December 01, 2017

Manufacturer origin attestation for device user authorization

Breno de Medeiros

Michael Dietz

Mengcheng Duan

Arnar Birgisson

Manufacturer origin attestation for device user authorization

ABSTRACT

This disclosure describes techniques for user authentication and authorization for devices with support for manufacturer origin attestation. A client attestation certificate allows an authorization service to associate a device with its manufacturer (client). A unique client identifier is assigned by the authorization service to the client. The client assigns a unique instance certificate to each device. During initial authorization, each device uses a trusted local channel to establish identification before the authorization service and obtain an authorization code. The authorization code, the device instance attestation certificate chain, and a proof of possession of the instance key, in the form of a signed message that includes the authorization code, are supplied by the device for verification. Upon verification of the supplied code and the signed message, the authorization service returns user credentials for the device.

KEYWORDS

- Manufacturer attestation
- Authorization service
- Attestation service
- User credentials
- Certificate chain
- OAuth
- Internet of Things (IoT)
- Access Token

BACKGROUND

Provisioning of devices for applications often includes creating device authorization credentials. The credentials are often based on proof-of-possession of a private key. Currently used authentication mechanisms based on device cryptographic credentials are vulnerable to supply chain attacks. Mechanisms based on bearer tokens are vulnerable to token theft. Establishing device credentials that also authenticate the manufacturer of the device can provide additional guarantees for users and also allow for the implementation of controls based on manufacturer-restricted content licenses.

DESCRIPTION

This disclosure describes techniques for performing user authentication and authorization for devices and device-based applications that incorporate manufacturer origin attestation.

Client registration

Registration with a user's authentication/authorization service (or identity provider) is undertaken by a manufacturer of devices or developer of device-based apps (hereinafter referred to as the 'client"). With the client registration process, a client attestation certificate is associated with the client by the authorization service. The authorization service is in possession of a public key, while the client is in possession of the associated private key.

The private key serves as a cryptographic signing key for the client. In addition, a unique client identifier is assigned by the authorization service to the client. The unique client identifier can be an arbitrary identifier that is unique to the authorization service registration records, or can be an identifier cryptographically derived from the client attestation certificate, such as a certificate fingerprint. The client attestation certificate can be used to issue additional certificates for specific instances of devices and apps created by the client.

Instance certificate assignment

The client assigns a unique certificate to each device or instance of an application running on the device. The instance certificate is certified by the attestation certificate of the client, with a direct signature or a chain of intermediate certificates as necessary. This creates an instance attestation certificate chain. The authenticity of the chain can be verified as belonging to the registered client by the authorization service. The last certificate in the chain is referred to as the instance certificate and the corresponding private key as the instance key.

The instance key is stored by the device, and in the case of a key associated with an app, protected from access by other apps on the same device. For example, if the device includes a trusted platform module (TPM), the TPM can be involved in the generation of the instance certificate. This provides a strong guarantee against theft of the private key.

For example, a manufacturer of smart speakers can obtain a client attestation certificate. Subsequently, for each speaker manufactured, a unique instance certificate for each device is created and certified with the client attestation certificate or with an intermediate certificate, where the intermediate certificate is itself certified by the client attestation certificate. The resulting instance attestation certificate chain can be installed on the speaker.

Initial authentication/authorization request

At the time of initializing the use of the device or app, the client device or app communicates with the authorization service to request that a user credential, e.g., an authorization code, be placed in the device. The user credential enables the device or app to access resources on behalf of the user. Initially, the client device or app is not authorized and holds no credentials for the user account with the authorization service.

The authorization can be carried out from a second trusted device or an app that holds the credentials. For the trusted device to establish the identification of the new device or app to the authorization service, it is sufficient for the trusted device to securely obtain the instance certificate, or associated fingerprint, from the second trusted device. A trusted local channel between the devices can be used for this purpose. For example, the trusted local channel may be a browser that the user is signed in to.

Authorization code exchange

During the authorization code exchange, the device or app and the authorization service are in direct communication. The authorization code, the instance attestation certificate chain, and a proof of possession of the instance key, in the form of a signed message that includes the authorization code, are supplied by the device or app.

The authorization code is validated by the authorization service. In addition, the instance attestation certificate chain is validated to be a valid certificate chain, where each certificate includes a valid signature issued by the parent. The root certificate of the chain is verified to be registered by the client that matches the client identifier previously presented at the authorization request that produced the authorization code. The authorization service also verifies that the leaf certificate of the chain (the instance certificate) matches the fingerprint presented in the original authorization request.

In addition, the authorization server can verify the signature that proves possession of the instance key and validate that the authorization code is embedded in the proof. For example, the proof can be a signed message in the JSON Web Token format (JWT), Security Assertion Markup Language (SAML) assertion format, etc.

The signature incorporates the instance certificate fingerprint and the authorization code, and is signed with the instance key. The signature represents a cryptographic attestation that states that: (a) The device or app holds legitimate certificates and keys issued by the client manufacturer; and (b) it is the same instance or unit that the user intended to grant access.

The authorization service returns credentials for the device. The authorization service can return a short timespan credential, e.g., an access token per the OAuth standard, in exchange for the authorization code. While this illustrative example describes use of an OAuth credential, the described techniques can utilize other authorization/authentication protocols that return a single use artifact. The access token can be used by the device to access user resources. The authorization service can also simultaneously issue a long timespan credential, e.g., a refresh token per the OAuth standard, that can be used to obtain additional credentials in the future, e.g., after the original credential has expired. The authorization service can ensure that an authorization code is exchanged only once to provide freshness of device authentication.

The user-specific device credentials created using the disclosed techniques can be used to improve security of long timespan credentials by the creation of an authorization-event specific client credential, and by issuing a key and certificate specific to that authorization. That certificate is signed with the instance key, and is an extension of the instance attestation certificate chain.

This allows the authorization server to cache the authorization-event specific public key. Identifiers for the instance certificate need not be stored on the server. This can lead to less sensitive information being stored on the server, and allows for the authorization-event specific private keys to be deleted from the device, e.g., when the device is reset. This effectively revokes

authorization without involvement of the authorization service. Since the authorization-event specific key is generated dynamically, it is less vulnerable to supply-chain attacks.

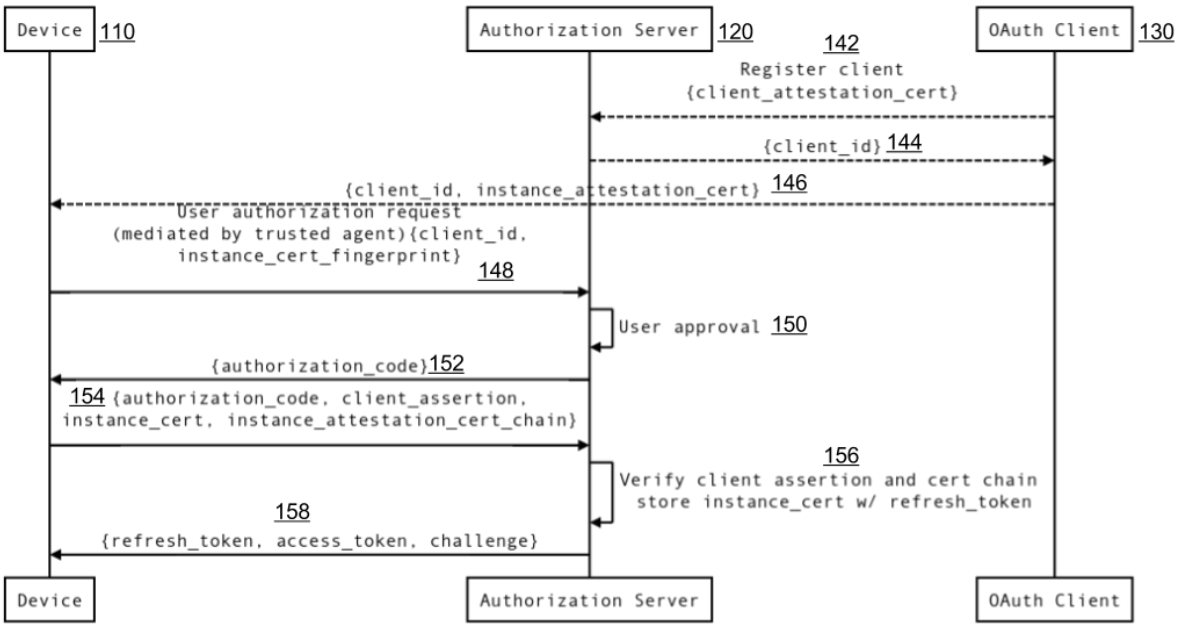

Fig. 1: Device authorization in an OAuth based service

Fig. 1 illustrates the use of the dynamically generated keys in an OAuth standard based service for authorization. The device class or type is registered (130) as a client (142) with an authorization server (120) by the manufacturer of the device. The authorization server provides a client identifier (144) to the manufacturer. Each manufactured device (110) is then assigned an instance attestation certificate chain and the client identifier (146) during manufacture or provisioning.

A user authorization request, along the client identifier and the instance attestation certificate, and mediated by a trusted agent, is transmitted (148) by the device to the authorization server. The client identifier and the instance attestation certificate are verified by the authorization server to successfully approve the user (150). Upon approval, an authorization code (152) which serves as a user credential, is provided to the device.

During subsequent communication, the device can transmit a signed message (154) that includes the authorization code, a client assertion, the instance certificate, and the instance attestation certificate chain. The authorization server verifies (156) the contents of the signed message, and on successful verification, the instance certificate is stored along with a refresh token.

The refresh token, an access token, and an optional challenge is returned (158) to the device. The challenge is used by the client to create subsequent assertions that provide a freshness guarantee by requiring a response from the device.

The described techniques can also incorporate key establishment wherein the device generates key negotiation material, such as a pre-key factor, instead of a new certificate. This can be useful in devices with hardware constraints that can lead to the construction of cryptographically weak certificates. During authorization code exchange, the device assertion can include a reference to the pre-key material. The server then responds with its own pre-key factor, in addition to any tokens. A key establishment process allows the client instance to derive a private key incorporating both sets of pre-key material, in a manner that enables the authorization server to derive the corresponding public key only. The server can bind this newly negotiated key pair to the refresh token for client authentication, as described above. The client instance can record the private key as its authorization-event specific, client key.

The described techniques can be used in hardware such as security cameras, smart home speakers, Internet-connected media playback devices, etc. and are suitable for deployment of Internet-of-Things (IoT) device hubs and other computing devices such as smartphones and other computing appliances. The described techniques can also be adopted as extension of OAuth2, and present a complementary approach to OAuth2 POP draft standard

CONCLUSION

This disclosure describes techniques for user authentication and authorization for devices with support for manufacturer origin attestation. Device attestation is enabled through the use of manufacturer installed keys. User authorization is represented by key pairs generated on demand by the device. The techniques are suitable for deployment on a variety of devices, and can be adapted easily to different standards and protocols for use in authorization and authentication.