Technical Disclosure Commons

Defensive Publications Series

August 2021

Mechanism to Authenticate a Reader to a Credential

David Mercer

Steve Paik

Ross Hewit

Recommended Citation

Mercer, David; Paik, Steve; and Hewit, Ross, "Mechanism to Authenticate a Reader to a Credential", Technical Disclosure Commons, (August 30, 2021)
https://www.tdcommons.org/dpubs_series/4562

Mechanism to Authenticate a Reader to a Credential

ABSTRACT

Access to data objects stored on a credential such as a badge, smart card, etc. is typically limited to user authorization through the use of a user-entered PIN or other mechanism. This disclosure describes techniques to enable a credential reader to authenticate itself to a credential and access protected objects on the credential without user interaction and without the use of any global credential. The techniques define a simplified public-key infrastructure (PKI) hierarchy appropriate for typical credentials, which are usually low-powered, passive, and offline.

KEYWORDS

- Personal identity verification (PIV)
- FIPS-201
- ISO-7816
- Identification card
- Smart card
- Employee badge
- Near-field communication (NFC)
- Certificate authority
- Credential reader

BACKGROUND

A credential is an electronic device, e.g., a smart card, a smartphone, a smartwatch, etc., that stores data objects accessible under an access control policy. The stored data objects can have varying degrees of sensitivity, e.g., photos, personal information, clearance information, emergency contact information, etc. Access to these data objects is typically limited to user authorization through the use of a user-entered PIN or other mechanism. An example is a bank card, where a PIN code is entered to access the signing key of the card used to authorize a transaction.

A credential reader is an electronic device capable of requesting data objects from a credential through a contact or contactless electronic interface. An example is a mobile phone that employs a near-field communication (NFC) interface to read NFC-enabled tags. Provisioning is the act of writing data to credentials or to readers.

A trust root is an entity trusted by a credential to digitally sign leaf credentials which enable the credential to be trusted by other devices. In this context, the word ‘credentials,’ plural in form but singular in meaning, refers to information, such as a certificate or signature from a certificate authority, that establishes the identity of a device or entity and enables others to trust the device or entity. Trust roots are expressed as trusted certificate authorities identified by a signed X.509 certificate.

DESCRIPTION

This disclosure describes techniques to enable a credential reader to authenticate itself to a credential (e.g., smartcard, badge, smartphone, smartwatch, etc.) and access protected objects on the credential without user interaction and without the use of any global credential. In view of the limited processing power and non-existent network connectivity for some credentials, e.g., badges, smart cards, etc., the techniques define a simplified public-key infrastructure (PKI) hierarchy appropriate for low-powered, passive, offline credentials.

Per the techniques, prior to its issuance, a credential is provisioned with the public keys of one or more trust roots. The expression of these public keys may be through an X.509 certificate, a raw public key, or another data object that includes the public key of the trust root.

The credential stores these keys such that they are protected from deletion, e.g., by non-trusted entities.

Readers are also issued credentials by trust roots, which establishes their identity and issuing authority to a credential (such as a smart card or a badge). The credentials of a reader comprise a private key (securely generated by the reader) and a public key. The identity and issuing authority of the credentials of the reader are expressed as a signed X.509 certificate, a digital signature of the public key performed by the trust root (including the identity of the trust root), or through another mechanism that securely identifies the issuing authority of the credentials of the reader and makes such credentials tamper-evident.

Access control lists for various data objects on a credential (e.g., smartcard, badge, etc.) are written such that specific operations (e.g., read, sign, decrypt, etc.) can only be performed by readers to whom credentials have been issued from authorized trust roots.

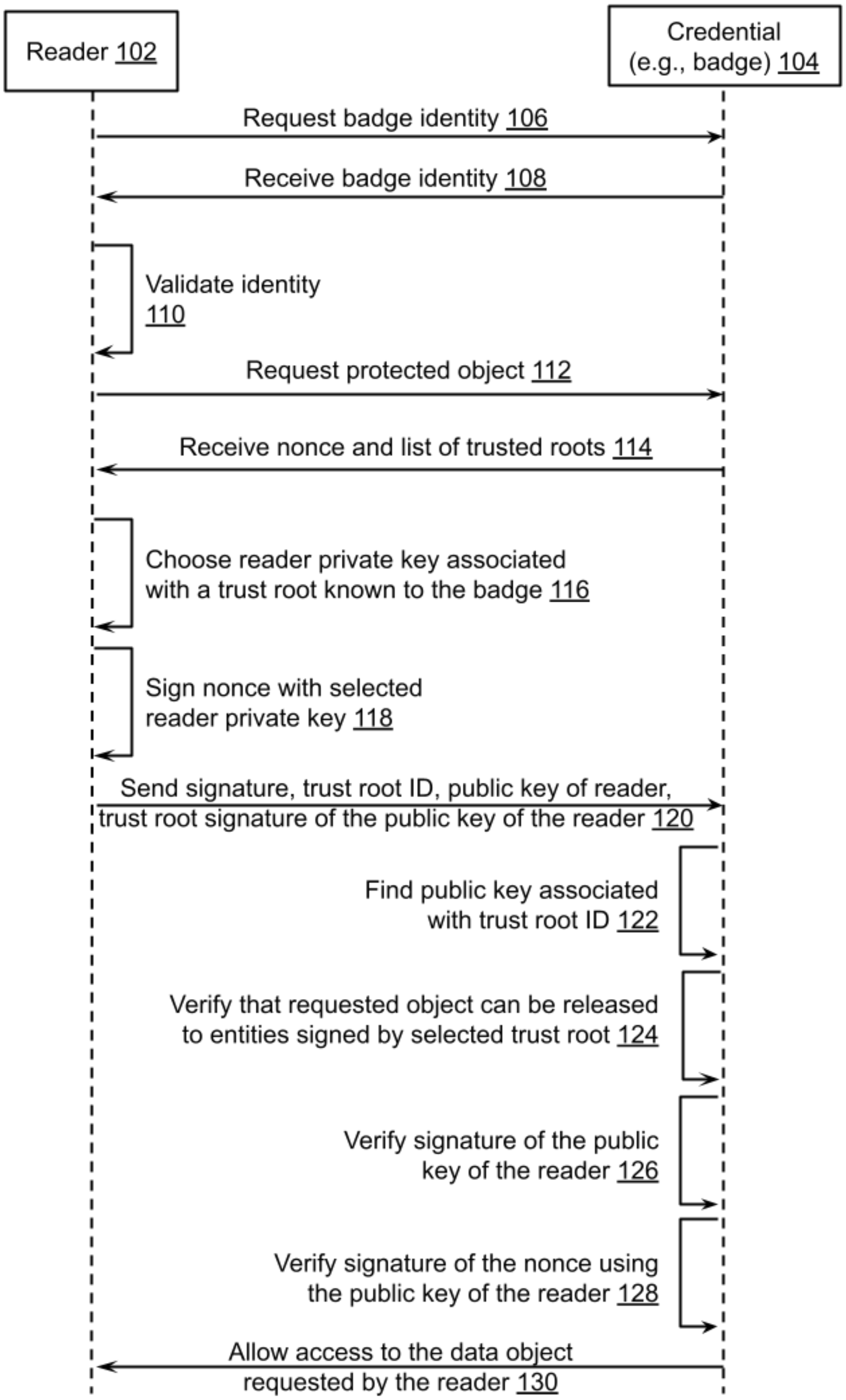
Reader 102
Credential (e.g., badge) 104
Request badge identity 106
Receive badge identity 108
Validate identity 110
Request protected object 112
Receive nonce and list of trusted roots 114
Choose reader private key associated with a trust root known to the badge 116
Sign nonce with selected reader private key 118
Send signature, trust root ID, public key of reader, trust root signature of the public key of the reader 120
Find public key associated with trust root ID 122
Verify that requested object can be released to entities signed by selected trust root 124
Verify signature of the public key of the reader 126
Verify signature of the nonce using the public key of the reader 128
Allow access to the data object requested by the reader 130

Fig. 1: Authenticating a reader to a credential

Fig. 1 illustrates authenticating a reader (102) to a credential (104), e.g., a badge, to enable the reader to securely read objects stored on the credential. The reader requests the badge identity (106) from the credential and receives it (108). The reader validates the badge identity (110). The reader requests to read a specific protected object (112) from the credential. The credential responds to the reader with an authentication request comprising a random nonce and a list of trust roots authorized to access the object (114). This list may be expressed as a list of X.509 certificates, a list of public key hashes, a simple list of trust root names, or by another mechanism that enables a reader to uniquely identify a trust root.

The reader searches for credentials issued to it by that trust root to identify one that is included in the list. If it finds none, processing terminates. Upon discovery of credentials issued by an appropriate trust root (116) the reader returns to the credential a digital signature of the nonce (118), the identity of the discovered trust root, the public key of the credential of the reader used to sign the nonce, and the digital signature of the public key of the credential by the discovered trust root (120).

Upon receipt of the above, the credential performs the following operations. It retrieves the public key associated with the trust root identified by the reader (122). If none can be found, processing terminates. It verifies that the data object requested by the reader can be accessed with credentials issued by the identified trust root (124). If not, processing terminates. It verifies the signature of the public key of the reader (126) using the public key of the trust root. If verification fails, processing terminates. It verifies the signature of the nonce using the public key of the reader (128). If verification fails, processing terminates. Upon successful verification, the credential allows access to the data object requested by the reader (130).

In this manner, the described techniques enable authorized entities of various classes, e.g., security officers, boarding gate agents, etc., to use credential readers that enable authentication to a credential to allow limited reading of protected information off the credential.

Effectively, the techniques provide a mechanism for a badge and a reader to develop trust in each other. The trust mechanisms are usable even with badges or other credentials that have low compute power and no network connectivity.

CONCLUSION

This disclosure describes techniques to enable a credential reader to authenticate itself to a credential and access protected objects on the credential without user interaction and without the use of any global credential. The techniques define a simplified public-key infrastructure (PKI) hierarchy appropriate for typical credentials, which are usually low-powered, passive, and offline.

REFERENCES

[1] https://github.com/makinako/OpenFIPS201 accessed Jul. 27, 2021.

[2] https://csrc.nist.gov/publications/detail/fips/201/2/final accessed Jul. 27, 2021.

[3] https://csrc.nist.gov/publications/detail/sp/800-73/4/final accessed Jul. 29, 2021.