

WHITE PAPER

# VERISIGN<sup>®</sup> CODE SIGNING CERTIFICATE FOR ADOBE<sup>®</sup> AIR<sup>™</sup> APPLICATIONS

BUILDING TRUSTED RELATIONSHIPS  
WITH SOFTWARE USERS





## CONTENTS

- 1 WHAT IS CODE SIGNING AND WHY IS IT IMPORTANT?
- 2 WHO NEEDS CODE SIGNING CERTIFICATES?
- 2 WHAT DOES CODE SIGNING LOOK LIKE TO CONSUMERS?
- 3 TECHNICAL OVERVIEW: (OPTIONAL READING)
  - 3 WHAT IS A DIGITAL CERTIFICATE?
  - 3 CAs
  - 3 HOW DOES ADOBE AIR CODE SIGNING WORK WITH VERISIGN CODE SIGNING CERTIFICATE?
  - 4 TIMESTAMPING
  - 4 THE SIX STEPS TO SIGNING CODE
- 5 CONCLUSION
- 5 LEARN MORE
- 5 ABOUT VERISIGN





# VERISIGN® CODE SIGNING CERTIFICATE FOR ADOBE® AIR™ APPLICATIONS

## WHAT IS CODE SIGNING AND WHY IS IT IMPORTANT?

Code Signing, sometimes called Object Signing, is a technology that has the ultimate objective of helping software publishers to increase the adoption of downloaded applications. Code Signing helps by making end users just as confident about installing and using software obtained via the Internet as they are with software obtained in person. When people buy software in a store, the safety of installing and using that software is obvious. They can observe who published the software, and they can see whether the package has been opened. These factors enable them to make decisions about which software to purchase and how much to trust that software.

Contrast this to users downloading software from the Internet where they do not have the same confidence-building experience. Perhaps they even receive a message warning them about the dangers of using the software, which diminishes their confidence even more. The Internet lacks the tangible information provided by packaging, shelf space, shrink wrap, and the like. Without an assurance of the software's integrity or its publisher's credentials, customers have difficulty deciding how much to trust it. They can be justifiably concerned that it might be from a source other than that stated or might have been tampered with, and may therefore contain viruses or other forms of malware. Reluctant to proceed with their downloads, people may well go to a retail outlet instead — or simply not install the software at all. This represents a loss of business for the commercial software vendor and an inconvenience or worse for the customer, who might not even be able to find it on store shelves.

To increase the adoption of software via the online channel, preserve their business reputation, and build a trusted relationship with customers, software publishers must protect their products from those who would alter it with malicious intent. These publishers must show users that they have

implemented strong protective measures. To maximize the software adoption they also need to allow their applications to run without requiring that users alter their security settings while avoiding security warning messages about unknown sources that scare away customers.

VeriSign® addresses these challenges with a solution that digitally “shrink-wraps” the software with a digital signature. Using a platform signing utility, publishers can digitally sign their applications on Adobe® AIR™ with a VeriSign® Code Signing Certificate, which verifies the authenticity of the publisher and the integrity of the product. Because VeriSign's root certificates come pre-installed on most end users' Windows and Mac OS computers, they are trusted by these operating systems. In other words, VeriSign code signing certificates help software to install smoothly, without triggering the security warnings and error messages that make potential users anxious.

When end users download software signed with a VeriSign Code Signing Certificate, they can be assured of:

- Content Source—The publisher identified in the code signing certificate
- Content Integrity—The software has not been altered or corrupted since it was signed.

Users benefit from this software accountability because they know who published the software, and they know the code has not been tampered with. In the extreme case that software performs unacceptable or malicious activity on their computers, users can also pursue recourse against the publisher. This accountability and potential recourse serve as a strong deterrent to the distribution of harmful code.





## WHITE PAPER

Software publishers benefit from VeriSign Code Signing Certificates because their use puts trust in their name and makes their products harder to falsify. By signing code, publishers build a trusted relationship with users, who then learn to confidently download and install signed software from that publisher. And users can make educated decisions about which software to download. Choosing VeriSign for the Certificate Authority (CA) increases the odds that users will adopt their software, because it's the world's most trusted and widely recognized Internet security brand.<sup>1</sup>

### WHO NEEDS CODE SIGNING CERTIFICATES?

Any publisher who plans to distribute code or content over the Internet or over corporate extranets risks impersonation and tampering. VeriSign Code Signing Certificate for Adobe AIR protects against these hazards. For this reason Adobe requires that all applications on Adobe AIR be digitally signed.

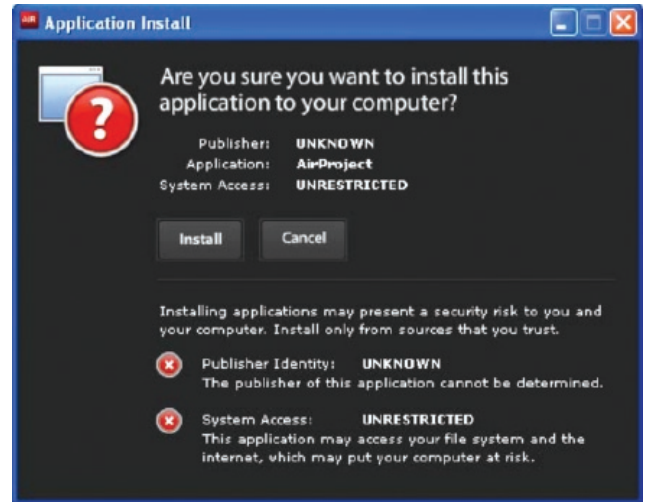
VeriSign offers a Class 3 certificate designed for not only commercial software publishers, but also companies and other organizations that publish software. This class of certificate provides greater assurance about the identity of a publishing organization and is designed to represent the level of assurance provided today by retail channels for software.

Software publishers can also sign their applications on Adobe AIR using a self generated certificate. In these cases, the integrity of the application is maintained via digital signatures. However, the publisher is listed as "unknown."

### WHAT DOES CODE SIGNING LOOK LIKE TO CONSUMERS?

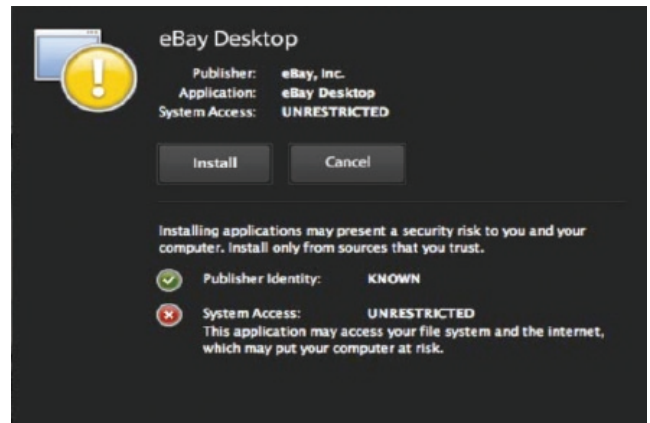
If a user attempts to install an AIR application that has been signed by a certificate from an unknown authority, or an self-signed certificate, the following screen will appear:

Figure 1: Unsigned Applications



By contrast, if a user attempts to install an AIR application that has been signed by a certificate from a known certificate authority, then an Adobe AIR installation dialog box with the publisher's identity such as the following will appear:

Figure 2: Applications signed with a VeriSign Code Signing Certificate



Through Adobe AIR, the user is informed of:

- The true identity of the publisher
- The extent of the application's system access





## TECHNICAL OVERVIEW

What Is a Digital Certificate?

A digital certificate is a form of electronic credential for the Internet. Similar to a driver's license, employee ID card, or business license, a digital certificate is issued by a trusted third party to establish the identity of the certificate holder. The third party who issues certificates is known as a Certificate Authority (CA).

Digital-certification technology is based on the theory of public key cryptography. In public key cryptography systems, every entity has two complementary keys – a public key and private key – which function only when they are held together. Public keys are widely distributed to users, while private keys are kept safe and only used by their owner. Any code digitally signed with the publisher's private key can only be successfully verified using the complementary public key. Another way to look at this is that code successfully verified using the publisher's public key (which is sent along with the digital signature), can only have been digitally signed using the publisher's private key (thus authenticating the source of the code), and has not been tampered with.

The purpose of a digital certificate is to reliably link a public/private key pair with its owner. When a CA such as VeriSign issues a certificate, it verifies that the owner is not claiming a false identity. Just as when a government issues you a passport, it is officially vouching for the fact that you are who you say you are. When a CA issues you a digital certificate, it is putting its name behind the statement that you are the rightful owner of your public/private key pair.

CAs

CAs, such as VeriSign, are organizations that issue digital certificates to applicants whose identity they are willing to vouch for. Each certificate is linked to the certificate of the CA that signed it.

As the Internet's leading CA, VeriSign has the following responsibilities:

- Publishing the criteria for granting, revoking, and managing certificates
- Granting certificates to applicants who meet the published criteria
- Managing certificates (for example, enrolling, renewing, and revoking them)
- Storing VeriSign root keys in an exceptionally secure manner
- Verifying evidence submitted by applicants
- Providing tools for enrollment
- Accepting the liability associated with these responsibilities
- Timestamping digital signatures

How Does Adobe AIR Work with VeriSign Code Signing Certificates?

Adobe AIR relies on industry-standard cryptography techniques such as X.509 v3 certificates and Public Key Cryptography Standard (PKCS) #7 and #10. These are well proven cryptography protocols, which ensure a robust implementation of code signing technology. Adobe AIR uses digital signature technology to assure users of the origin and integrity of software. In digital signatures, the private key generates the signature, and the corresponding public key validates it.

To use Adobe AIR and digital signature technology to develop applications, the publisher obtains a VeriSign Code Signing Certificate and develops and signs its application by following the procedure described below under The Six Steps to Signing Code.





## WHITE PAPER

Then, when an end user runs an Adobe AIR-based application that encounters the package:

1. Adobe AIR validates the publisher's certificate. Using the VeriSign root public key, which is already embedded in the application, it verifies the authenticity of the Code Signing Certificate (which is itself signed by the CA certificate that is signed by the VeriSign root private key).
2. Using the publisher's public key contained within the publisher's certificate, Adobe AIR decrypts the signed hash that the publisher created during the signing process.
3. Adobe AIR runs the code through the same hashing algorithm as the publisher, creating a new hash.
4. Adobe AIR compares the two hashes. If they are identical, it proves that the content was not altered after the signature was generated. The end user then has confidence that the code was signed by the publisher identified in the certificate and that the code has not been altered since it was signed.

The entire process is seamless and transparent to end users, who see only a message that the content was signed by its publisher.

### Timestamping

Because key pairs are based on mathematical relationships, which can theoretically be "cracked" with a great deal of time and effort, it is a well-established security principle that digital certificates should expire. Your VeriSign Code Signing Certificate will expire one, two, or three years after it is issued. However, most software is intended to have a longer lifetime. To avoid having to resign software every time your certificate expires, VeriSign provides a timestamping service. Now, when you sign code, a hash of your code will be sent to VeriSign to be timestamped. As a result, when your code is downloaded, clients will be able to distinguish between:

- Code signed with a revoked certificate, which should NOT be trusted
- Code signed with a certificate that was valid at the time the code was signed but has subsequently expired, which SHOULD be trusted

This means that you will not need to worry about resigning code when your VeriSign Code Signing Certificate expires. This service is free to all VeriSign commercial and individual Code Signing Certificate customers. VeriSign was the first CA that provided a timestamping server compatible with Adobe Air timestamping capabilities.

The Six Steps to Signing Code  
Signing Code is an easy six-step process:

**Step 1: Make sure you are running the correct versions of all tools.** The tools include:

- Adobe AIR runtime
- A development environment such as Dreamweaver CS3, Flex3 SDK, Flex Builder 3, Flash CS3 Professional, or AIR SDK

**Step 2: Apply for a VeriSign Code Signing Certificate for Adobe AIR.**

*Click here* for instructions on obtaining a VeriSign Code Signing Certificate. In the process of applying for a VeriSign Code Signing Certificate, your browser will generate a private key. Use Firefox to ensure consistency across operation systems. The private key is security stored in the certificate store on your computer. This key is never sent to VeriSign, so if you lose this private key, you will be unable to sign code. If this key is lost or stolen, please contact VeriSign immediately.

**Step 3: Pick up your Certificate.**

Once you have completed the application process, VeriSign will take a number of steps to verify your identity. For commercial publishers, VeriSign does a considerable amount of background checking. As a result, it will take approximately 3-5 days to verify your information and issue a certificate.

At the end of this process, VeriSign will send you an email containing a personal identification number (PIN). Follow the instructions in this email to pick up your certificate. You should back up or export your certificate and private key so





that it can be stored in a safety deposit box or other secure location.

Please note that you must use the same machine to apply for and obtain your certificate. You can then use the private key and certificate to sign files on a different machine.

#### Step 4: Prepare your files to be signed (.air or .airi).

.air files allow a publisher to create an air application that can be signed later. .airi files are intermediate files created to allow a build system to sign the applications. This is useful as the organization's credentials are usually managed by a build engineer.

#### Step 5: Sign your files.

The specifics of this step vary depending on the development platform. Following are the online description of the steps:

*Dreamweaver CS3:*

[www.adobe.com/go/learn\\_dw\\_air\\_signature\\_en](http://www.adobe.com/go/learn_dw_air_signature_en)

*Flex3 SDK:*

[www.adobe.com/go/learn\\_flexsdk\\_air\\_signature\\_en](http://www.adobe.com/go/learn_flexsdk_air_signature_en)

*Flex Builder 3:*

[www.adobe.com/go/learn\\_flexbuilder\\_air\\_signature\\_en](http://www.adobe.com/go/learn_flexbuilder_air_signature_en)

*Flash CS3 Professional:*

[www.adobe.com/go/learn\\_flash\\_air\\_signature\\_en](http://www.adobe.com/go/learn_flash_air_signature_en)

*AIR SDK:*

[www.adobe.com/go/learn\\_airsdk\\_air\\_signature\\_en](http://www.adobe.com/go/learn_airsdk_air_signature_en)

#### Step 6: Test the Adobe AIR Installation.

Double-click the AirProject.air file and the AIR installation process will begin (make sure you have the Adobe AIR runtime installed). As the first step of the installation process, the Adobe AIR installer displays the information about the publisher of the code—VeriSign. The dialog box shown in Figure 2 (see page 2) is displayed, reassuring users that the application is from a trusted publisher and has not been tampered with since it was published.

## CONCLUSION

Adobe and VeriSign are committed to making the Internet a secure and viable platform for commerce and the distribution of content. With Adobe AIR and VeriSign Code Signing Certificates, your code will be as safe and trustworthy to your customers as it would be if you shrink-wrapped it and sold it off a store shelf.

## LEARN MORE

For more information, visit [www.verisign.com](http://www.verisign.com) or call 1-866-893-6565 or 1-650-426-5112, option 3.

## ABOUT VERISIGN

VeriSign is the trusted provider of Internet infrastructure services for the digital world. Billions of times each day, companies and consumers rely on our Internet infrastructure to communicate and conduct commerce with confidence.

Visit us at [www.Verisign.com](http://www.Verisign.com) for more information.

