

AuthAgent X.509

Digital Certificate Authentication



AuthAgent X.509 is an implementation of public key and digital certificate authentication for embedded devices. It is designed for use as an authentication mechanism for various network security protocols and also as a standalone authentication agent that can be used by embedded applications. It provides the ability to validate certificates issued by a trusted Certificate Authority (CA) and includes advanced features such as certificate generation and revocation. Given its small footprint and ability to scale out optional features, **AuthAgent X.509** is ideally suited for use in embedded environments.

The X.509 Standard

In its simplest form, a digital certificate could just contain a public key and a name. To be useful, however, the certificate should also contain other fields such as an expiration date, the name of the CA that issued the certificate, a serial number, and other pertinent information. The popular ITU X.509 standard provides a structure for public-key certificates. X.509 digital certificates include not only an entity's name and public key, but also other information about the entity. **AuthAgent X.509** enables a certificate authenticator

Features

- ❖ Robust authentication framework using ITU-T X.509 digital certificates
- ❖ Support for various PKCS formats and X.509v3 extensions
- ❖ Interoperable with standard X.509 implementations on other platforms
- ❖ Support for validating certificates against a list of trusted certificates
- ❖ Support for Certificate Revocation List and OCSP
- ❖ APIs for customizing the certification validation procedure
- ❖ Support for multiple CPU types of either endian-ness including PowerPC, MIPS, X86, ARM/ XScale
- ❖ Royalty-free full source distribution

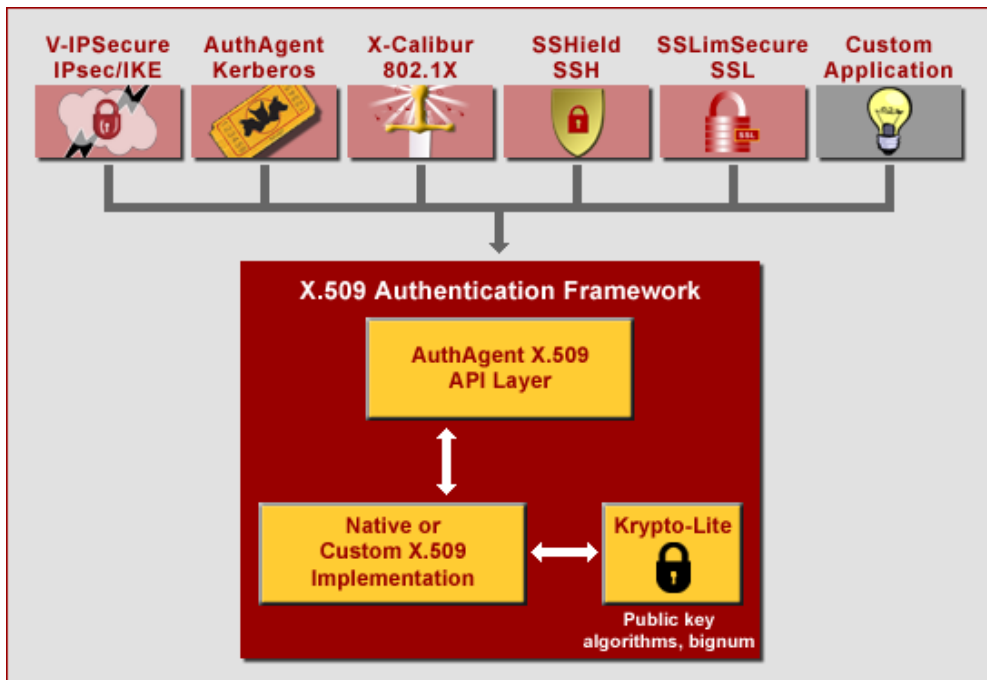


Fig 1: AuthAgent X.509 Operation

to verify the certificate's subject, and also obtain other trust-worthy information about the certificate's subject. It provides flexible APIs for validating certificates against a list of trusted CAs and for customizing the certification validation procedure based on various attributes retrieved from the certificate.

Digital Certificates

Public Key Cryptography provides a form of encryption that uses a key pair

that consists of two related keys -- a public key and a private key. This type of cryptography provides a scalable form of encryption that does not rely on the sharing of secrets. The public key can be used to verify a message signed with the corresponding private key or to encrypt a message that can only be decrypted using the corresponding private key. A Digital Certificate binds an identity to a key pair and is issued by a trusted third-party called a Certificate Authority (CA). It is digitally signed with the CA's private key after it has verified the entity's identity and hence, it is tamper-proof and easily portable which makes it ideal for embedded device authentication.

Validity Period & Revocation

X.509 certificates define a validity period which should be shorter than the expected factoring time of a brute force attack on the public-key algorithm. This plays an important role in the key size of the key pair to avoid such attacks. **AuthAgent X.509** supports this feature when the current time is available on the embedded device via manual settings, real-time clock hardware, or via an (S)NTP client. However, if an entity's private key is compromised before it expires, or if the CA's key is compromised or it can no longer vouch for the certificate holder, the certificate needs to be revoked. X.509 Certificates can be revoked by the CA that issued them. **AuthAgent X.509** also supports RFC 3280 Certificate Revocation Lists (CRLs) which are time-stamped lists of certificates that are revoked but have not yet expired. In **AuthAgent X.509**'s implementation, a CRL is optionally checked against when verifying a

certificate. **AuthAgent X.509** also supports RFC 2560 for Online Certificate Status Protocol (OCSP), using which the revocation status of certificates can be checked in a more real-time manner as compared to CRL which is used offline.

Certificate Formats

X.509 Certificates, private keys, CRLs, certificate requests can be distributed in various file formats. **AuthAgent X.509** supports the following file formats:

- ❖ PEM-formatted Base-64 Certificates
- ❖ PKCS12 certificate-key pair
- ❖ PKCS7 signed certificates and CRLs
- ❖ PKCS10 certificate request
- ❖ PKCS8 private key

Implementation Abstraction

AuthAgent X.509 provides a library with an API that is independent of the underlying X.509 implementation. This enables the software using X.509 based digital certificates for authentication to be designed and implemented, independent of the changes in the X.509 implementation. A default implementation that reads PEM formatted certificates and uses ASN.1 objects is included internally. **AuthAgent X.509** also includes I/O abstractions for storing, modifying, and retrieving trusted CA certificates and CRLs.

X.509 Applications

AuthAgent X.509 can be used as a stand-alone authentication mechanism for embedded applications in situations

Also Available

- ❖ **AuthAgent RADIUS**
A Remote Authentication Agent
- ❖ **AuthAgent Kerberos**
A Kerberos Authentication Agent
- ❖ **V-IPSecure**
Network Layer Security
- ❖ **SSHield**
Secure Shell

Custom Solutions

Customized validation procedures for self-signed certificates and customized implementations of **AuthAgent X.509** for your unique application needs are available through expert help from **TeamF1**'s professional services team.

Customization Flexibility

- ❖ Available in full-source format
- ❖ Certification validation procedure can be customized
- ❖ API abstractions that allow any custom X.509 implementation to be used
- ❖ Unwanted components can be scaled out

where device identity or access control has to be established. Additionally, **AuthAgent X.509** is natively integrated with TeamF1's network security protocol implementations providing authentication for **SSLimSecure** (SSL), **SSHield** (SSH), **V-IPSecure** (IPsec), and **X-Calibur** (802.1X). It can also be used for the initial identification phase of Kerberos authentication in PKINIT mode, and can be integrated with various third-party protocol implementations.

Email: sales@TeamF1.com
Web: www.TeamF1.com
Ph: 510-505-9931 ext. 5
Fax: (510) 505-9941



© 2003-2007 TeamF1, Inc.