IEEE SA STANDARDS ASSOCIATION

AcpcTreeId Tutorial: WAVE, Activation Codes for Pseudonym Certificates (ACPC), and the AcpcTreeId

IEEE 1609 Working Group, d0.3, 2020-11-08

General

This tutorial provides an introduction to the terms and concepts related to the AcpcTreeId identifier and its use in a Wireless Access in Vehicular Environments (WAVE) system. For more detailed information, specifications and descriptions regarding the use of this identifier, please see IEEE Std 1609.2.1-2020.

Background: Rationale for ACPC

A Wireless Access in Vehicular Environments (WAVE) system is a radio communication system intended to provide seamless, interoperable services to transportation. These services include those recognized by the U. S. National Intelligent Transportation Systems (ITS) architecture (ARC-IT – The Architecture Reference for Cooperative and Intelligent Transportation, https://local.iteris.com/arc-it/) and many others contemplated by the automotive and transportation infrastructure industries around the world, such as communication between vehicles and infrastructure (V2I), and communication among vehicles (V2V). Related IEEE WAVE standards include IEEE Std 1609.0, IEEE Std 1609.2, IEEE Std 1609.2.1, IEEE Std 1609.3, IEEE Std 1609.4, IEEE Std 1609.11, and IEEE Std 1609.12.

Many applications that use the WAVE system use the security services provided by IEEE Std 1609.2. This is a collection of security mechanisms that includes, among other things, mechanisms for messages to be digitally signed using IEEE 1609.2 certificates. IEEE Std 1609.2 specifies how certificates are to be used by end entities to sign messages; IEEE Std 1609.2.1 provides mechanisms for provisioning and managing those certificates to those end entities.

During the lifetime of the system, the system management may determine that an active end entity (i.e., an end entity with valid certificates) should no longer be trusted by other end entities in the system. In this case, there are three options available to the system:

1. Take no action against the end entity's current certificates but stop issuing new certificates to the end entity. When the end entity's current certificates expire, it ability to sign messages will expire.
2. Revoke the certificates by publishing the certificate identifiers on a Certificate Revocation List (CRL). The CRL is distributed to all end entities in the system that might receive messages from the newly-untrustworthy end entity and acts as instructions to those end-entities not to trust those messages
3. Use the Activation Codes for Pseudonym Certificates (ACPC) mechanism to manage the availability of certificates on the end entity.

Options 1 and 2 have some drawbacks:

- Option 1 allows an end entity to continue to operate until its certificates expire. In some models, this might be for more than a year. As such, Option 1 on its own either leaves the system open to persistent misbehavior or requires short-lived certificates and frequent end-to-end connectivity between end entities and the certificate management system (CMS).
- Option 2 allows prompt removal of end entities. However, the CRLs can grow large in a system with long-term end entity provisioning and a large quantity of end entity compromise. This results in significant expense to store the CRL and to check each certificate against the CRL when that certificate is first processed. CRL usage requires end entities to access broadcast information from the CMS frequently but does not require frequent interactive connectivity between the end entity and the CMS.

Option 3, ACPC in IEEE Std 1609.2.1 has the properties that:

- End entities can be removed from the system quickly
- The system relies entirely on broadcast messages, not on interactive connectivity
- Although large amounts of data are broadcast, only a small amount of data need be stored by each end entity
- Certificates are managed on the send side, not on the receive side; there is no additional burden on the receiver of messages
- The system behavior scales well, even with is a high level of compromise in the system.

An end entity that uses ACPC is issued with certificates that are encrypted and can only be decrypted when the end entity receives an access code. Each access code decrypts only the certificates that are valid for a particular time period. A new access code for each end entity is broadcasted (using a mechanism known as *binary hash trees* to allow for efficient transmission of each individual end entity's access code) shortly before the start of the next time period. Access codes are broadcasted only for those end entities known by the system still to be trustworthy. Since end entities that are believed untrustworthy do not receive access codes to their own certificates they cannot decrypt those certificates for the forthcoming time period and so cannot create signed messages with a valid certificate that would be trusted by the receivers of those messages.

ACPC binary trees and AcpcTreeId

Binary trees. As noted above, the access codes are generated and transmitted using a mechanism known as a binary tree. This binary tree is generated by a component known as the Certificate Access Manager (CAM). While details of the binary tree are not important for this tutorial, functionally the binary tree is equivalent to simply sending out an individual access code for each trusted end entity, and the use of the binary tree as such is simply for bandwidth efficiency; the reader is referred to IEEE Std 1609.2.1 subclauses 9.4 and 9.5 for more details. The use of the binary tree is highlighted in this tutorial because (a) this is where the "tree" in the name "AcpcTreeId" comes from and (b) the ACPC system naturally groups end entities into clusters wherein all the end entities in that cluster get their access code from the same binary tree and no end entities outside that cluster get their access code from that binary tree.

Identifiers for access codes. For an end entity to successfully decrypt its certificates, it requires the correct access code for that end entity for the current time period. Therefore, the access code must be

uniquely identifiable as the one appropriate for that end entity. IEEE Std 1609.2.1 has a hierarchical identifier scheme for access codes:

- An identifier for the binary tree used to generate the access codes is assigned to the CAM that generates that binary tree
- Within the binary tree, the CAM assigns an identifier value to the end entity. This is typically a bit string of length 32-40 bits and serves not only as an identifier but also as an indicator of the path to be taken through the binary tree to determine that end entity's unique access code.

In order for the identifier of the end entity to be globally unique, the identifier of the binary tree is uniquely assigned. This is the AcpcTreeId that is assigned within this registry.

AcpcTreeId Requirements and Management

The AcpcTreeId is required to be globally unique.

A CAM may be assigned multiple AcpcTreeId values.

The AcpcTreeId has no structure: it is simply an eight-byte octet string.

The AcpcTreeId is administered by the IEEE Registration Authority, which is the sole authorized administrator for this identifier space. A small range of AcpcTreeId values are assigned to IEEE Std 1609.2.1 and reserved for allocation by standards action, i.e. by an amendment to or revision of IEEE Std 1609.2.1.

AcpcTreeIdRequests

Each request for an AcpcTreeId assignment sent to the IEEE Registration Authority (RA) will be reviewed before approval. This review process will typically be completed in less than 30 days. All information is kept nonpublic until the request is approved. To be eligible for an AcpcTreeId, an applicant must assert an intent to operate a CAM. An applicant may request up to 16 AcpcTreeId values at a time. If the applicant makes multiple requests totaling more than 16 AcpcTreeId values, the RA may request evidence that the already assigned values are in use and may deny the request for more values if the existing values are not being used.

IEEE does not honor requests for specific AcpcTreeId assignments.

All AcpcTreeId assignments are public unless otherwise requested.

An application form for a new AcpcTreeId assignment may be found at https://standards.ieee.org/products-services/regauth/acpctid/acpctid-application.html.