group (the French representative within CEN and ISO).

PDF is a document format that initially was maintained by Adobe. The PDF specification is available free of charge since 1993. PDF version 7 was released in November 2006 (Adobe.com, 2006), and in July 2008 this PDF version became available as an ISO standard (ISO, 2008). In the ISO standard for the PDF format there are details concerning support for digital signatures (ISO, 2008; Lowagie, 2013). Since the publication of the PDF standard (ISO, 2008), specific versions for specific purposes have been developed and standardised by ISO, for example the PDF/A standard for archiving purposes (ISO, 2005).

PDF is implemented in a number of software applications. **iText** is a library for PDF generation written mainly in Java (Itextpdf.com, 2015), and was initially provided under the MPLv1 and LGPLv2 licenses. However, the license was changed to the AGPLv3 license on 5 Dec. 2009 with the release of version 5.0.0. The latest stable version is 5.5.3, which was released on 17 Sep. 2014. The library is widely adopted in applications that include functionality for creation of PDF documents and digital signatures (using the Bouncy Castle Crypto API). **PDFBox** is a library that provides an API in Java to handle PDF documents (Apache.org, 2015), including signing and validation of digital signatures (using the Bouncy Castle Crypto API). The latest stable version is 1.8.7, which was released on 19 Sep. 2014, and is provided under the Apache License v2. It is used by eID-DSS, which is Belgium's solution for digital signatures for the eID solution for handling PDF documents. **Bouncy Castle** is a lightweight cryptography API for Java and C# (Bouncycastle.org, 2015). The latest stable version is 1.53, which was released on 28 Sep. 2014, and it is provided under the MIT license. It is a popular library that is utilised by various other open source projects and tools (apart from iText and PDFBox), including the SignServer application framework (Signserver.org, 2015). **Adobe Acrobat XI Pro** (Adobe.com, 2015) is software that can be used to view, create, manipulate, print and manage PDF documents. The latest stable version is 11.0.09 (released on 16 Sep. 2014), and it is provided under a proprietary license.

3 Research approach

As the first part of our approach we establish a characterisation of the use of digital signatures in Swedish Governmental agencies. The data collection is made easier to answer in Sweden, which has a very strict policy on governmental responses to

requests for public documents. In Sep. 2014 we sent an email in plain text to 71 Governmental agencies (the 16 IT intensive Governmental agencies in the Swedish e-Delegation, a selection of 35 other Swedish Governmental agencies, and all 20 Swedish Provincial offices). The email contained six requests: 1) Examples of one (or several) digitally signed PDF documents created within the organisation and sent to another public organisation; 2) Examples of one (or several) digitally signed PDF documents created within the organisation and sent to a corporation (or other private organisation) or to a private individual; 3) Examples of one (or several) digitally signed PDF documents submitted to the organisation from another public organisation; 4) Examples of one (or several) digitally signed PDF documents submitted to the organisation from a company (or other private organisation) or from a private individual; 5) The documents (i.e. documentation from identification of needs, evaluations, decisions, contracts, procurement documents and other related documents, etc.) that relate to that (or those) contract(s) and (or) development projects related to the software and the systems used for creating and managing digitally signed PDF documents in the organisation; and 6) The documents (i.e. agreements, regulations, policy, strategy, instructions and other documents) that regulate and describe how digital signatures (and software for digital signatures) are to be used in the organisation.

Second, we analyse the interoperability of open source and proprietary licensed software for digital signatures in PDF documents. This was done by generating and signing test documents using different software and signature settings that were subsequently validated using different software. Specifically, the test documents were created by enumerating all 42 combinations of a specific software for generation (LibreOffice Writer 4.2.6.3 *or* Microsoft Word 13), a specific software for signing (Adobe Acrobat XI Pro *or* iText 5.5.3 *or* PDFBox 1.8.7), a specific signature format setting for signing (CMS *or* CAdES for Adobe Acrobat XI Pro, CMS *or* CAdES for iText 5.5.3, CMS for PDFBox 1.8.7), and a specific hash algorithm setting for signing (SHA256 for Adobe Acrobat XI Pro, SHA1 *or* SHA256 *or* SHA384 *or* SHA512 *or* RIPEMD160 for iText 5.5.3, SHA1 *or* SHA224 *or* SHA256 *or* SHA384 *or* SHA512 *or* MD5 *or* RIPEMD128 *or* RIPEMD160 *or* RIPEMD256 for PDFBox 1.8.7). Software tools used for validation of each of the 42 test documents were Adobe Acrobat XI Pro, iText 5.5.3, and PDFBox 1.8.7.

Third, we analyse the effectiveness of software provided under an open source license for validation of digital signatures in PDF documents. This was done by validating different PDF documents using different software. Specifically, the PDF documents provided by the Swedish Governmental agencies and Provincial offices according to requests 1-4 were used. In addition, all digitally signed PDF documents from a large corpus of one million US governmental documents of mixed file formats randomly selected from the ".gov" domain (http://digitalcorpora.org/corpora/govdocs) were used (approximately 20% of those files were PDF documents, of which 156 were digitally signed PDF documents). Software tools used for validation of these documents were Adobe Acrobat XI Pro (primarily for validation of the documents provided by the Swedish Governmental agencies), three versions of iText

(v1.1.0: the first version with support for digital signatures, v2.1.7: the last LGPL licensed version, and v5.5.3: the latest version), and two versions of PDFBox (v1.6.0: the first version with support for digital signatures, and v1.8.7: the latest version). iText v1.1.0 and v2.1.7 was used in combination with Bouncy Castle Crypto API v1.38, whereas iText v5.5.3 was used in combination with Bouncy Castle Crypto API v1.53 (the latest version). PDFBox 1.6.0 was used in combination with Bouncy Castle Crypto API v1.38, whereas PDFBox 1.8.7 was used in combination with Bouncy Castle Crypto API v1.53.

For the second and third part, custom made shell scripts were used to integrate and utilise the iText and PDFBox libraries for signing and validation of the PDF documents, and for extracting meta data (e.g. date for signing) from the documents.

4 Results

4.1 On use of digital signatures in Swedish Governmental agencies

After sending the email that contained the 6 requests for information to each of the 71 Swedish Governmental agencies and Provincial offices including reminders we received totally 39 responses with answers. However, only a few of these contained any of the requested documents. In total, 15 examples of PDF documents according to requests 1 through 4 (see section 3) were provided by 10 of the Governmental agencies (of which four were provided by Provincial offices). In total, eight documents were provided by the Governmental agencies (excluding Provincial offices) of which two were according to request 1, two according to request 2, one according to request 3, and three according to request 4. All seven documents provided by the Provincial offices were according to request 3, but all of these documents were physically signed documents that had been scanned rather than documents with digital (cryptographic) signatures.

Concerning requests 5 and 6 to Governmental agencies (excluding Provincial offices), one agency has provided documents, 33 agencies have stated that there are no such documents. Concerning requests 5 and 6 to Provincial offices, two have stated that there are no such documents, one has stated that there are no documents for request 5 but that documents for request 6 will be provided (at time of writing not yet received).

4.2 Interoperability of software for digital signatures

All 42 generated and digitally signed PDF files could be validated successfully by iText 5.5.3 and PDFBox 1.87. Adobe Acrobat XI Pro failed to validate four of the 42 files that were signed using PDFBox 1.8.7 and the hash algorithms RIPEMD128 and RIPEMDF256. The reason for this is that those hash algorithms are not supported in Adobe Acrobat XI Pro.

4.3 Open source software support for validation of digital signatures

For the eight PDF documents provided by Swedish Governmental agencies, two documents were successfully validated by all the tested tools (in all versions). Three additional documents were successfully validated by all the tested tools (except iText v1.1.0, since the hash algorithm used for signing the document is not supported in this version). The remaining three documents could not be validated by any of the tested tools (in any version). One of these could not be validated since no digital signature could be found according to all tested tools, and the other two documents had a proprietary signature format that was not supported in any of the tested tools. Four of the five documents that could be validated were signed in 2014 and the fifth document was signed in 2013.

For the 156 signed PDF documents from the ".gov" domain, six documents could not be successfully validated using any of the tested tools (in any version). By attempting to validate the signatures in these six documents in Adobe Acrobat XI Pro it was found that the signatures contained either "incorrect", "unrecognized", "corrupt", or "suspicious" data. The remaining 150 documents could be successfully validated by all the tested tools (except iText v1.1.0, for which 20 additional documents could not be validated since the hash algorithms used for signing the documents are not supported in this version). The 150 documents that could be validated were signed in the interval 2000-2009 and the majority of documents were signed during 2008-2009, see Table 1.

Table 1. Number of signed documents per year for the 150 validated ".gov" PDF-documents

Year	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009
# doc	5	13	4	7	6	8	9	19	50	29

5 Analysis

First, from the results (section 4.1) it is clear that the use of digital signatures in Swedish Governmental agencies (including Provincial offices) is very limited. Very few Governmental agencies have provided any documents on how digital signatures for documents are created, managed, and used within the organisation. Few digitally signed documents have been provided (only eight in total, of which five actually contained valid signatures). Hence, in general there seems to be a lack of policy for and deployment of digital signatures and their use in Swedish Governmental agencies. Except from the Swedish law concerning qualified electronic signatures and requirements on qualified certificates and the issuing of such certificates (SFS, 2000), there are no nationwide regulations or recommendations concerning when and in what contexts digital signatures shall be used.

Second, from the results (section 4.2) it is evident that the open source licensed tools are at least as effective as the proprietary licensed tool Adobe Acrobat XI Pro for signing PDF documents and validating signatures in PDF documents, and that the

tested tools to a very large extent are interoperable. The lack of support for two specific hash algorithms (RIPEMD128 and RIPEMD256) in Adobe Acrobat XI Pro in the interoperability test may not be surprising, since support for these is not required according to the ISO specification for PDF (ISO, 2008, p. 476).

Third, from the results (section 4.3) it is clear that an overwhelming majority of the 156 digitally signed PDF documents from the ".gov" domain could be successfully validated in all the tested tools. It can also be noted that documents as old as 15 years could be validated. This shows, from a long-term maintenance perspective, that it is possible to validate older digitally signed documents using contemporary software. It also shows that software older than the digitally signed documents can be used for validation, since the digital signatures in the five files provided by Swedish Governmental agencies are from 2013-14 and could be validated with iText v2.1.7 (released 7 Jul. 2009) in combination with Bouncy Castle Crypto API v1.38 (released 7 Nov. 2007) and with PDFBox v1.6.0 (released 1 Jul. 2011) in combination with Bouncy Castle Crypto API v1.38 (released 7 Nov. 2007).

We acknowledge that software solutions for digital signing of PDF documents may be combined with hardware modules for enhanced security. However, as the overarching goal of our study is to establish the availability and effectiveness of open source software for digital signing of PDF documents, the analysis of such solutions is beyond the scope of our study. For future research it is relevant to also consider other file formats and associated open source software, in other usage contexts.

One openness aspect to consider is the licensing conditions for the standards for cryptographically protected message formats and hash functions used by the tested software tools, since it impacts on the possibility to (legally) provide implementations of the standards under an open source license. A patent search shows that there are no disclosed patents in the IETF patents database[2] for the most recent version of CMS (IETF RFC 5652, issued in 2009). However, there are two disclosed patents for an earlier version of CMS (IETF RFC 2630, issued in 1999) concerning technology used by S/MIME, a standard for public key encryption and signing of MIME (Multipurpose Internet Mail Extensions) data. For CAdES (RFC 5126) there have been no patent disclosures. For the ETSI version of CAdES (TS 101 733 V.1.7.4) there are no disclosed patents in the ETSI patents database[3]. Hence, the latest versions of CMS and CAdES do not seem to be encumbered by patents, and are therefore possible to implement in open source software. However, there may still be undisclosed patents that have not been reported to IETF or ETSI. A patent search in the ISO patent database[4] on a standard (ISO/IEC 10118-3:2004) involving the SHA and RIPEMD hash function families shows that there are several disclosed patents. Hence, these hash functions may cause problems when implementing such standards in open source software from a legal standpoint. There may also be other undisclosed patents that have not been reported to ISO for this standard.

[2] https://datatracker.ietf.org/ipr/
[3] http://ipr.etsi.org/
[4] http://www.iso.org/iso/standards_development/patents

Concerning certain standards for digital signatures (e.g. CAdES) there are a number of profiles offering different protection levels for different user groups, including CAdES-BES (basic form), CAdES-X (extended), and CAdES-LT (long term). This proliferation into different variants for the same standard may imply increased complexity and effort when developing and testing software that implements the standard.

6 Discussion and Conclusion

Use of readily available and effective open source tools for digital signing and validation of PDF documents is a strategy that simplifies the implementation of digital signatures for an organisation and that promotes a long-term sustainable software system with associated communities. In particular, open source licensed solutions offer more flexibility concerning which software version to use and when to update to a new version, something which often is critical for the stability of systems implementing digital signatures. Further, use of open standards for digital signatures is of vital importance since they promote interoperability between (both open source and proprietary licensed) software systems, and also ensure long-term sustainability of the digital signatures.

Concerning the current limited use of digital signatures in the Swedish context, we envisage that increased efforts on provision of infrastructure for digital signatures at national level will promote increased use of digital signatures. Further, complex software solutions for digitally signing documents may inhibit adoption of digital signatures and improved system support simplifying this process can promote broader adoption of digital signatures.

In conclusion, our study shows that there are open source licensed tools available for digital signing and validation of PDF documents that are at least as effective as the proprietary licensed tool Adobe Acrobat XI Pro. Further, it is shown that the tested (open source and proprietary licensed) software tools to a large extent are interoperable. It is also shown that there is very limited use of digital signatures for documents in the context of Swedish Governmental agencies. The findings from our study therefore make an important contribution to practice and policy.

References

Adobe.com: PDF Reference – Adobe Portable Document Format, Version 1.7. www.adobe.com/content/dam/Adobe/en/devnet/pdf/pdfs/pdf_reference_1-7.pdf, Accessed 5 Jan. 2015 (2006)

Adobe.com: Acrobat XI Pro. http://www.adobe.com/products/acrobatpro.html, Accessed 5 Jan. 2015 (2015)

Apache.org: Apache PDFBox – A Java PDF Library. https://pdfbox.apache.org/, Accessed 5 Jan. 2015 (2015)

Bird, G. B.: The Business Benefit of Standards. StandardView, Vol. 6, No. 2, pp. 76-80. (1998)

Bouncycastle.org: The Legion of the Bouncy Castle. https://www.bouncycastle.org/, Accessed 5 Jan. 2015 (2015)

EC: DIRECTIVE 1999/93/EC OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL on a Community framework for electronic signatures. 13 Dec. 1999, Official Journal of the European Union, L13/12 (1999)

EC: Commission decision on the publication of reference numbers of generally recognised standards for electronic signature products in accordance with Directive 1999/93/EC of the European Parliament and of the Council. 14 Jul. 2003, Official Journal of the European Union, L175/45 (2003)

EC: REGULATION (EU) No 910/2014 OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL on electronic identification and trust services for electronic transactions in the internal market and repealing Directive 1999/93/EC. 23 July 2014, Official Journal of the European Union, L257/73 (2014)

E-signatures-standards.eu: e-signatures standards – making e-signatures easy. http://www.e-signatures-standards.eu/, Accessed 5 Jan. 2015 (2013)

Ghosh, R. A.: Open Standards and Interoperability Report: An Economic Basis for Open Standards. FLOSSPOLS, Deliverable D4, 12 Dec., Maastricht (www.flosspols.org) (2005)

ISO: Document management -- Electronic document file format for long-term preservation -- Part 1: Use of PDF 1.4 (PDF/A-1). ISO/TC 171/SC 2, ISO 19005-1:2005 (2005)

ISO: Document management -- Portable document format -- Part 1: PDF 1.7. ISO/TC 171/SC 2, ISO 32000-1:2008 (2008)

Itextpdf.com: iText – Programmable PDF Software. http://itextpdf.com/, Accessed 5 Jan. 2015 (2015)

Joinup.eu: Digital Signature Service. https://joinup.ec.europa.eu/asset/sd-dss/description, Accessed 5 Jan. 2015 (2011)

Kaur, R. and Kaur, A.: Digital Signature. In Proceedings of the International Conference on Computing Sciences (ICCS 2012), pp. 205-301, 14-15 Sep. 2012, doi: 10.1109/ICCS.2012.25 (2012)

Lowagie, B.: Digital Signatures for PDF documents. http://itextpdf.com/book/digitalsignatures20130304.pdf, Accessed 5 Jan. 2015 (2013)

Lundell, B.: Why do we need Open Standards?, In Orviska, M. and Jakobs, K. (Eds.) Proceedings 17th EURAS Annual Standardisation Conference 'Standards and Innovation', The EURAS Board Series, Aachen, ISBN: 978-3-86130-337-4, pp. 227-240 (2012)

Lundell, B., Lings, B. and Syberfeldt, A.: Practitioner perceptions of Open Source software in the embedded systems area. Journal of Systems and Software, Vol. 84, No. 9, pp. 1540–1549 (2011)

Roy, A. and Karforma, S.: A survey on digital signatures and its applications. Journal of Comper and Information Technology (IJCIT), Vol. 3, pp. 45-69 (2012)

SFS: Lag om kvalificerade elektroniska signaturer. Statens författningssamling, SFS 2000:832, 2 Nov. 2000 (2000)

Signserver.org: SignServer – PKI by PrimeKey. http://www.signserver.org/, Accessed 5 Jan. 2015 (2015)

UK: Open Standards Principles: For software interoperability, data and document formats in government IT specifications. Cabinet Office, U.K., 1 Nov. (2012)

WEF: The Global Information Technology Report 2014, World Economic Forum, Geneva, ISBN-13: 978-92-95044-63-0 (2014)