Network Working Group
Internet Draft
Intended status: Informational
Expires: May 30, 2021

Y. Chen
J. Wang
B. Zhang
Z. Fan
X. Ma
Z. Li
J. Xie

November 30, 2020

China Academy of Information and Communications Technology

Trusted Resolution System and Protocol Extension

draft-chen-trusted-resolution-00

Status of this Memo

Copyright Notice

Abstract

The Handle System [1][2]is a name service system for handle resolution and management over the public Internet. Handle System protocol [3] is designed to be transmitted as a byte stream via a TCP connection. This document describes a Trusted Resolution System and the protocol extension based on Handle System protocol. Trusted resolution aims to achieve credibility verification through data signing. The Trusted Resolution System determines whether to perform trusted resolution and verification on the response according to the trusted flag requested by the client.

Table of Contents

1. Introduction

RFC 3650-RFC 3652[1],[2][3] provide an open protocol, a general-purpose global name service, and a reference implementation of the protocol. In this document, the Trusted Resolution System receives requests from the client and requests to each handle resolution service according to the redirection information to obtain the final response data. The client could choose whether or not to request trusted resolution result when resolving. If the trust-flag in the request is set to 1, the server is expected to return responses including signatures and verifications that would be verified.

2. Conventions used in this document

In this document, handle name and bytes stream are case sensitive, unless otherwise stated.

3. Connection Establishment

In order to send a sequence of bytes stream information based on TCP, a connection between the client and the Trusted Resolution System must be established. The Trusted Resolution System then establishes TCP connections to Handle System respectively.

4. Trusted Resolution Overview

4.1. Trusted Resolution Process

The Handle System is an extensible hierarchical service system, which typically consists of the Global Handle Registry (GHR) and Local Handle Services (LHS).

Trusted resolution is developed to realize credibility verification through data signing and issuing certificates in the Handle System. Signatures and certificates are generated for security purpose. The Trusted Resolution System would verify the signature using the public key available from the service information. By default, handle resolution does not require any trusted resolution.

Figure 1 shows an example of the process of the trusted resolution.

Query: for any given handle, the Trusted Resolution System can query the GHR for its naming authority. The system obtained final handle information after recurse request to the LHS using local service information returned by the GHR.

Build a verification chain: the Trusted Resolution System builds up a chain based on the resolution results.

Verify: the verification is completed by verifying handle values, the verification chain. In each step, the verification results are printed in the log and are reported the client ultimately.

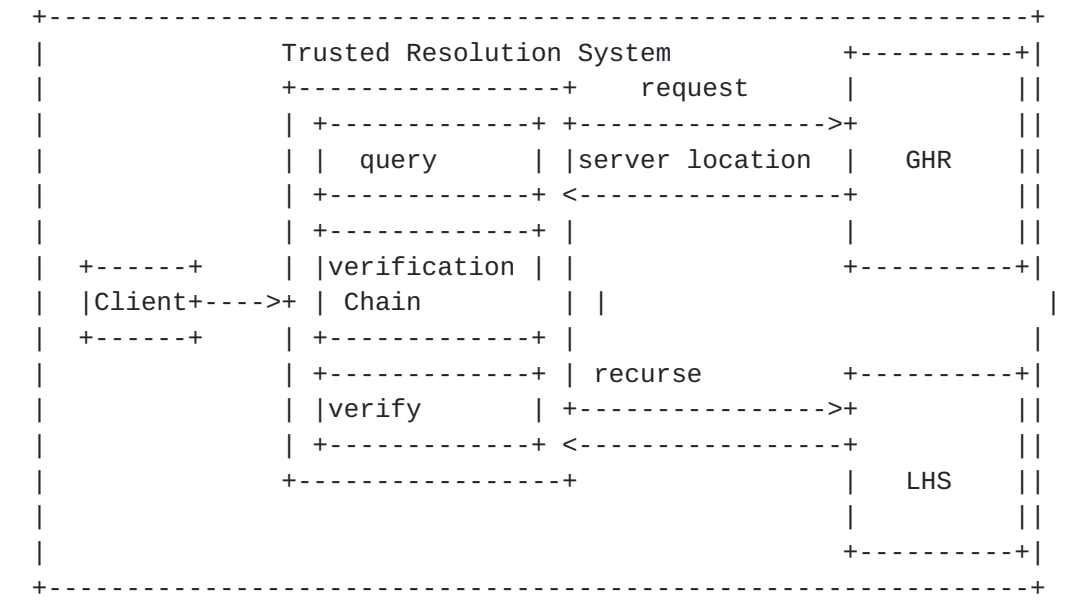


Figure 1 An Example of Trusted Resolution Process

4.2. Trusted Root

Trusted root of should be globally unique within the whole Handle System. Certificate and public key of the trusted root are stored in the configuration files of the Trusted Resolution System. Self-signed certificate of the trust root is generated and verified when the system starts up.

4.3. Trusted Handle

4.3.1. Handle Signatures

Each node in the handle system has a unique pair of public and private key. To ensure that the handles at all levels could be verified, it is necessary to use the private key of the upper node to sign, that is, to generate the signature value.

According to reference Handle System implementation, signatures of handle values are built up and stored in HS_SIGNAURE data type of which data contains a typical JWT(Json Web Token) structure [4].

There are two ways to generate a signature to ensure that no data is tampered:

o Sign handle values partially

o Sign all handle value

Signatures of the handle values are generated in the following process:

1. Get the signer information, private key, Handle, index, etc.;

2. Generate payload from data to be signed which includes handle name, values, singer information, expiration, sign digest, "iss" that represents signer while "sub" represents the handle been signed.

3. Generate the signature data.

4.3.2. Handle Certificates

The PKI (Public Key Infrastructure) system technically solves security issues such as network identity authentication and data the integrity. Public Key Infrastructure is a universal security infrastructure that uses the principles and technologies of asymmetric encryption algorithms to implement and provide security services.

A digital certificate is a combination of a user's identity and the public key held by it. Before the combination, a trusted authority-Certificate Authority (CA) is used to verify the user's identity. The certificate combined with the user's identity and the corresponding public key is digitally signed to prove the validity of the certificate.

The certificates of the top-level naming authority handles, which are managed by GHR, are issued by the trust root. Each naming authority is entitled to issue certificates of its sub-naming authorities. LHS is entitled to manage handles under given sets of naming authorities, and no certificates need to be issued to local handles.

According to reference Handle System implementation, public key and certificates chain information of handle values are built up and stored in HS_CERT data type of which data contains a typical JWT(Json Web Token) structure [4].

Information such as the issuer, the subject, expiration, and authority are defined when issuing. The HS_CERT data type provides a structure to store JWT in its handle data field. JWT is composed of header, payload, and signature, each part of the which is encoded by Base64URLSafe before processed.

Process of generating certificates of a sub-naming authority is as follows:

1. Obtain the issuer information (signer information), namely private key, handle, index of the handle value where the signer's public key is stored.

2. Get certificate information prepared, including information of issuer, expiration time, start time, permissions.

3. Generate the message payload, sign it with the private key of the

signer, and generate the certificate body.

4. Load the public key of the sub-naming authority and re-sign for verification.

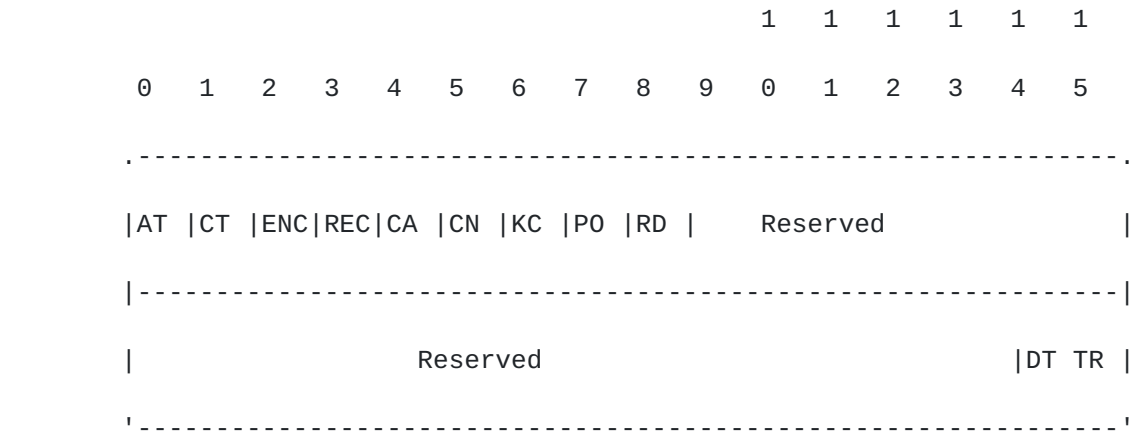
4.4. Signature Algorithms

Trusted Resolution System in the handle system would check that whether the signature algorithm type matches the public key type. Following kind of signature algorithm types are used in the system:

- o RSA-sha256 algorithm is used to generate base64 string in the certification data and signatures.
- o SM2 is a new cryptographic algorithm published by the Chinese State Cryptography Administration, and its encryption strength is 256 bits.
- o SM3 is a cryptographic hash algorithm, the hash value length is 32 bytes. Sm3 algorithm is used when generate digest data of the handle values for the SM2 signing.

5. Trust resolution protocol

The <OpFlag> is a 32-bit bit-mask that defines various control options for protocol operation according to [3]. In addition to the predefined bits, bit30 and bit31 of <OpFlag> are used for trusted resolutions.



DT - bit30 of <OpFlag>, bit that indicates whether to do trusted resolution. DT bit is the trust request flag that indicates whether the client would verify message from the server. If the DT bit is set (to 1), results from the server would be verified. Otherwise, no verification would be performed.

TR - bit31 of <OpFlag>, trusted resolution result bit. TR bit is the trust result flag that indicates whether the message verification succeed. If the TR bit is set (to 1), the response is trusted.

Otherwise, the response verification failed.

5.1. Trusted Query request

The Message Header of any trust query request must set its DT bit of

< OpFlag > (to 1). The default value of this bit is 0 which means that the response would not be checked.

5.2. Successful verification

TR bit of < OpFlag > should be set (to 1) if the response is trusted. Successful verification indicates that the signature of the handle value and certificate matches keypair of the server, the signer of the signature has sufficient permission to sign the handle, and that the signature has not expired.

5.3. Unsuccessful verification

A zero value for TR bit of < OpFlag > indicates that the response fails to pass the trusted resolution. Value set of the requested handle would be returned in response to any handle resolution request whether it is trusted or not.

6. Security Considerations

Data integrity under the protocol is achieved via the server's digital signature. Care must be taken to protect the server's private key from any impersonation attack.

7. IANA Considerations

8. References

8.1. Normative References

[1] Sun, S. and L. Lannom, "Handle System Overview", RFC 3650 November 2003.

[2] Sun, S., Reilly, S. and L. Lannom, "Handle System Namespace and Service Definition", RFC 3651, November 2003.

[3] Sun, S. and L. Lannom, "Handle System Overview", RFC 3652 November 2003.

[4] Jones, et al.,"JSON Web Token (JWT)", RFC 7519, May 2015.

9. Acknowledgments

This document was prepared using 2-Word-v2.0.template.dot. The Trusted Resolution System described in this document relies on works and protocols put forward by RFC 3650, RFC 3651,RFC 3652[1][2][3].

Authors' Addresses

Yuying Chen
CAICT
No.52 Huayuan North Road, Haidian District
Beijing, Beijing, 100191
China

Phone: +86 188 1008 2358
Email: chenyuying@caict.ac.cn

Jiahui Wang
CAICT
No.52 Huayuan North Road, Haidian District
Beijing, Beijing, 100191
China

Phone: +86 186 0156 0021
Email: wangjiahui@caict.ac.cn

Bo Zhang
CAICT
No.52 Huayuan North Road, Haidian District
Beijing, Beijing, 100191
China

Phone: +86 159 1112 3285
Email: zhangbo3@caict.ac.cn

Zhipeng Fan
CAICT

No.52 Huayuan North Road, Haidian District
Beijing, Beijing, 100191
China

Phone: +86 159 1112 3285
Email: fanzhipeng@caict.ac.cn

Xufeng Ma
CAICT
No.52 Huayuan North Road, Haidian District
Beijing, Beijing, 100191
China

Phone: +86 188 1143 3140
Email: maxufeng@caict.ac.cn

Zhiping Li
CAICT
No.52 Huayuan North Road, Haidian District
Beijing, Beijing, 100191
China

Phone: +86 185 1107 1386
Email: lizhiping@caict.ac.cn

Jiagui Xie
CAICT
No.52 Huayuan North Road, Haidian District
Beijing, Beijing, 100191
China

Phone: +86 150 0138 5070
Email: xiejiagui@caict.ac.cn