Network Working Group
Internet-Draft
Intended status: Informational
Expires: March 22, 2018

P. Hallam-Baker
Comodo Group Inc.

September 18, 2017

Mathematical Mesh: Platform Configuration
draft-hallambaker-mesh-platform-01

Abstract

The Mathematical Mesh ?The Mesh? is an end-to-end secure infrastructure that facilitates the exchange of configuration and credential data between multiple user devices. This document describes how Mesh profiles are stored for application access on Windows, Linux and OSX platforms.

This document is also available online at http://prismproof.org/Documents/draft-hallambaker-mesh-platform.html [1] .

Table of Contents

1. Introduction

This document describes recommended platform specific configuration for Mathematical Mesh applications. The use of common conventions for storage of profiles and private keys allows mesh enabled applications to interoperate on the same machine.

Protecting private key material from disclosure to other processes presents complex and difficult technical challenges. Ensuring that a key is properly erased from storage before memory is released relies on a complex series of assumptions about memory management at the compiler, operating system and the platform level.

For maximum security, the use of private key storage facilities provided by the platform is preferred.

2. Definitions

This section presents the related specifications and standard, the terms that are used as terms of art within the documents and the terms used as requirements language.

2.1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119] .

2.2. Defined Terms

The terms of art used in this document are described in the Mesh Architecture Guide [draft-hallambaker-mesh-architecture] .

2.3. Related Specifications

The architecture of the Mathematical Mesh is described in the Mesh Architecture Guide [draft-hallambaker-mesh-architecture] . The Mesh documentation set and related specifications are described in this document.

2.4. Implementation Status

The implementation status of the reference code base is described in the companion document [draft-hallambaker-mesh-developer] .

3. Windows Platform Configuration

The Windows Configuration is stored in a combination of Windows Key Store entries, registry entries and data files.

The profiles that are available to a user are specified as Windows registry keys.

Cached and archival copies of profiles are stored on the local machine as data files with file names and locations specified in the Windows registry.

Cryptographic keys are stored in a Windows key store.

To locate a device, application or personal profile, an application:

Searches for a Windows registry entry that matches the relevant criteria.

Retrieves the profile data from either a local cached copy or the corresponding portal.

Accesses the corresponding private keys through the Windows key store.

The Windows Key store is the natural storage location for cryptographic keys on the Windows platform as keys are at minimum protected by the operating system access control mechanism. The Windows key store also permits the use of cryptographic hardware devices.

3.1. Registry Key Entries

All keys used by the Mathematical Mesh are stored in the following Windows registry location:

HKEY_CURRENT_USER\SOFTWARE\CryptoMesh

This location has the following sub keys:

PersonalProfiles (Default) -> UDF fingerprint of the default personal profile

PersonalProfiles\<UDF> (Default) -> File location for the profile.

Archive -> File location for the profile archive.

Portals -> Multistring containing portal accounts to which the profile is registered. The default portal is first.

ApplicationProfiles Web -> UDF fingerprint of default Web Application profile

SSH -> UDF fingerprint of default SSH Application profile

Network -> UDF fingerprint of default network Application profile

Mail -> UDF fingerprint of default Mail Application profile

<UDF Fingerprint of profile> -> File location of profile

DeviceProfiles (Default) -> UDF fingerprint of default device profile

<UDF Fingerprint of Device profile> -> File location of device profile

3.2. Data File Locations

ApplicationData \CryptoMesh\

3.3. Key Store Entries

3.4. Profiles

3.4.1. Locating a personal profile

To locate the default personal profile, an application:

Retrieves the key PersonalProfiles\(Default) to get <UDF>

Locates the profile with identifier <UDF>

To locate the personal profile with identifier UDF, an application:

Retrieves the key PersonalProfiles\<UDF>

Retrieves the latest version of the profile from the location specified in PersonalProfiles\<UDF>\(Default)

If necessary, the profile is refreshed from one of the accounts specified in PersonalProfiles\<UDF>\Portal

In case of an inconsistency being detected, the application MAY use the archived copies of the profile to resynchronize.

Note that having been connected to a profile at some time in the past does not guarantee that a device currently has access, even if the device in question was an administration device for the profile.

3.4.2. Locating a device profile

To locate a device profile an application

3.4.3. Locating an application profile

To locate a device profile an application

4. OSX Platform Configuration

The OSX configuration is stored in a combination of a master configuration file, profile data files and the OSX KeyChain

The profiles that are available to a user are stored in a JSON configuration file

Cached and archival copies of profiles are stored on the local machine as data files with file names and locations specified in the JSON configuration file

Cryptographic keys are stored in the OSX Key Chain.

File locations

The JSON Configuration file is stored in ~/.cryptomesh/profiles.json

Profile data files are stored in a directory ~/.cryptomesh/<UDF>

The latest copy of the profile is stored in <UDF>.mmm

An archive containing all the stored profiles is stored in <UDF>.all.mmm

4.1. Key Storage

Private keys are stored in the OSX Key Manager in some fashion to be decided later.

5. Linux Platform Configuration

The Linux configuration is stored in a combination of a master configuration file, profile data files and private key files.

The file layout of the Linux configuration and data files is identical to that of OSX.

5.1. Key Storage

Private Keys are stored in the locations that the Linux applications that are to use them expect to find them.

6. IANA Considerations

None

7. Acknowledgements

TBS

8. References

8.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997.

8.2. Informative References

[draft-hallambaker-mesh-architecture]
Hallam-Baker, P., "Mathematical Mesh: Architecture", draft-hallambaker-mesh-architecture-03 (work in progress), May 2017.

[draft-hallambaker-mesh-developer]
Hallam-Baker, P., "Mathematical Mesh: Reference Implementation", draft-hallambaker-mesh-developer-04 (work in progress), September 2017.

8.3. URIs

[1] http://prismproof.org/Documents/draft-hallambaker-mesh-platform.html

Author's Address

Phillip Hallam-Baker
Comodo Group Inc.

Email: philliph@comodo.com