

# Digital signature

From Wikipedia, the free encyclopedia.

In cryptography, **digital signatures** are a method of authenticating digital information often treated, sometimes too closely, as analogous to a physical signature on paper. Whilst there are analogies, there are also differences which can be important. The term *electronic signature*, although sometimes used for the same thing, has a distinct meaning: it refers to any of several, not necessarily cryptographic, mechanisms for identifying the originator of an electronic message. In common law, such electronic signatures have included cable and Telex addresses, as well as FAX transmission of handwritten signatures on a paper document. (For example, see *Cloud Corp v Hasbro* in the legal cases section below).

## Contents

- 1 Introduction
  - 1.1 Operational outline — public key cryptography
  - 1.2 Operational outline — digital signatures
- 2 The current state of use — legal and practical
- 3 Evidential status
- 4 Some digital signature algorithms
- 5 Legal aspects
  - 5.1 United States
  - 5.2 England, Scotland and Wales
  - 5.3 India
  - 5.4 New Zealand
  - 5.5 United Nations Commission on International Trade Law
- 6 Legal cases

## Introduction

A digital signature is itself simply a sequence of bits conforming to one of a number of standards. It is the generation of those bits, and their interpretation at some later time/place, and the cryptographic protocols and algorithms which used to govern both which give a digital signature bit sequence meaning in contrast to just any bit sequence.

Most digital signatures rely on public key cryptography, and a basic understanding of the principles of these schemes is required to understand how digital signatures work.

## Operational outline — public key cryptography

The method depends on the fact that anyone can transform a message into cyphertext using a public key, but that the 'matching' private key is needed to reverse that transformation. The following is a very brief outline.

Consider *Alice and Bob*. Bob wants Alice (and other people, for that matter) to be able to send secure secret messages to him. To permit this, Bob (most carefully indeed) generates a *key pair* consisting of two (mathematically) related "keys". One key is called the *public key*, and the other, the *secret* or *private key*. In the most useful of these algorithms, it is impractical, even for a well-funded organization, to compute the private key from the public key (this is the *private key / public key* property). Bob keeps his private key secret, and publishes his public key (on his webpage, for example, or by sending it to a key server, or by going to a key signing party).

Alice (or someone else) retrieves Bob's public key (at the same party perhaps) and, using it to control the appropriate encryption algorithm, scrambles or *encrypts* the message. For quality algorithms (at least in the belief of those well-informed on the subject), once encrypted using Bob's public key, the cyphertext cannot be descrambled or *decrypted* without the private key. Thus, no one who intercepts the cyphertext will be able to read it, even knowing Bob's public key. When Bob receives

the message, he decrypts it using his private key (kept secret since generation time and so known only to him). Therefore, the message will be secure against the unauthorized, and Bob and Alice do not need a "secure channel" to exchange a shared key.

The above is a simple outline of the method, and does not deal with the details of how the key pairs are generated, how they are applied to encrypt and decrypt the message, and what prevents an attacker with access to the scrambled message and the public key from retrieving the unscrambled message or the secret key. All are critical to security in that if any is done improperly the message is very likely to be readable by someone other than Bob. See public key cryptography.

An important feature of public/private key pairs is that their functions are interchangeable. A message encrypted with the public key can only be decrypted with the private key (as above), but also a message encrypted with the private key can only be decrypted using the public key. It is this feature that digital signatures are based upon.

## Operational outline — digital signatures

Now consider a somewhat different circumstance, in which Bob wants to send a message to Alice and wants to be able to prove it came from him (but doesn't care whether anybody else reads it). In this case, Bob sends a cleartext copy of the message to Alice, along with a copy of the message encrypted with his *private key* (**not** the public one). Alice (or any other recipient) can then check whether the message really came from Bob by decrypting the cyphertext version of the message with Bob's public key and comparing it with the cleartext version. If they match, the message was really from Bob, because the private key was needed to create the signature and no one but Bob has it. The cyphertext version is Bob's *digital signature* for the message because anyone can use Bob's public key to verify that Bob created it.

More usually, Bob applies a cryptographic hash function to the message and encrypts the resulting message digest using his private key. This makes the signature much shorter and thus saves both time (since hashing is generally much faster than public-key encryption), and space (since even an encyphered message digest is much shorter than the cyphertext version of the entire plaintext). However, with poor quality message digest algorithms or ones with too short a digest, this scheme may be susceptible to attack, in particular (but not exclusively) a birthday attack.

*To finish: current and future applications, actual algorithms, standards, why not adopted as widely as expected, etc.*

## The current state of use — legal and practical

Digital signature schemes all have several prior requirements without which no such signature can mean anything, whatever the cryptographic theory or legal provision.

- First, quality algorithms. Some public key algorithms are known to be insecure, practicable attacks against them having been identified.
- Second, quality implementations. An implementation of a good algorithm (or protocol) with mistake(s) will not work. (Software developers typically expect about 1 defect per 1,000 lines, unless intense efforts have been taken to raise its quality, in which case 1 defect per 1,000,000 lines is typically expected).
- Third, the private key must remain actually secret; if it becomes known to some other party, that party can produce *perfect* digital signatures of anything whatsoever.
- Fourth, distribution of public keys must be done in such a way that the public key claimed to belong to Bob actually belongs to Bob, and vice versa. This is commonly done using a public key infrastructure and the public key ↔ user association is attested by the operator of the PKI (called a certificate authority). For 'open' PKIs in which anyone can request such an attestation (universally embodied in an identity certificate), the possibility of mistake is non trivial. Commercial PKI operators have suffered several publicly known problems. Such mistakes could lead to falsely signed, and thus wrongly attributed, documents.
- Fifth, users (and their software) must carry out the signature protocol properly.

Only if each and every one of these conditions is met will a digital signature actually be evidence of who sent the message.

Legislatures, being importuned by businesses expecting to profit from operating a PKI, or by the technological avant-garde advocating new solutions to old problems, have enacted statutes and/or regulations in many jurisdictions authorizing, endorsing, encouraging, or permitting digital signatures and providing for (or limiting) their legal effect. The first appears to have been in Utah, followed closely by Massachusetts and California. Assorted non-US countries have also passed statutes or

issued regulations in this area as well and the UN has had an active model law project for some time. These enactments (or proposed enactments) vary from place to place, have typically embodied expectations at variance (optimistically or pessimistically) with the state of the underlying cryptographic engineering, and have had the net effect of confusing potential users and specifiers, nearly all of whom are not cryptographically knowledgeable. Adoption of technical standards for digital signatures have lagged behind much of the legislation, delaying a more or less unified engineering position on interoperability, algorithm choice, key lengths, etc and so on what the engineering is attempting to provide.

*See also:* ABA digital signature guidelines

## Evidential status

Many of the legal enactments (statute or regulation) surrounding digital signatures is concerned with their admissibility as evidence. More controversial, however, is their actual value as evidence. Unlike a traditional handwritten signature, a digital signature may be generated automatically, without the knowledge of the authorized user. It is generated by complex software, operating on an operand whose nature and existence cannot be fully or directly verified by the authorized user. Whereas the existence of a digital signature can be evidentially significant in establishing that an electronic communication is uncorrupted, and that it had a certain provenance, it cannot of itself provide any evidence as to whether a particular individual intended or authorized or associated himself or herself with any such communication. In that regard, the term "signature" is potentially misleading as the engineering does not now, and may possibly not be able to, coincide with the assumptions underlying many of the legal enactments. Legal enactments which affirmatively declare that a digital signature is presumptively deemed a valid signature are at variance with the possibilities afforded by the cryptography.

However, if the right software is used in the right way, including not leaking the private key, then the digital signature on some message can be created only by definite actions of the person in question, therefore validating the use of digital signatures.

## Some digital signature algorithms

- RSA
- DSA
- ElGamal

## Legal aspects

Legislation concerning the effect and validity of digital signatures includes:

### United States

- Uniform Electronic Transactions Act (UETA)
- Electronic Signatures in Global and National Commerce Act (E-SIGN), at 15 U.S.C. 7001 (<http://www4.law.cornell.edu/uscode/15/7001.html>) et seq.

### England, Scotland and Wales

- Electronic Communications Act, 2000 (<http://www.legislation.hmso.gov.uk/acts/acts2000/20000007.htm#7>)

### India

- Information Technology Act, 2000 (<http://www.mit.gov.in/it-bill.asp>)

### New Zealand

- Electronic Transactions Act, 2003 sections 22-24 (<http://www.legislation.govt.nz/>)

## United Nations Commission on International Trade Law

- UNCITRAL Model Law on Electronic Signatures (2001), a strong influence in the field. (<http://www.uncitral.org/english/texts/electcom/ecommerceindex.htm>)

## Legal cases

Court decisions discussing the effect and validity of digital signatures or digital signature-related legislation:

- *In re Piranha, Inc.*, 2003 WL 21468504 (N.D. Tex) (UETA does not preclude a person from contesting that he executed, adopted, or authorized an electronic signature that is purportedly his).
- *Cloud Corp. v. Hasbro*, 314 F.3d 289 (7th Cir., 2002)[1] (<http://www.emlf.org/Resources/cloud.pdf>) (E-SIGN does not apply retroactively to contracts formed before it took effect in 2000. Nevertheless, the statute of frauds was satisfied by the text of E-mails plus an (apparently) written notation.)
- *Sea-Land Service, Inc. v. Lozen International*, 285 F.3d 808 (9th Cir., 2002) [2] (<http://www.admiraltylawguide.com/circt/9thsealandlozen.pdf>) (Internal corporate E-mail with signature block, forwarded to a third party by another employee, was admissible over hearsay objection as a party-admission, where the statement was apparently within the scope of the author's and forwarder's employment.)

Retrieved from "[http://en.wikipedia.org/wiki/Digital\\_signature](http://en.wikipedia.org/wiki/Digital_signature)"

Categories: Cryptography

---

- This page was last modified 08:56, 1 Mar 2005.
- All text is available under the terms of the GNU Free Documentation License (see **Copyrights** for details).