# Network Protocols VLAN and EAP

## EFI\_VLAN\_CONFIG\_PROTOCOL Test

Reference Document:

*UEFI Specification*, EFI\_VLAN\_CONFIG\_PROTOCOL Section.

### Set()

|  |  |  |  |
| --- | --- | --- | --- |
| Number | GUID | Assertion | Test Description |
| 5.27.1.1.1 | 0xedbb5f4f, 0x4de7, 0x43ff, 0x82, 0x1c, 0x13, 0x80, 0x98, 0x95, 0xd1, 0x76 | EFI\_VLAN\_CONFIG\_PROTOCOL.SET - SET() returns EFI\_INVALID\_PARAMETER with an invalid VlanId. | Call Set() with valid parameters except an invalid VlanId, The return status should be EFI\_INVALID\_PARAMETER. |
| 5.27.1.1.2 | 0x9c1292c2, 0xe03a, 0x438d, 0x9d, 0xab, 0x4e, 0xd0, 0xa9, 0xa8, 0xb6, 0x83 | EFI\_VLAN\_CONFIG\_PROTOCOL.SET - SET() returns EFI\_INVALID\_PARAMETER with an invalid Priority. | Call Set() with valid parameters except an invalid Priority, The return status should be EFI\_INVALID\_PARAMETER. |
| 5.27.1.1.3 | 0xe3584990, 0x0b04, 0x48ea, 0x96, 0x3d, 0x36, 0xf7, 0x62, 0x29, 0x9f, 0x42 | EFI\_VLAN\_CONFIG\_PROTOCOL.SET - SET() returns EFI\_SUCCESS with a valid VlanId and a valid Priority. | 5.27.1.1.3 – 5.27.1.1.6 belong to one case  1. Call Set()with a valid VlanId and a valid Priority, The return status should be EFI\_SUCCESS. |
| 5.27.1.1.4 | 0xc14eb533, 0xc076, 0x4a9e, 0xb5, 0x6d, 0xea, 0x00, 0xce, 0xac, 0x7b, 0x2d | EFI\_VLAN\_CONFIG\_PROTOCOL.Find - Find() returns EFI\_SUCCESS with the same VlanId, a valid NumberOfVlan and a valid Priority. | 2. Call Find()with the same VlanId, a valid NumberOfVlan and a valid Priority. The return status should be EFI\_SUCCESS. The NumberOfVlan should be 1. The output VlanId and Priority in the Entries should be the same value of VlanId/Priority which are set in step1. |
| 5.27.1.1.5 | 0x48deb1ad, 0xd59b, 0x404e, 0x88, 0xe7, 0x42, 0x53, 0xac, 0x0e, 0xce, 0x22 | EFI\_VLAN\_CONFIG\_PROTOCOL.SET - SET() returns EFI\_SUCCESS with the same VlanId and a different Priority. | 3. Call Set()with the same VlanId and a different Priority, The return status should be EFI\_SUCCESS. |
| 5.27.1.1.6 | 0x98f1580a, 0xa2b6, 0x4e61, 0x8b, 0xc9, 0x31, 0xb1, 0xac, 0xb0, 0x20, 0xb5 | EFI\_VLAN\_CONFIG\_PROTOCOL.Find - Find() returns EFI\_SUCCESS with the same VlanId, a valid NumberOfVlan and a valid Priority. | 4. Call Find()with the same VlanId, a valid NumberOfVlan and a valid Priority. The return status should be EFI\_SUCCESS. The NumberOfVlan should be 1. The output VlanId and Priority in the Entries should be the same value of VlanId/Priority which are set in step2. |

### Find()

|  |  |  |  |
| --- | --- | --- | --- |
| Number | GUID | Assertion | Test Description |
| 5.27.1.2.1 | 0x07f07b52, 0x93e8, 0x43fe, 0xa7, 0x84, 0x01, 0x71, 0x02, 0x37, 0x66, 0x82 | EFI\_VLAN\_CONFIG\_PROTOCOL.Find - Find() returns EFI\_INVALID\_PARAMETER with an invalid VlanId. | Call Find() with valid parameters except an invalid VlanId, The return status should be EFI\_INVALID\_PARAMETER. |
| 5.27.1.2.2 | 0xedb0b22d, 0xa6b5, 0x497d, 0xbb, 0x9b, 0x75, 0x85, 0x47, 0x11, 0x35, 0xc5 | EFI\_VLAN\_CONFIG\_PROTOCOL.Find - Find() returns EFI\_INVALID\_PARAMETER with NumberOfVlan been NULL. | Call Find() with valid parameters except NumberOfVlan being NULL, The return status should be EFI\_INVALID\_PARAMETER. |
| 5.27.1.2.3 | 0x57d7d76b, 0x6b88, 0x44c9, 0x85, 0x0c, 0x0f, 0xb6, 0xcb, 0xe1, 0x9c, 0xd9 | EFI\_VLAN\_CONFIG\_PROTOCOL.Find - Find() returns EFI\_INVALID\_PARAMETER with Entries been NULL. | Call Find() with valid parameters(a valid NumberOfVlan and an NULL VlanId) except Entries being NULL, The return status should be EFI\_INVALID\_PARAMETER. |
| 5.27.1.2.4 | 0x1de50bab, 0x1f3a, 0x4c62, 0x82, 0x93, 0x6e, 0x9a, 0x11, 0x03, 0xce, 0x6f | EFI\_VLAN\_CONFIG\_PROTOCOL.Find - Find() returns EFI\_INVALID\_PARAMETER with Entries been NULL. | Call Find() with valid parameters(a valid NumberOfVlan and a valid VlanId) except Entries being NULL, The return status should be EFI\_INVALID\_PARAMETER. |
| 5.27.1.2.5 | 0x49d1f535, 0x3b53, 0x4892, 0x98, 0x02, 0x5c, 0x19, 0xa3, 0x6d, 0xd1, 0x53 | EFI\_VLAN\_CONFIG\_PROTOCOL.Find - Find() returns EFI\_SUCCESS or EFI\_NOT\_FOUND with valid parameters. | Call Find() with valid parameters(a valid NumberOfVlan, a valid Entries and NULL VlanId), The return status should be EFI\_SUCCESS or EFI\_NOT\_FOUND. If EFI\_SUCCESS. NumberOfVlan should be greater than 0 and Entries should not be NULL. If EFI\_NOT\_FOUND, NumberOfVlan should be 0 and Entries should be NULL. |
| 5.27.1.2.6 | 0x2f95fed6, 0xed1b, 0x4ac0, 0x9a, 0xa8, 0x97, 0x81, 0x76, 0xac, 0xcf, 0x8d | EFI\_VLAN\_CONFIG\_PROTOCOL.Find - Find() returns EFI\_SUCCESS with valid parameters. | 5.27.1.2.6 – 5.27.1.2.7 belong to one case  1. Call Set() to config a Vlan  2. Call Find() with the same VlanId The return status should be EFI\_SUCCESS. The NumberOfVlan should be 1. The output VlanId and Priority in the Entries should be the same value of VlanId/Priority which are set in step 1. |
| 5.27.1.2.7 | 0xf4d6c7d9, 0x21bf, 0x48b5, 0xb0, 0x1d, 0x10, 0xf4, 0xfa, 0x11, 0xf7, 0x8d | EFI\_VLAN\_CONFIG\_PROTOCOL.Find - Find() returns EFI\_NOT\_FOUND with valid parameters. | 3. Call Remove() to delete the same VlanId  4. Call Find() with the same VlanId The return status should be EFI\_NOT\_FOUND. The NumberOfVlan should be 0. The Entries should be NULL. |

### Remove()

|  |  |  |  |
| --- | --- | --- | --- |
| Number | GUID | Assertion | Test Description |
| 5.27.1.3.1 | 0x1adaa7a4, 0xd1d3, 0x49d5, 0x97, 0xb4, 0xe4, 0x0f, 0x63, 0x1b, 0x68, 0xd0 | EFI\_VLAN\_CONFIG\_PROTOCOL.Remove - Remove() returns EFI\_INVALID\_PARAMETER with an invalid VlanId. | Call Remove()with valid parameters except an invalid VlanId, The return status should be EFI\_INVALID\_PARAMETER. |
| 5.27.1.3.2 | 0xaa94b834, 0xf247, 0x4530, 0xb0, 0x6a, 0x49, 0x4e, 0x10, 0x37, 0xb5, 0xe5 | EFI\_VLAN\_CONFIG\_PROTOCOL.Remove - Remove() returns EFI\_NOT\_FOUND with an not set VlanId. | Call Remove()with valid parameters except an not set VlanId, The return status should be EFI\_NOT\_FOUND. |
| 5.27.1.3.3 | 0x30991f39, 0x7410, 0x46ed, 0xa5, 0xe6, 0xdb, 0xc9, 0xf7, 0x86, 0x4f, 0xd3 | EFI\_VLAN\_CONFIG\_PROTOCOL.Remove - Remove() returns EFI\_SUCCESS with valid parameters. | 5.27.1.3.3 – 5.27.1.3.4 belong to one case  1. Call Set() to configure a VlanId  2. Call Remove() with the same VlanId. The return status should be EFI\_SUCCESS. |
| 5.27.1.3.4 | 0x28b96fd8, 0xc729, 0x4906, 0xa6, 0xbd, 0xda, 0xe4, 0xd0, 0x2a, 0x82, 0x1e | EFI\_VLAN\_CONFIG\_PROTOCOL.Remove - Remove() returns EFI\_NOT\_FOUND with valid parameters. | 3. Call Remove() with the same VlanId. The return status should be EFI\_NOT\_FOUND. |