IEEE P802.11  
Wireless LANs

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Resolution to Comments on Annex ZA4.1 | | | | |
| Date: 2015-01-11 | | | | |
| Author(s): | | | | |
| Name | Company | Address | Phone | email |
| Santosh Abraham | Qualcomm | \ |  | sabraham@qti.qualcomm.com |
| S.K. Yong | Apple |  |  | skyong@apple.com |

Abstract

This document contains resolutions to CIDs 1700, 1457, 1456, 1455, 1414, 1413, 1545, 1544

***Instructions to Editor: Modify Annex ZA.4 as indicated using track changes below.***

**ZA.4: Bloom Filter use in pre-association discovery**(Informative)

**ZA.4.1: Determining the Bloom Filter Size, *m***

Let:

*p* denote the desired false positive probability,

n denote the number of service hashes to be represented in the Bloom filter,

*k* denote the number of Bloom filter hash functions used and

*m* denote the size of the Bloom filter in bits.

The first step is to determine *m*  and *k* given *p* and *n*.

The variables *p*, *n*, *m* and *k* are related to each other with the following approximation [1]

(1)

Given *m* and *n* the value of *k* that yields the lowest false positive probability is given by:

(2)

Substituing (2) in (1) and reording terms, the value of *m* rounded to the neasrest multiple of 8 is given as

(3)

For example, for *n*=25 services and *p*=0.01, the size of the Bloom fitler *m* is 240 bits and the required number of hash function is 7.

**ZA.4.2: Example of Bloom filter created for 25 Service Names**

As given in the example above a 240 bit Bloom filter is computed as follows:

Step 1: Compute the service hash for each service name:

**Table ZA1**

|  |  |
| --- | --- |
| **Service Name** | **Service Hash** |
| print.color.lowresolution | 0x8e6c129fa542 |
| print.color.highresolution | 0xeb73b428f96a |
| print.color.glossy | 0xc8fe546aa1ea |
| print.color.matte | 0x3cddeeb3fc59 |
| print.color.postersize | 0xd7c93fa7a287 |
| books.children.fairytailes | 0x03b70e3e544c |
| books.children.comics | 0x9a9fa4edf3f0 |
| books.teens.romance | 0xf55d38b9dc16 |
| books.teens.sciencefiction | 0x0926ea5d7162 |
| books.teens.comics | 0xdc089f43ee2a |
| movies.drama.hollywood | 0xd7dd9dcec4c2 |
| movies.comedy.hollywood | 0xe0a970abb0ed |
| movies. horror.hollywood | 0x4866f977d754 |
| movies. comedy.indian.bollywood | 0x2f0138cb47da |
| movies.mystery.indian.bollywood | 0xe285ba70ec5e |
| restaurant.thai | 0xf8554b070e7f |
| restaurant.indian | 0xa65b38901845 |
| restaurant.nepali | 0x2366584290b2 |
| restaurant.chinese | 0xd1bed1a18875 |
| restaurant.italian | 0x0b587cb14ac6 |
| sports.soccer.worldcup | 0xdb5cf0ac1954 |
| sports.football.nfl | 0x8154b24e15b9 |
| sports.hockey.nhl | 0xeb7bb0b28ec1 |
| sports.basketball.nba | 0x82bdb7b2fdb7 |
| sports.baseball.mlb | 0x25875583a74b |

Step 2: Map each service hash into the appropriate bits of the the 240 bit (30 octet) Bloom filter using the Bloom filter hash function given in Section 10.25.3.4.4

Resulting Bloom filter (in hex notation) from service hashes in Table ZA1: 0x1c0eba1383b70071658d57de7d7aab3ee1efd9679e1cf2b1bd5d4a456362