code signing

THAWTE IS A LEADING GLOBAL PROVIDER OF CODE SIGNING CERTIFICATES

thawte®

code signing

Developers know that their creations are not immune from malicious tampering. Particularly for software distributed via the Internet, tampering is a common concern. Software can be modified to include malware, and the software's original, non-malicious author will be blamed for the harm the modified software does. Developers want customers to trust their software, and a code signing certificate from Thawte makes that possible.

What is Code Signing?

Code signing is a way to protect software from unauthorized, undetected modification. When signed using a code signing certificate, the software's exact content is locked into the signature; customers can easily verify the signature to determine the software author's identity, and most operating systems (OSs), web browsers, and other software can validate the signature to ensure that the software is unmodified from its originally-authored form.

Should signed software prove to be malicious or damaging, customers have recourse with the publisher—that publisher is clearly identified in the signature, and the signature proves that the software was not corrupted or modified by anyone other than the publisher. This accountability provides a strong deterrent to the distribution of harmful code.

WHAT IS CODE SIGNING USED FOR?

In today's world, *all* executable computer code should be signed, whether it is distributed via the Internet, as part of an OS, or sold from shelves in retail stores. Code signing helps establish trust between customers and software publishers. A customer decides that they trust a particular software publisher, and the signed code tells the customer that the software did, in fact, come from that publisher—allowing the customer's trust to carry over to the software.

WHAT DOES CODE SIGNING *NOT* DO?

Signed code is not guaranteed to be non-harmful; a malicious or incompetent software publisher could still create harmful software and digitally sign it. However, the signature makes it impossible for the author to do so anonymously—and most malware authors rely on anonymity to protect themselves from the consequences of their creations.

Again, code signing certifies the *identity* of the publisher and the *integrity* of the software; it does not certify the purpose or quality of the software.

WHERE DOES TRUST BEGIN?

Customers' trust in signed code begins with their trust in a Certification Authority (CA), such as Thawte. Thawte's detailed identity verification procedures ensure that code signing certificates act as a kind of digital ID card—a digital certificate that names a particular company is issued *only* to that company, ensuring that nobody else can impersonate that company using the company's digital ID. Trust in Thawte allows those customers to have confidence in software publishers.

Your Customers' Code Signing Experience

Slightly different techniques let users know when code is signed. Sometimes this experience takes place within a web browser, while other times is communicated from the computer's operating system. For example, Adobe Air warns when an unsigned application is installed (see Figure 1).

Figure 1: A warning for an unsigned application.

Using a signed application ensures that the publisher is known, prompting a different dialog box (see Figure 2).

Figure 2: Evidence of a signed application.

Browsers, such as Microsoft Internet Explorer, can often be configured with a variety of security levels, including levels that prohibit the installation of unsigned code that has been downloaded from the Internet. Browser extensions often support code signing; in Firefox, for example, installing a browser add-on displays the name of the publisher for signed extensions (see Figure 3).

Figure 3: Displaying the name of the publisher for signed extensions.

Java Virtual Machines (JVMs) are also designed to detect signed code and to display information about the code signing certificate, helping users decide whether to trust the code (see Figure 4).

Figure 4: JVMs also display certificate information.

Code signing is also used in productivity applications such as Microsoft Office, in operating system security features such as Windows Software Restriction Policies. Today, nearly every aspect of operating systems, platforms, and applications that rely on external code, will display visual cues like the ones seen here to help users make trust decisions. In every case, signed code is always displayed as "known", which is inherently more trustworthy than "unknown".

How Code Signing Works

Code signing uses a combination of encryption keys and encryption technologies to ensure the integrity of code and to communicate the identity of code publishers.

PUBLIC AND PRIVATE KEYS

At the heart of code signing are two asymmetric encryption keys called the *public* and *private* keys. As their names imply, the public key is widely available; it is typically included along with the signed code. The private key is available only to the code publisher. Anything encrypted with one key can be decrypted only by the other key, hence the *asymmetric* qualifier. This forms the basis for code signing.

ENCRYPTION

Encryption uses mathematical algorithms to create a representation of data that can be read only by using the proper decryption key. In the case of code signing, a publisher uses their private key to encrypt key information about their code. The information used depends on the exact software platform, but in general, this key information can be compared with the actual code to determine whether the code has been altered. Only the software publisher's public key can decrypt this signature; anyone, then, can decrypt the signature and use the decrypted information to determine whether the actual code has been altered. If the decryption was successful, then the publisher's identity is verified, because only the publisher would have access to the private key needed to encrypt the information in such a way that the public key could be used to decrypt it.

TIME STAMPING

Encryption is not foolproof or unbreakable; however, breaking modern encryption would require an immense amount of time and effort. Certificates can also be lost or compromised in other ways, and for these and other reasons, code signing certificates always expire, usually after one year. This expiration gives the issuing CA an opportunity to re-validate the certificate holder's identity, if needed, and to renew the certificate.

However, your software code is likely to be in use long after your certificate expires. For this reason, many platforms and operating systems support *time stamping*, a means of applying a digitally-signed date and time stamp that indicates when code was signed. So long as the time stamp is within the validity period of your certificate—meaning your certificate was not expired at the time of signing—then your code signature remains valid. Not every means of signing code supports time stamping; consulting your developer documentation for details.

Code Signing Solutions from Thawte

Thawte offers a variety of code signing certificates designed for use with different software platforms and operating systems.

CODE SIGNING CERTIFICATES

Although the basic technology behind code signing is universal, each major software development platform requires that certificates be packaged in particular ways. For this reason, Thawte offers code signing certificates in a variety of formats:

- Apple® Developer Certificate
- JavaSoft™ Developer Certificate
- Microsoft® Authenticode® (Multi-Purpose) Certificate
- Microsoft® Office and VBA Developer Certificate
- Adobe® AIR™ Developer Certificate

Thawte Multi-Purpose Code Signing Certificates provide interoperability across different developer platforms and are extremely useful if you want to use a single certificate to sign code for multiple browsers, platforms, or operating systems. Start by requesting a Microsoft Authenticode Certificate, which can be saved on your computer's disk. The certificate can then be imported into the Windows registry and then exported to a variety of other formats.

Key Code Signing Tools

In addition to a code signing certificate, you will need the right developer tools to actually apply a digital signature, using the certificate, to your code. Tools include the Java Developer Kit, Microsoft Windows SDK, Microsoft Office, and others.

Useful Links

You may find the following URLs to be useful:

- Code signing FAQs available at:
  hhttp://www.thawte.com/resources/ssl-information-center/ssl-beyond-ecommerce/code-signing-faq/index.html

- Review code signing certificates and start your purchase at:
  https://www.thawte.com/code-signing/index.html

About Thawte

Thawte is a CA that issues code signing and SSL digital certificates to organizations and individuals worldwide. Thawte performs various levels of verification and authentication depending on the certificate product. Thawte digital certificates interoperate smoothly with the most common web servers, browsers, and other applications, so you can rest assured that the purchase of a Thawte digital certificate will add integrity to your published software.

Contact Details

If you have further questions, or would like to speak with a Sales Advisor, please feel free to contact us:

- E-mail: sales@thawte.com
- North America: +1 888 484 2983
- International: +27 21 819 2800
- Fax: +27 21 819 2960
- Live Chat: https://www.thawte.com/chat/chat_retail_new.html