

APPLICATION NOTE

MQTT communication APIs

1 개요

2 소개

본 문서는 ACKSYS WaveManager 또는 MQTT 브로커/클라이언트 구조의 네트워크 서비스에서 WaveOS 기반 ACKSYS 제품을 모니터링하고, 설정을 변경하거나 테스트 명령을 실행할 때 사용하는 MQTT API를 설명합니다.

3 사전 요구 사항

WaveOS 4.28.1.1 이상을 사용하는 ACKSYS 제품

4 일반 사항

설정, 통계 및 명령은 JSON 형식으로 전송합니다.

MQTT 메시지 크기를 줄이기 위해 값이 없는 필드는 전송하지 않습니다. DHCP 모드의 IP 주소처럼 정보가 없거나, 무선 통계의 SignalQuality처럼 향후 사용을 위해 정의된 필드가 이에 해당합니다. 단, 값이 빈 문자열인 필드는 반드시 전송해야 합니다.

5 장비 설정 전송

장비 설정은 다음 두 단계로 전송합니다.

1. 설정 상태

장비의 설정 상태는 장비의 정적 정보와 각 설정 구성 요소의 상태를 나타냅니다. 시스템, 물리 인터페이스, 네트워크, 무선 및 통계 설정의 상태는 반환된 JSON 데이터의 해시값으로 표시합니다.

장비는 다음 경우에 MQTT 브로커로 설정 상태를 전송합니다.

- 장비 시작 시
- 장비 설정 변경 시
- WaveManager 서비스 또는 MQTT 클라이언트의 설정 상태 요청 수신 시

장비의 설정 상태는 JSON 형식으로 전송하며, 데이터 항목은 다음과 같습니다.

항목	설명	자료형/형식
ProductId	장비 고유 식별자	String
Code	장비 제품 코드	Int
SerialNumber	장비 일련번호	String NNNNNNNN /
DefaultIPAddress	제품에 접근할 수 있는 IPv4 또는 IPv6 주소입니다. 기본 게이트웨이에서 사용하는 IP 주소이며, 기본 게이트웨이가 여러 개이면 메트릭이 가장 낮은 경로를 사용합니다.	String / N.N.N.N or XXXX:XXXX:XX XX:XXXX:XXXX :XXXX:XXXX: XXXX
GlobalConfigurationHash	장비 설정 파일의 해시값	String
SystemConfigurationHash	반환된 시스템 데이터의 해시값	String
PhysicalInterfaceConfigurationsHash	반환된 물리 인터페이스 데이터의 해시값	String
CapabilityConfigurationHash	국가 코드에 따른 무선 기능·지원 사양 데이터의 해시값	String
WirelessConfigurationsHash	반환된 무선 데이터의 해시값	String
NetworkConfigurationsHash	반환된 네트워크 데이터의 해시값	String
StatusServiceConfigurationHash	반환된 통계 서비스 데이터의 해시값	String
ProtocolVersion	통신 프로토콜 버전	String

5.2 설정 데이터

WaveManager 서비스는 하나 이상의 해시값이 변경되면 해당 구성 요소의 데이터를 요청합니다. MQTT 클라이언트에서도 이 데이터를 요청할 수 있습니다.
설정 데이터는 다음 JSON 형식으로 요청합니다.

```
'ExpectedConfigurationData' : [
    {'SystemConfiguration'},
    {'WirelessConfigurations'}
]
```

조회할 구성 요소의 식별자를 목록으로 전송합니다. 각 설정 구성 요소의 식별자는 다음과 같습니다.

구성 요소	식별자
전체 설정	AllConfigurations
시스템 설정	SystemConfiguration
물리 인터페이스 설정	PhysicalInterfaceConfigurations
무선 설정	WirelessConfigurations
국가 코드별 무선 기능·지원 사양	CapabilityConfiguration
네트워크 설정	NetworkConfigurations
통계 서비스 설정	StatusServiceConfiguration

장비는 요청받은 설정 구성 요소의 데이터를 다음 JSON 형식으로 전송합니다.

```
{
  'ProductId' : 'router_id', 'SystemConfiguration' : {
    'system_data'
  },
  'WirelessConfigurations' : [
    {'wireless_data'},
    ....
  ],
  등
}
```

5.2.1 시스템 설정 데이터

시스템 설정은 다음 JSON 형식으로 반환합니다.

```
'SystemConfiguration' : {
  'system_data'
}
```

“system_data”로 반환하는 시스템 데이터는 다음과 같습니다.

항목	설명	자료형	허용 값
Hostname	장비 이름	String	
Version	장비 버전	String	
Description	장비 설명(설치 위치)	String	
FirmwareId	장비 펌웨어 ID	String	
FirmwareVersion	장비 펌웨어 버전	String	
FirmwareExtension	장비 펌웨어 확장 정보	String	
NTPServerNames	NTP 서버 목록	[String]	
NTPServerPort	NTP 서버 포트	Int	

5.2.2 물리 인터페이스 데이터

물리 인터페이스 설정은 다음 JSON 형식으로 반환합니다.

```
'PhysicalInterfaceConfigurations' : [
  {'itf_1_data'},
  {'itf_2_data'}
]
```

물리 인터페이스 설정은 인터페이스 목록으로 반환합니다. 다음 표는 물리·무선 인터페이스의 데이터 구조입니다(« itf_1_data », « itf_2_data » 등).

항목	설명	자료형/형식	허용 값
Name	인터페이스 이름입니다. 네트워크 설정 및 무선 역할과 연결하는 인터페이스 식별자로 사용합니다.	String	radio0, radio1, eth0, eth1, wwan0, wwan0-gnss, etc.
Type	물리 인터페이스 유형	Int/Enumeration	Unknown = 0,

항목	설명	자료형/형식	허용 값
			Ethernet = 1, Wireless = 2, Cellular = 3, GNSS = 4
Label	인터페이스 레이블	String	
State	인터페이스 상태	Int/Enumeration	Disabled = 0, Enabled = 1
Band	무선 인터페이스의 사용 대역입니다. 단일 대역 지원은 11AC Wave2 카드부터 제공됩니다.	Int/Enumeration	Unknown = 0, Band_24 = 1, Band_5 = 2, Band_6 = 3
WifiMode	무선 인터페이스의 802.11 모드	Int/Enumeration	None = 0, B_Only = 1, G_Only = 2, Mixed_B_G = 3, A_Only = 4, N_G = 5, N_BG = 6, N_A = 7, AC = 8, AX = 9
ChannelWidth	무선 인터페이스의 채널 대역폭	Int/Enumeration	None = 0, HT20 = 1, HT40_Below = 2, HT40_Above = 3, HT40 = 4, VHT20 = 5, VHT40 = 6, VHT80 = 7, VHT80_80 = 8, VHT160 = 9, HE20 = 10, HE40 = 11, HE80 = 12, HE80_80 = 13, HE160 = 14
PrimaryWideSegment	주 무선 세그먼트	Int/Enumeration	Channel_36 = 36, Channel_40 = 40, Channel_44 = 44, Channel_48 = 48, Channel_XX = XX, etc
PrimaryChannel	무선 인터페이스의 주 채널	Int/Enumeration	Auto_Channel = 0, Channel_36 = 36, Channel_40 = 40, Channel_44 = 44, Channel_48 = 48, Channel_XX = XX, etc
SecondaryWideSegment	보조 무선 세그먼트	Int/Enumeration	Channel_36 = 36, Channel_40 = 40, Channel_44 = 44, Channel_48 = 48, Channel_XX = XX, etc.
MaximalTransmitPower_dBm	무선 인터페이스의 최대 송신 출력입니다. null이면 무선 카드의 최대 출력을 사용합니다.	Int	

항목	설명	자료형/형식	허용 값
MTU_O	최대 전송 단위(MTU): 인터페이스에서 송수신할 수 있는 최대 패킷 크기(바이트)	Int32	
MACAddress	인터페이스의 MAC 주소입니다. 이더넷 및 Wi-Fi 인터페이스에만 반환합니다.	String XX:XX:XX: XX:XX:XX	

5.2.3 무선 기능·지원 사양 데이터

무선 인터페이스의 지원 사양은 다음 JSON 형식으로 반환합니다.

```
'CapabilityConfiguration' : {
  'wireless_capabilities_data'
}
```

"wireless_capabilities_data"로 반환하는 무선 지원 사양 데이터는 다음과 같습니다.

항목	설명	자료형	허용 값
CountryCode	ISO 3166-1 국가 코드	String	
WirelessCapabilities	무선 지원 사양 목록	[{'wireless_capabilities_1_data'}, {'wireless_capabilities_2_data'}]	

각 무선 인터페이스별로 지원 사양을 나열합니다. 각 항목의 데이터 형식은 다음과 같습니다.
« wireless_capabilities_1_data », « wireless_capabilities_2_data », etc.

항목	설명	자료형	허용 값
PhysicalInterfaceName	연결된 물리 인터페이스 이름	String	radio0, radio1, etc.
SupportedBands	무선 인터페이스에서 사용 가능한 주파수 대역 목록	[{'supported_band_1_data'}, {'supported_band_2_data'}]	

지원하는 각 대역의 상세 정보는 다음과 같습니다(« supported_band_1_data », « supported_band_2_data » 등).

항목	설명	자료형/열거형	허용 값
Band	무선 인터페이스에서 사용 가능한 주파수 대역: 단일 대역 지원은 11AC Wave2 카드부터 제공됩니다.	Int/Enumeration	Unknown = 0, Band_24 = 1, Band_5 = 2, Band_6 = 3
Description	사용 가능한 주파수 대역 설명	String	
SupportedWifiModes	지정한 대역에서 무선 인터페이스가 사용할 수 있는 Wi-Fi 모드 목록	[{'supported_wifi_mode_1_data'}, {'supported_wifi_mode_2_data'}]	

지원 대역 내 각 Wi-Fi 모드의 데이터 구조는 다음과 같습니다. « supported_wifi_mode_1_data », « supported_wifi_mode_2_data », etc.

항목	설명	자료형/열거형	허용 값
WifiMode	무선 인터페이스에서 사용 가능한 Wi-Fi 모드	Int/Enumeration	None = 0, B_Only = 1, G_Only = 2, Mixed_B_G = 3, A_Only = 4, N_G = 5, N_BG = 6, N_A = 7, AC = 8, AX = 9
Description	사용 가능한 Wi-Fi 모드 설명	String	
SupportedChannelWidths	지정한 주파수 대역과 Wi-Fi 모드에서 사용 가능한 채널 대역폭 목록	[{'supported_channel_width_1_data'}, {'supported_channel_width_2_data'}]	

지정한 주파수 대역과 Wi-Fi 모드에서 각 채널 대역폭의 데이터는 다음과 같습니다(« supported_channel_width_1_data », « supported_channel_width_2_data » 등).

항목	설명	자료형/열거형	허용 값
ChannelWidth	무선 인터페이스의 채널 대역폭	Int/Enumeration	None = 0, HT20 = 1, HT40_Below = 2, HT40_Above = 3, HT40 = 4, VHT20 = 5, VHT40 = 6, VHT80 = 7, VHT80_80 = 8, VHT160 = 9, HE20 = 10, HE40 = 11, HE80 = 12, HE80_80 = 13, HE160 = 14
Description	사용 가능한 채널 대역폭 설명	String	
SupportedPrimaryChannels	지정한 주파수 대역, Wi-Fi 모드 및 채널 대역폭에서 사용 가능한 주 채널 목록	[{'supported_channel_1_data'}, {'supported_channel_2_data'}]	
SupportedPrimaryWideSegments	지정한 주파수 대역, Wi-Fi 모드 및 채널 대역폭에서 사용 가능한 주 세그먼트 목록	[{'supported_segment_1_data'}, {'supported_segment_2_data'}]	

지정한 주파수 대역, Wi-Fi 모드 및 채널 대역폭에서 각 채널의 데이터는 다음과 같습니다(« supported_channel_1_data », « supported_channel_2_data » 등).

항목	설명	자료형/열거형	허용 값
Channel	무선 인터페이스의 채널 값	Int/Enumeration	Auto_Channel = 0, Channel_36 = 36, Channel_40 = 40, Channel_44 = 44, Channel_48 = 48, Channel_XX = XX, etc .
Description	사용 가능한 채널 설명입니다. 주파수, 최대 송신 출력 및 DFS 지원 정보를 포함합니다.	String	
MaximalTransmitPower_dBm	채널의 최대 송신 출력	Int	
Frequency_GHz	채널 주파수	Double	
DFSSState	DFS 지원 여부	Int/Enumeration	NotSupported = 0, Supported = 1

지정한 주파수 대역, Wi-Fi 모드 및 채널 대역폭에서 각 주 세그먼트의 데이터는 다음과 같습니다(« supported_segment_1_data », « supported_segment_2_data » 등).

항목	설명	자료형/열거형	허용 값
Channel	무선 인터페이스의 채널 값	Int/Enumeration	Auto_Channel = 0, Channel_36 = 36, Channel_40 = 40, Channel_44 = 44, Channel_48 = 48, Channel_XX = XX, etc.
Description	채널 설명: 주파수, 최대 송신 출력 및 DFS 지원 정보 포함	String	
MaximalTransmitPower_dBm	채널의 최대 송신 출력	Int	
Frequency_GHz	채널 주파수	Double	
DFSSState	DFS 지원 여부	Int/Enumeration	NotSupported = 0, Supported = 1
SupportedPrimarySegmentChannels	지정한 주파수 대역, Wi-Fi 모드, 채널 대역폭 및 세그먼트에서 사용 가능한 채널 목록	[{'supported_segment_channel_1_data'}, {'supported_segment_channel_2_data'},]	

지정한 주파수 대역, Wi-Fi 모드, 채널 대역폭 및 주 세그먼트에서 각 채널의 데이터는 다음과 같습니다(« supported_segment_channel_1_data », « supported_segment_channel_2_data » 등).

항목	설명	자료형/열거형	허용 값
Channel	무선 인터페이스의 채널 값	Int/Enumeration	Auto_Channel = 0, Channel_36 = 36, Channel_40 = 40, Channel_44 = 44, Channel_48 = 48, Channel_XX = XX, etc.

항목	설명	자료형/열거형	허용 값
Description	채널 설명: 주파수, 최대 송신 출력 및 DFS 지원 정보 포함	String	
MaximalTransmitPower_dBm	채널의 최대 송신 출력	Int	
Frequency_GHz	채널 주파수	Double	
DFSState	DFS 지원 여부	Int/Enumeration	NotSupported = 0, Supported = 1
SupportedSecondaryWideSegments	지정한 주파수 대역, Wi-Fi 모드, 채널 대역폭, 주 세그먼트 및 주 채널에서 사용 가능한 보조 세그먼트 목록	[{'supported_secondary_segment_1_data'}, {'supported_secondary_segment_2_data'}]	

지정한 주파수 대역, Wi-Fi 모드 및 채널 대역폭에서 각 보조 세그먼트의 데이터는 다음과 같습니다.
« supported_secondary_segment_1_data », « supported_secondary_segment_2_data », etc.

항목	설명	자료형/열거형	허용 값
Channel	무선 인터페이스의 채널 값	Int/Enumeration	Auto_Channel = 0, Channel_36 = 36, Channel_40 = 40, Channel_44 = 44, Channel_48 = 48, Channel_XX = XX, etc.
Description	채널 설명: 주파수, 최대 송신 출력 및 DFS 지원 정보 포함	String	
MaximalTransmitPower_dBm	채널의 최대 송신 출력	Int	
Frequency_GHz	채널 주파수	Double	
DFSState	DFS 지원 여부	Int/Enumeration	NotSupported = 0, Supported = 1

5.2.4 무선 설정 데이터

장비의 무선 설정은 다음 JSON 형식으로 반환합니다.

```
'WirelessConfigurations': [
    {'wireless_1_data'},
    {'wireless_2_data'}
]
```

각 무선 설정의 데이터는 다음과 같습니다(« wireless_1_data », « wireless_2_data » 등).

항목	설명	자료형	허용 값
PhysicalInterfaceName	연결된 물리 인터페이스 이름	String	radio0, radio1, etc.
Name	무선 인터페이스 이름: 이름이 동적으로 생성되므로	String	Radio0w0, radio1w0, etc.

항목	설명	자료형	허용 값
	무선 인터페이스 이름으로 radio 섹션 이름을 사용합니다.		
Role	무선 역할: "Monitor" 역할은 이 버전에서 비활성화됩니다.	Int/Enumeration	Undefined = 0, Client = 1, AccessPoint = 2, Adhoc = 3, Monitor = 5, Mesh = 6, Repeater = 7, IsolatingAccessPoint = 8, RogueAP = 9, SRCC = 10
Ssid	서비스 세트 식별자(SSID)	String	
SecurityMode	보안 모드	Int/Enumeration	NONE = 0, WPA-PSK = 1, WPA2-PSK = 2, WPA3-PSK = 3, MIXED_WPA_WPA2_PSK = 4, MIXED_WPA2_WPA3_PSK = 5, WPA_EAP = 6, WPA2_EAP = 7, WPA3_EAP = 8, MIXED_WPA2_WPA3_EAP = 9, WPA3_OWE = 10, OSEN = 11, WEP_OPEN_SYSTEM = 12, WEP_SHARED_KEY = 13
SecurityKey	보안 키	String	

5.2.5 네트워크 설정 데이터

장비의 네트워크 설정은 다음 JSON 형식으로 반환합니다.

```
'NetworkConfigurations': [
    {'network_1_data'},
    {'network_2_data'}
]
```

각 네트워크 설정의 데이터는 다음과 같습니다(« network_1_data », « network_2_data » 등).

항목	설명	자료형	허용 값
Name	네트워크 식별자로 사용하는 인터페이스 이름	String	lan, net1, net2, etc.
Label	네트워크 레이블	String	AcksysLan, Lan1, etc.
State	네트워크 상태	Int/Enumeration	Disabled = 0, Enabled = 1
IPv4Mode	IPv4 모드	Int/Enumeration	None = 0, Static = 1, Dhcp = 2, VRRP = 3
IPv4Addresses	IPv4 주소 목록입니다. v4.0.0.1에서는 첫 번째 IPv4 주소만 반환하며, 고정 IP 모드에서만 제공합니다.	[{'ipv4_address_1_data'}, {'ipv4_address_2_data'}]	
IPv4Gateway	IPv4 게이트웨이(고정 IP 모드에서만 반환)	String / N.N.N.N	

항목	설명	자료형	허용 값
IPv4DNSAddresses	DNS 서버 목록(고정 IP 모드에서만 반환)	[String] / N.N.N.N	
IPv6Mode	IPv6 모드	Int/Enumeration	None = 0, Static = 1, SLAAC = 2, Dhcpv6 = 3
IPv6Addresses	IPv6 설정 목록(고정 IP 모드에서만 반환)	[{'ipv6_address_1_data'}, {'ipv6_address_2_data'}]	
IPv6Gateway	IPv6 게이트웨이(고정 IP 모드에서만 반환)	String XXXX:XXXX:XXXX:XX XX:XXXX:XXXX:XXXX: XXXX	/
AssociatedInterfaceNames	이 네트워크 인터페이스에 연결된 이더넷 및 무선 물리 인터페이스 목록	[String]	

선택한 설정에 따른 IPv4 및 IPv6 모드는 다음과 같습니다.

선택 모드	IPv4Mode	IPv6Mode
none	none	none
static	static	static
Dhcp	Dhcp	none
VRRP	VRRP	none
SLAAC	none	SLAAC
Dhcpv6	none	Dhcpv6

IPv4 설정의 데이터 구조는 다음과 같습니다.
« ipv4_address_2_data » 등)

(« ipv4_address_1_data »,

항목	설명	자료형	허용 값
IPv4Address	IPv4 주소	String / N.N.N.N	
IPv4Mask	IPv4 서브넷 마스크	String / N.N.N.N	

IPv6 설정의 데이터 구조는 다음과 같습니다.
« ipv6_address_2_data » 등)

(« ipv6_address_1_data »,

항목	설명	자료형	허용 값
IPv6Address	IPv6 주소 값	String / XXXX:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX	
IPv6PrefixLength	IPv6 접두사 길이	Int	

5.2.6 텔레메트리/상태 설정 데이터

장비의 텔레메트리 설정은 다음 JSON 형식으로 반환합니다.

```
'StatusServiceConfiguration': {
  'status_service_data'
}
```

« status_service_data »로 표시하는 통계 서비스 데이터는 다음과 같습니다.

항목	설명	자료형	허용 값
SamplingInterval_S	데이터 수집 주기(초)	Int	기본값: 10
OutputInterval_S	데이터 전송 주기(초)	Int	기본값: 10
WirelessCollectionState	무선 정보 수집 플러그인 상태	Int/Enum	Disabled = 0, Enabled = 1; 기본값: 활성화(1)
RoamingCollectionState	로밍 정보 수집 플러그인 상태	Int/Enum	Disabled = 0, Enabled = 1; 기본값: 비활성화(0)
GNSSCollectionState	GNSS 정보 수집 플러그인 상태	Int/Enum	Disabled = 0, Enabled = 1; 기본값: 활성화(1)
GNSSServerIPAddress	GNSS 서버 IP 주소	String / N.N.N.N	장비별로 다름
GNSSServerPort	GNSS 서버 포트	Int	장비별로 다름
CellularCollectionState	셀룰러 정보 수집 플러그인 상태	Int/Enum	Disabled = 0, Enabled = 1; 기본값: 활성화(1)
SyslogCollectionState	Syslog 수집 플러그인 상태	Int/Enum	Disabled = 0, Enabled = 1; 기본값: 비활성화(0)

5.3 통계 전송

장비는 SamplingInterval_S 주기로 텔레메트리 통계를 수집하고, OutputInterval_S 주기로 WaveManager 서비스 또는 MQTT 클라이언트에 전송합니다.

MQTT에서는 데이터 전송 순서가 보장되지 않습니다. 텔레메트리 데이터를 순서대로 정렬할 수 있도록 각 메시지에 순서 정보를 추가합니다.

텔레메트리 데이터는 다음 JSON 형식으로 전송합니다.

```
{
  "ProductId" : "router_id", "MessageOrder" : "order", "Data" : [
    {"Type" : 1, bandwidth_data},
    {"Type" : 2, wireless_data},
    {"Type" : 3, roaming_data},
    {"Type" : 7, cellular_data}
  ]
}
```

« MessageOrder »는 실시간 그래프 표시 시 메시지 순서를 관리하는 항목입니다. 자료형은 uint8이며, MQTT 서비스 초기화 시(재부팅 또는 설정 변경 등) 0부터 시작합니다. uint8의 최댓값에 도달한 후에는 다시 0으로 돌아갑니다.

사용하는 통계 유형은 다음과 같습니다.

식별자	설명	열거형 코드
BandwidthStatus	대역폭 상태	1
WirelessStatus	무선 상태(RSSI)	2
RoamingStatus	로밍 상태	3

식별자	설명	얼거형 코드
AssociationStatus	연결 상태	4
MeshLinkStatus	메시 링크 상태 *미지원	5
GNSSStatus	GNSS 상태	6
CellularStatus	셀룰러 상태	7
SystemStatus	동적 시스템 정보 상태	8
NetworkStatus	네트워크 인터페이스 상태	9
DiscoveryStatus	네트워크 검색 상태	10
SyslogStatus	Syslog 상태	11

5.3.1 대역폭 상태

대역폭 통계의 데이터 구조는 다음과 같습니다.

항목	설명	자료형/형식	허용 값
InterfaceName	인터페이스 이름	String	radio0w0, radio1w0, eth0, eth1, wwan0, etc.
InterfaceType	인터페이스 유형	Int/Enumeration	Unknown = 0, Ethernet = 1, Wireless = 2, Cellular = 3, Network = 5
DateTime_UTC	기록 날짜 및 시간(Unix 타임스탬프, ms)	Int64	
RXSpeed_Bs	초당 수신 바이트 수: WaveOS에서 최근 수집 데이터를 기준으로 계산합니다.	Int	
TXSpeed_Bs	초당 송신 바이트 수: WaveOS에서 최근 수집 데이터를 기준으로 계산합니다.	Int	
RXProcessed_Ps	초당 처리한 수신 패킷 수: WaveOS에서 최근 수집 데이터를 기준으로 계산합니다.	Int	
TXProcessed_Ps	초당 처리한 송신 패킷 수: WaveOS에서 최근 수집 데이터를 기준으로 계산합니다.	Int	
RXDropped_Ps	초당 폐기한 수신 패킷 수: WaveOS에서 최근 수집 데이터를 기준으로 계산합니다.	Int	
TXDropped_Ps	초당 폐기한 송신 패킷 수: WaveOS에서 최근 수집 데이터를 기준으로 계산합니다.	Int	
RXError_Ps	초당 오류가 발생한 수신 패킷 수: WaveOS에서 최근 수집 데이터를 기준으로 계산합니다.	Int	
TXError_Ps	초당 오류가 발생한 송신 패킷 수: WaveOS에서 최근 수집 데이터를 기준으로 계산합니다.	Int	

5.3.2 무선 상태(RSSI)

무선 통계(RSSI)의 데이터 구조는 다음과 같습니다.

항목	설명	자료형/형식	허용 값	TLV 코드 *
WirelessInterfaceName	무선 인터페이스 이름	String	Radio0w0, radio1w0, etc.	0xDFC08016
DateTime_UTC	기록 날짜 및 시간(Unix 타임스탬프, ms)	Int64		0xDFC08055

항목	설명	자료형/형식	허용 값	TLV 코드 *
BSSid	기본 서비스 세트 식별자(BSSID)	String		0xDFC08019
ChannelWidth	무선 인터페이스의 채널 대역폭	Int/Enumeration	None = 0, HT20 = 1, HT40_Below = 2, HT40_Above = 3, HT40 = 4, VHT20 = 5, VHT40 = 6, VHT80 = 7, VHT80_80 = 8, VHT160 = 9, HE20 = 10, HE40 = 11, HE80 = 12, HE80_80 = 13, HE160 = 14	
PrimaryWideSegment	주 무선 세그먼트	Int/Enumeration	Channel_36 = 36, Channel_40 = 40, Channel_44 = 44, Channel_48 = 48, Channel_XX = XX, etc	
PrimaryChannel	무선 인터페이스의 주 채널	Int/Enumeration	Auto_Channel = 0, Channel_36 = 36, Channel_40 = 40, Channel_44 = 44, Channel_48 = 48, Channel_XX = XX, etc. 채널 값이 -1이면 제품이 채널을 검색 중입니다.	
SecondaryWideSegment	보조 무선 세그먼트	Int/Enumeration	Channel_36 = 36, Channel_40 = 40, Channel_44 = 44, Channel_48 = 48, Channel_XX = XX, etc .	
SecondaryChannel	사용 중인 보조 채널	Int		
ConnectionState	무선 연결 상태	Int/Enumeration	NotConnected = 0, Connected = 1	0xDFC0801F
AssociatedClientsNumber	AP에 연결된 클라이언트 수	Int		0xDFC0801D
Signal_dBm	신호 세기(dBm)	Int		0xDFC08020
SignalQuality	신호 품질	Int		0xDFC0800F
Noise_dBm	노이즈 수준(dBm)	Int		0xDFC08010
SNR	신호 대 잡음비(SNR)	Int		0xDFC08013
Throughput_KBps	예상 처리량(Kbps)	Int		0xDFC08012

* 참고: 텔레메트리 TLV 코드는 ACKSYS 내부용이며 MQTT API에서는 사용하지 않습니다.

5.3.3 로밍 상태

로밍 상태의 데이터 구조는 다음과 같습니다.

항목	설명	자료형/형식	허용 값	TLV 코드 *
WirelessInterfaceName	스캔에 사용하는 무선 인터페이스 이름	String	Radio0w0, radio1w0, etc.	0xDFC08016
DateTime_UTC	스캔 날짜 및 시간(UTC, Unix 타임스탬프, ms)	Int64		0xDFC08055
TopAccessPoints	상위 AP 목록(지정된 형식의 문자열로 저장)	[String]		
CompatibleAccessPointsNumber	호환 AP 수: 무선 인터페이스와 SSID가 같은 AP	Int		
IncompatibleAccessPointsNumber	비호환 AP 수: 무선 인터페이스와 SSID가 다른 AP	Int		

* 참고: 텔레메트리 TLV 코드는 ACKSYS 내부용이며 MQTT API에서는 사용하지 않습니다.

« TopAccessPoints » 필드에는 지정된 형식의 문자열 목록을 저장하며, 각 문자열은 하나의 AP를 나타냅니다. 각 AP 항목의 구조는 다음과 같습니다.

« BSSid|RoamingType|Channel|Signal_dBm|SignalQuality|Noise_dBm|SNR|Throughput_KBps »

각 필드는 |(파이프 기호)로 구분합니다.

상위 AP의 개수는 전송 활성화 요청에서 지정하며, 기본값은 20입니다. « AccessPoints » 문자열의 각 필드는 다음과 같습니다.

항목	설명	자료형/형식	허용 값	TLV 코드 *
BSSid	기본 서비스 세트 식별자(BSSID)	String		0xDFC08019
RoamingType	로밍 상태 유형	Int/Enumeration	None = 0, Data = 1, Best = 2, Passive = 3, UseAsBest = 4, Eliminated = 5, Minimum = 6, Maximum = 7, Blacklist = 8, Ignored = 9	0xDFC08014
Channel	사용 채널	Int	-1이면 제품이 채널을 검색 중입니다.	0xDFC0801B
Signal_dBm	신호 세기(dBm)	Int		0xDFC08020
SignalQuality	신호 품질	Int		0xDFC0800F
Noise_dBm	노이즈 수준(dBm)	Int		0xDFC08010
SNR	신호 대 잡음비(SNR)	Int		0xDFC08013
Throughput_KBps	예상 처리량(Kbps)	Int		0xDFC08012

* 참고: 텔레메트리 TLV 코드는 ACKSYS 내부용이며 MQTT API에서는 사용하지 않습니다.

5.3.4 연결 상태

연결된 클라이언트의 데이터 구조는 다음과 같습니다.

항목	설명	자료형/형식	허용 값	TLV 코드 *
WirelessInterfaceName	클라이언트가 연결된 무선 인터페이스(AP) 이름	String	Radio0w0, radio1w0, etc.	0xDFC08016
DateTime_UTC	기록 날짜 및 시간(Unix 타임스탬프, ms)	Int64		0xDFC08055
Ssid	서비스 세트 식별자(SSID)	String		0xDFC08015
BSsid	기본 서비스 세트 식별자(BSSID)	String		0xDFC08019
Channel	사용 채널	Int	-1이면 제품이 채널을 검색 중입니다.	0xDFC0801B
Signal_dBm	신호 세기(dBm)	Int		0xDFC08020
SignalQuality	신호 품질	Int		0xDFC0800F
Noise_dBm	노이즈 수준(dBm)	Int		0xDFC08010
SNR	신호 대 잡음비(SNR)	Int		0xDFC08013
ClientMACAddress	클라이언트 MAC 주소	String XX:XX:XX:XX:XX:XX	/	0xDFC08023

* 참고: 텔레메트리 TLV 코드는 ACKSYS 내부용이며 MQTT API에서는 사용하지 않습니다.

5.3.5 메시 링크 상태

메시 링크의 데이터 구조는 다음과 같습니다.

항목	설명	자료형/형식	허용 값	TLV 코드
WirelessInterfaceName	메시 무선 인터페이스 이름	String	Radio0w0, radio1w0, etc.	0xDFC08016
DateTime_UTC	기록 날짜 및 시간(Unix 타임스탬프, ms)	Int64		0xDFC08055
MeshId	메시 링크 식별자	String		0xDFC08039
Metric	메시 링크 비용(메트릭)	Int		0xDFC0802D
DiscoveryTimeout	메시 링크 검색 제한 시간	Int		0xDFC0802E
DiscoveryRetries	검색 재시도 횟수	Int		0xDFC0802F
State	메시 링크 상태	Int/Enumeration	None = 0, Active = 1, Resolving = 2, Resolved = 3, Fixed = 4, DNSValid = 5,	0xDFC08030 => 0xDFC08034
DestinationMACAddress	목적지 MAC 주소	String XX:XX:XX:XX:XX:XX	/	0xDFC08036
NextHopMACAddress	다음 홉 MAC 주소	String XX:XX:XX:XX:XX:XX	/	0xDFC08037

* 참고: 텔레메트리 TLV 코드는 ACKSYS 내부용이며 MQTT API에서는 사용하지 않습니다.

5.3.6 GNSS 상태

GNSS 상태의 데이터 구조는 다음과 같습니다.

항목	설명	자료형/형식	허용 값	TLV 코드
PhysicalInterfaceName	연결된 GNSS 인터페이스 이름	String	wwan0-gnss	0xDFC08022
DateTime_UTC	기록 날짜 및 시간(Unix 타임스탬프, ms)	Int64		0xDFC08055
GNSSState	GNSS 위치 정보 유효 여부	Int/Enumeration	NotValid = 0, Valid = 1	0xDFC0802C
Longitude	경도	Double		0xDFC08017
Latitude	위도	Double		0xDFC08018
Altitude	고도	Double		0xDFC08028
Speed_Kmh	속도(km/h)	Double		0xDFC08029
CourseDirection	북쪽 기준 진행 방향 각도	Double		0xDFC0802A

* 참고: 텔레메트리 TLV 코드는 ACKSYS 내부용이며 MQTT API에서는 사용하지 않습니다.

5.3.7 셀룰러 상태

셀룰러 상태의 데이터 구조는 다음과 같습니다.

항목	설명	자료형/형식	허용 값	TLV 코드
PhysicalInterfaceName	연결된 셀룰러 인터페이스 이름	String	wwan0	0xDFC08022
DateTime_UTC	기록 날짜 및 시간(Unix 타임스탬프, ms)	Int64		0xDFC08055
IMEI	셀룰러 인터페이스의 국제 이동통신 단말기 식별번호(IMEI)	String		
Model	셀룰러 인터페이스 모델	String		
CurrentSIM	사용 중인 SIM 카드	Int/Enumeration	None = 0, SIM1 = 1, SIM2 = 2	0xDFC0803A
Signal_dBm	신호 세기(dBm)	Int		0xDFC0803B
DataCount	데이터 수	Int64		0xDFC0803C
RegisterState	셀룰러 망 등록 상태	Int/Enumeration	NotRegistered= 0, Registered = 1, Searching = 2, Roaming = 3, Unknown = 4	0xDFC0803D
SIMSwitchMode	SIM 전환 모드	Int/Enumeration	Auto = 0, Manual = 1	0xDFC0803F
Operator	통신 사업자 이름	String		0xDFC08040
MACAddress	셀룰러 인터페이스의 MAC 주소	String XX:XX:XX:XX:XX:XX		

* 참고: 텔레메트리 TLV 코드는 ACKSYS 내부용이며 MQTT API에서는 사용하지 않습니다.

5.3.8 시스템 상태

시스템 상태의 데이터 구조는 다음과 같습니다.

항목	설명	자료형/형식	허용 값
HorlogeState	제품 시계 동기화 상태	Int/Enumeration	NotSynchronized = 0, Synchronized = 1
HorlogeSynchronizationMethod	시계 동기화 방식	Int/Enumeration	None = 0, NTP = 1, Cellular = 2
SystemUpTime_S	시스템 가동 시간(초)	Int64	

5.3.9 네트워크 상태

DHCP, DHCPv6, SLAAC, VRRP 등 동적 모드에서 사용하는 네트워크 인터페이스의 데이터 구조는 다음과 같습니다.

항목	설명	자료형/형식	허용 값
NetworkInterfaceName	네트워크 인터페이스 이름(식별자)	String	lan, net1, net2, etc.
IPv4Addresses	IPv4 주소 목록: v4.0.0.1에서는 반환된 첫 번째 IPv4 주소만 사용합니다.	[{'ipv4_address_1_data'}, {'ipv4_address_2_data'}]	
IPv4Gateway	IPv4 게이트웨이	String / N.N.N.N	
IPv4DNSAddresses	IPv4 DNS 서버 목록	[String / N.N.N.N]	
IPv6Addresses	IPv6 설정 목록	[{'ipv6_address_1_data'}, {'ipv6_address_2_data'}]	
MACAddress	네트워크 인터페이스의 MAC 주소	String / XX:XX:XX:XX:XX:XX	

IPv4 설정의 데이터는 다음과 같습니다(« ipv4_address_1_data », « ipv4_address_2_data » 등).

항목	설명	자료형/형식	허용 값
IPv4Address	IPv4 주소	String / N.N.N.N	
IPv4Mask	IPv4 서브넷 마스크	String / N.N.N.N	

IPv6 설정의 데이터는 다음과 같습니다(« ipv6_address_1_data », « ipv6_address_2_data » 등).

항목	설명	자료형/형식	허용 값
IPv6Address	IPv6 주소	String XXXX:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX	/
IPv6PrefixLength	IPv6 접두사 길이	Int	
Type	IPv6 주소 유형	Int/Enumeration	LLA = 1, ULA = 2, GUA = 3

5.3.10 네트워크 검색 상태

네트워크 검색 상태는 다음 JSON 형식으로 반환합니다.

'Neighbors': [

```
{'neighbor_1_data'},
{'neighbor_2_data'}
]
```

인접 장비의 데이터 구조는 다음과 같습니다(neighbor_1_data, neighbor_2_data).

항목	설명	자료형/형식	허용 값
Interface	인터페이스 이름	String	eth0
Protocol	네트워크 검색 프로토콜	String	LLDP
Neighbor	검색된 인접 장비의 이름	String	acksys
Chassis	검색된 새시 ID	String / XX:XX:XX:XX:XX:XX	
Port	검색된 포트 이름	String	eth1
IPAddresses	검색된 IPv4 및 IPv6 주소	[N.N.N.N,xxxx:xxxx:xxx:xxxx:xxxx:xxxx:xxxx:xxxx]	
VLAN	검색된 VLAN 목록	[Object]	

VLAN 항목의 데이터 구조는 다음과 같습니다(vlan_1_data, vlan_2_data 등).

항목	설명	자료형/형식	허용 값
VlanId	VLAN 식별자	Int	4
Pvid	포트 VLAN 여부	true/false	
Name	VLAN 이름	String	vlan4

5.3.11 Syslog 상태

Syslog 상태의 데이터 구조는 다음과 같습니다.

항목	설명	자료형/형식	허용 값	TLV 코드 *
DateTime.UTC	기록 날짜 및 시간(UTC, Unix 타임스탬프, ms)	Int64		
Message	지정된 형식의 Syslog 메시지	String		

* 참고: 텔레메트리 TLV 코드는 ACKSYS 내부용이며 MQTT API에서는 사용하지 않습니다.

각 Syslog 메시지의 구조는 다음과 같습니다.

« Facility|SeverityLevel|TextLog »

각 필드는 |(파이프 기호)로 구분합니다.

« Message » 문자열의 각 필드는 다음과 같습니다.

항목	설명	자료형/형식	허용 값	TLV 코드 *
Facility	Syslog 메시지를 생성한 서비스 유형	Int/Enumeration	Kernel = 0, User = 1, Email = 2, System = 3, Security = 4, Internal = 5, Printer = 6, Network = 7, UUCP = 8,	

항목	설명	자료형/형식	허용 값	TLV 코드 *
			Clock = 9, Authorization = 10, FTP = 11, NTP = 12, Log_Audit = 13, Log_Alert = 14, Clock_Daemon = 15, Configuration = 16, Local_Use1 = 17, Local_Use2 = 18, Local_Use3 = 19, Local_Use4 = 20, Local_Use5 = 21, Local_Use6 = 22, Local_Use7 = 23	
SeverityLevel	Syslog 메시지의 심각도	Int/Enumeration	Emergency =0, Alert =1, Critical =2, Error =3, Warning =4, Notice =5, Info =6, Debug = 7	
TextLog	Syslog 메시지 내용	String		

* 참고: 텔레메트리 TLV 코드는 ACKSYS 내부용이며 MQTT API에서는 사용하지 않습니다.

5.3.12 압축·인코딩된 통계 전송

최적의 네트워크 성능을 유지하려면 통계 메시지의 크기가 4 KB를 넘지 않아야 합니다. 이 크기를 초과하면 정해진 알고리즘(예: GZIP)으로 압축한 뒤 Base64로 인코딩하여 정해진 JSON 형식으로 전송해야 합니다.

압축 후 Base64로 인코딩한 데이터는 다음 JSON 구조로 전송합니다.

```
{
  "CompressionAlgorithm" : 1,
  "CompressedEncodedData" : "compressed_encoded_statistics"
}
```

대역폭 통계의 데이터 구조는 다음과 같습니다.

항목	설명	자료형/형식	허용 값
CompressionAlgorithm	압축 알고리즘	Int/Enumeration	BZIP2= 1, GZIP= 2
Data	압축 후 Base64로 인코딩한 데이터	string	

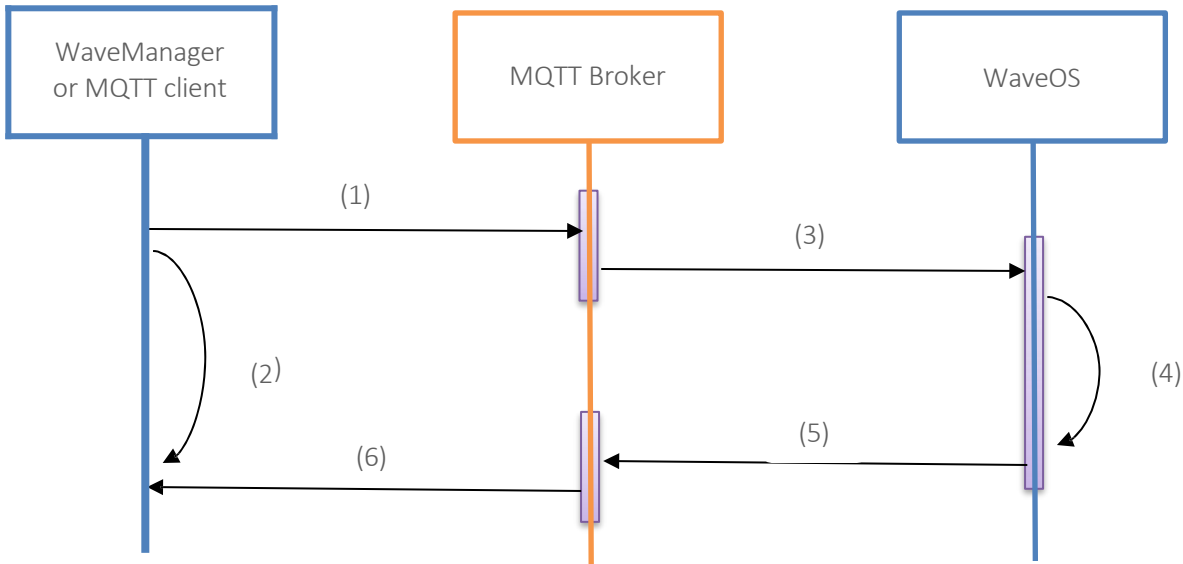
5.3.13 MQTT 클라이언트 연결 해제 처리

MQTT 클라이언트와 라우터의 연결이 끊긴 후 다시 연결되면, 최초로 수집한 텔레메트리 통계만 전송합니다.

5.4 명령 전송

명령(설정 변경 또는 동작)의 전송 및 응답 확인 절차는 다음과 같습니다. 아래는

WaveManager 서비스가 MQTT 클라이언트로 동작하는 예입니다.



1. 설정 변경 또는 동작 명령을 JSON 형식으로 MQTT 브로커에 전송합니다.
2. WaveManager 서비스는 명령별 제한 시간까지 장비의 응답을 기다립니다. 제한 시간이 지나면 해당 명령을 실패로 처리합니다.
3. MQTT 브로커가 명령 데이터를 장비에 전달합니다.
4. 장비가 지정된 명령을 실행합니다.
5. 장비가 명령 적용 상태를 MQTT 브로커에 전송합니다.
6. MQTT 브로커가 명령의 상태와 결과를 WaveManager 서비스에 전달합니다.
7. 제한 시간이 지난 후 장비의 응답이 도착하면, 시간 초과와 새 명령 상태를 모두 반영하여 명령 상태를 갱신합니다.

지원하는 명령 유형은 두 가지입니다.

1. **설정 변경(Update):** 장비 설정을 변경하는 명령
2. **동작(Action):** 장비 설정에 영향을 주지 않으며, 주로 네트워크 상태를 테스트하는 명령

5.4.1 설정 변경

설정 변경 명령은 다음 JSON 형식으로 전송합니다.

```

{
  "Id": int_update_multiple_command_id, "UpdateCommands": [
    { "update_command_data",
      .....
    ]
}

```

여러 설정 변경 명령을 포함하는 데이터 구조는 다음과 같습니다.

항목	설명	자료형	허용 값
Id	일괄 설정 변경 명령 ID	Int	
UpdateCommands	설정 변경 명령 목록	[{'update_commande_1_data'}, {'update_commande_2_data'}]	

각 설정 변경 명령 « update_command_data »에는 다음 항목이 포함됩니다.

항목	설명	자료형	허용 값
Id	개별 설정 변경 명령 ID	Int	
Type	설정 변경 유형	Int/Enumeration	FirmwareUpdate = 1, ConfigurationFileUpdate = 2, SystemUpdate = 3, PhysicalInterfaceUpdate = 4, WirelessUpdate = 5, NetworkUpdate = 6, WebServerUpdate = 7, StatusSettingUpdate = 8, HighVolumeDataCollectionUpdate=9, ResetToFactoryUpdate=10
Data	설정 변경 데이터	설정 변경 유형별 JSON 객체	

제품 설정은 전체(향후 지원 예정) 또는 일부만 변경할 수 있습니다. 각 설정 메시지의 필수 필드는 "Mandatory", 선택 필드는 "Optional" 속성으로 표시합니다.

설정 메시지에는 최소 하나의 설정 필드가 포함되어야 합니다. 일부 설정만 변경할 때 생략할 수 있는 항목은 선택 필드뿐입니다.

선택 필드를 지정하지 않으면 제품은 해당 필드의 기존 값을 유지합니다.

5.4.1.1 펌웨어 업데이트

펌웨어 업데이트 명령의 데이터는 다음과 같습니다.

항목	설명	자료형	필수 여부	허용 값
URL	펌웨어 파일 URL	String	필수	

5.4.1.2 설정 파일 업데이트

설정 파일 업데이트의 데이터는 다음과 같습니다.

항목	설명	자료형	필수 여부	허용 값
URL	설정 파일 URL	String	필수	

5.4.1.3 시스템 설정 변경

시스템 설정 변경은 다음 데이터 구조로 전송합니다.

항목	설명	자료형	필수 여부	허용 값
Hostname	장비 이름	String	선택	
Description	장비 설명(설치 위치)	String	선택	
NTPServerNames	NTP 서버 목록	[String]	선택	

항목	설명	자료형	필수 여부	허용 값
NTPServerPort	NTP 서버 포트	Int	선택	

변경할 시스템 항목만 지정하여 시스템 설정의 전체 또는 일부를 변경할 수 있습니다.

5.4.1.4 물리 무선 인터페이스 설정 변경

하나 이상의 물리 무선 인터페이스 설정을 다음 JSON 형식으로 전송하여 채널/역할을 변경합니다.

```
[
    {'wireless_element_update_data'},
    .....
]
```

물리 무선 인터페이스 설정 목록을 전송합니다. 각 인터페이스의 설정 데이터는 다음과 같습니다. « wirelessss_element_update_data »:

항목	설명	자료형	필수 여부	허용 값
Name	물리 인터페이스 이름(식별자)	String	필수	radio0, radio1, etc.
State	물리 인터페이스 상태	Int/Enumeration	선택	Disabled = 0, Enabled = 1
WifiMode	무선 인터페이스의 802.11 모드	Int/Enumeration	선택	None = 0, B_Only = 1, G_Only = 2, Mixed_B_G = 3, A_Only = 4, N_G = 5, N_BG = 6, N_A = 7, AC = 8, AX = 9
ChannelWidth	무선 인터페이스의 채널 대역폭	Int/Enumeration	선택	None = 0, HT20 = 1, HT40_Below = 2, HT40_Above = 3, HT40 = 4, VHT20 = 5, VHT40 = 6, VHT80 = 7, VHT80_80 = 8, VHT160 = 9, HE20 = 10, HE40 = 11, HE80 = 12, HE80_80 = 13, HE160 = 14
PrimaryWideSegment	주 무선 세그먼트	Int/Enumeration	선택	Channel_36 = 36, Channel_40 = 40, Channel_44 = 44, Channel_48 = 48, Channel_XX = XX, etc.
PrimaryChannel	무선 인터페이스의 주 채널	Int/Enumeration	선택	Auto_Channel = 0, Channel_36 = 36, Channel_40 = 40, Channel_44 = 44,

항목	설명	자료형	필수 여부	허용 값
				Channel_48 = 48, Channel_XX = XX, etc.
SecondaryWideSegment	보조 무선 세그먼트	Int/Enumeration	선택	Channel_36 = 36, Channel_40 = 40, Channel_44 = 44, Channel_48 = 48, Channel_XX = XX, etc.
MaximalTransmitPower_dBm	무선 인터페이스의 최대 송신 출력	Int	선택	

대역, Wi-Fi 모드, 채널 대역폭, 주 채널, 보조 채널, 최대 송신 출력 중 변경할 항목만 지정하여 설정의 전체 또는 일부를 변경할 수 있습니다.

5.4.1.5 논리 무선 인터페이스 설정 변경

하나 이상의 무선 인터페이스 설정을 다음 JSON 형식으로 전송하여 SSID/보안 모드를 변경합니다.

```
[
    {'wireless_interface_update_data'},
    .....
]
```

무선 인터페이스 설정 목록을 전송합니다. 각 인터페이스의 설정 데이터는 다음과 같습니다.

« wireless_interface_update_data »:

항목	설명	자료형	필수 여부	허용 값
Name	무선 인터페이스 이름: WaveOS가 이름을 동적으로 생성하는 문제를 피하기 위해 radio 섹션 이름을 사용합니다.	String	필수	Radio0w0, radio1w0, etc.
Role	무선 역할: "Monitor" 역할은 이 버전에서 비활성화됩니다.	Int/Enumeration	선택	Undefined = 0, Client = 1, AccessPoint = 2, Adhoc = 3, Monitor = 5, Mesh = 6, Repeater = 7 IsolatingAccessPoint = 8, RogueAP = 9, SRCC = 10
Ssid	서비스 세트 식별자(SSID)	String	선택	
SecurityMode	보안 모드	Int/Enumeration	선택	NONE = 0, WPA-PSK = 1, WPA2-PSK = 2, WPA3-PSK = 3, MIXED_WPA_WPA2_PSK = 4, MIXED_WPA2_WPA3_PSK = 5, WPA_EAP = 6, WPA2_EAP = 7, WPA3_EAP = 8, MIXED_WPA2_WPA3_EAP = 9, WPA3_OWE = 10, OSEN = 11, WEP_OPEN_SYSTEM = 12, WEP_SHARED_KEY = 13

항목	설명	자료형	필수 여부	허용 값
SecurityKey	보안 키	String	선택	

역할, SSID, 보안 중 변경할 항목만 지정하여 무선 인터페이스 설정의 전체 또는 일부를 변경할 수 있습니다.

5.4.1.6 네트워크 설정 변경

하나 이상의 네트워크 인터페이스 설정을 다음 JSON 형식으로 전송하여 네트워크 설정을 변경합니다.

```
[
    {'network_interface_update_data'},
    .....
]
```

네트워크 인터페이스 설정 목록을 전송합니다. 각 인터페이스의 설정 데이터는 다음과 같습니다.

« network_interface_update_data » :

항목	설명	자료형	필수 여부	허용 값
Name	네트워크 인터페이스 이름(네트워크 식별자)	String	필수	
Label	네트워크 레이블	String	선택	
State	네트워크 상태	Int/Enumeration	선택	Disabled = 0, Enabled = 1
IPv4Mode	IPv4 모드	Int/Enumeration	선택	None = 0, Static = 1, Dhcp = 2, VRRP = 3
IPv4Addresses	IPv4 주소 목록: 첫 번째 릴리스에서는 마지막 주소만 설정에 반영합니다.	[{ 'ipv4_address_1_data' }, { 'ipv4_address_2_data' }]	선택	
IPv4Gateway	IPv4 게이트웨이	String / N.N.N.N	선택	
IPv4DNSAddresses	IPv4 DNS 서버 목록	[String/ N.N.N.N]	선택	
IPv6Mode	IPv6 모드	Int/Enumeration	선택	None = 0, Static = 1, SLAAC = 2, Dhcpv6 = 3
IPv6Addresses	IPv6 설정 목록	[{ 'ipv6_address_1_data' }, { 'ipv6_address_2_data' }]	선택	
IPv6Gateway	IPv6 게이트웨이	String XXXX:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX	선택	
AssociatedInterfaceNames	현재 이 네트워크 인터페이스에	[String]	선택	

항목	설명	자료형	필수 여부	허용 값
	연결된 이더넷 물리 인터페이스 및 무선 인터페이스 목록			

IPv4, IPv6, 연결된 물리 인터페이스 중 변경할 항목만 지정하여 네트워크 설정의 전체(향후 지원 예정) 또는 일부를 변경할 수 있습니다.

선택한 설정에 따른 IPv4 및 IPv6 모드는 다음과 같습니다.

선택 모드	IPv4Mode	IPv6Mode
none	none	none
static	static	static
Dhcp	Dhcp	none
VRRP	VRRP	none
SLAAC	none	SLAAC
Dhcpv6	none	Dhcpv6

IPv4 설정의 데이터는 다음과 같습니다(« ipv4_address_1_data », « ipv4_address_2_data » 등).

항목	설명	자료형	허용 값
IPv4Address	IPv4 주소	String / N.N.N.N	
IPv4Mask	IPv4 서브넷 마스크	String / N.N.N.N	

IPv6 설정의 데이터는 다음과 같습니다(« ipv6_address_1_data », « ipv6_address_2_data » 등).

항목	설명	자료형	허용 값
IPv6Address	IPv6 주소 값	String / XXXX:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX:XXXX	
IPv6PrefixLength	IPv6 접두사 길이	Int	

5.4.1.7 웹 서버 설정 변경

웹 서버 설정 변경 명령의 데이터는 다음과 같습니다.

항목	설명	자료형	필수 여부	허용 값
WebServerMode	웹 서버 접속 모드	Int/Enumeration	선택	None = 0, HTTP_Only = 1, HTTPS_Only = 2, Mixed_HTTP_HTTPS = 3
HTTPPort	HTTP 포트	Int	선택	0~65625
HTTPSPort	HTTPS 포트	Int	선택	0~65625
HTTPCertificateURL	HTTPS 인증서 파일 URL	String	선택	

5.4.1.8 상태 수집 설정 변경

텔레메트리 설정 변경 명령의 데이터는 다음과 같습니다.

항목	설명	자료형	필수 여부	허용 값
SamplingInterval_S	수집 주기(초)	Int	선택	
OutputInterval_S	전송 주기(초)	Int	선택	

5.4.1.9 대용량 데이터 수집 설정 변경

로밍 데이터 수집을 활성화/비활성화하는 데이터는 다음과 같습니다.

항목	설명	자료형	필수 여부	허용 값
DataType	대용량 데이터 유형	Int/Enumeration	필수	Roaming = 1, Syslog = 2
State	수집 상태	Int/Enumeration	필수	0~65625
Options	수집할 데이터를 선택하는 고급 설정	동작 유형별 JSON 객체	선택	0~65625

5.4.1.9.1 로밍 데이터 수집 활성화

로밍 데이터 수집 활성화 옵션은 다음과 같습니다.

항목	설명	자료형	허용 값
MaximumTopAccessPoints	반환할 상위 AP의 최대 개수	Int	

5.4.1.9.2 동작: Syslog 데이터 수집 활성화

Syslog 데이터 수집 활성화 옵션은 다음과 같습니다.

항목	설명	자료형	허용 값
MinimumSeverityLevel	최소 Syslog 심각도	Int/Enumeration	Emergency =0, Alert =1, Critical =2, Error =3, Warning =4, Notice =5, Info =6, Debug = 7

10. 공장 초기화

해당 없음

11. 설정 변경 명령에 대한 장비 응답

설정 변경 명령의 응답은 다음 JSON 형식으로 전송합니다.

```
{
  "ProductId": "product_id",
  "Id": int_update_multiple_command_id, "UpdateResults": [
    { "update_result_data" },
    ...
  ]
}
```

일괄 설정 변경 명령에 대한 응답 데이터 구조는 다음과 같습니다.

항목	설명	자료형	허용 값
ProductId	장비 고유 식별자	String	
Id	일괄 설정 변경 명령 식별자	Int	
UpdateResults	각 설정 변경 명령의 결과 목록	[{'update_commande_1_result_data'}, {'update_commande_2_result_data'}]	

개별 설정 변경 응답의 데이터 구조는 다음과 같습니다. « update_commande_1_result_data »

항목	설명	자료형	허용 값
Id	개별 설정 변경 식별자	Int	
State	설정 변경 상태	Int/Enumeration	Failed = 0, Success = 1
ErrorMessage	설정 변경 적용 실패 시 오류 메시지	String	

5.4.2 동작

하나 이상의 동작 명령은 다음 JSON 형식으로 전송합니다.

```
{
  "Id" : int_action_multiple_command_id, "ActionCommands": [
    {'action_command_data'},
    .....
  ]
}
```

여러 동작 명령을 포함하는 데이터 구조는 다음과 같습니다.

항목	설명	자료형	허용 값
Id	동작 명령 ID	Int	
ActionCommands	동작 명령 목록	[{'action_command_1_data'}, {'action_command_2_data'}]	

개별 동작 명령의 데이터는 다음과 같습니다. « action_command_data »

항목	설명	자료형	허용 값
Id	개별 동작 ID	Int	
Type	동작 유형	Int/Enumeration	ConfigurationFileDownloadAction = 1, LedTrackingAction = 2, WifiScanAction = 3, PingAction = 4, TracerouteAction = 5, BandwidthTestAction = 6, DNSTestAction = 7, Reboot = 8, CloudServiceRestart = 9, HighVolumeDataCollectionAction = 10
Data	동작 데이터	동작 유형별 JSON 객체	

5.4.2.1 동작: 설정 파일 다운로드

설정 파일 다운로드 동작의 데이터는 다음과 같습니다.

항목	설명	자료형	허용 값
URL	설정 파일을 저장할 URL	String	

5.4.2.2 동작: 장비 식별 LED 활성화/비활성화

장비 식별 LED 활성화/비활성화 동작의 데이터는 다음과 같습니다.

항목	설명	자료형	허용 값
State	LED 상태	Int/Enumeration	Disabled = 0, Enabled = 1
Delay_M	활성화 지연 시간(분)	Int	

5.4.2.3 동작: Wi-Fi 스캔

Wi-Fi 스캔 동작의 데이터는 다음과 같습니다.

항목	설명	자료형	허용 값
WirelessInterfaceName	Wi-Fi 스캔에 사용하는 물리 무선 인터페이스 이름	String	radio0, radio1, etc.

5.4.2.4 동작: Ping

Ping 동작의 데이터는 다음과 같습니다.

항목	설명	자료형	허용 값
Host	원격 호스트	String	

5.4.2.5 동작: Traceroute

Traceroute 동작의 데이터는 다음과 같습니다.

항목	설명	자료형	허용 값
Host	원격 호스트	String	

5.4.2.6 동작: 대역폭 테스트

대역폭 테스트 동작의 데이터는 다음과 같습니다.

항목	설명	자료형	허용 값
Mode	테스트 모드	Int/Enumeration	Server = 1, Client = 2
Protocol	테스트 프로토콜	Int/Enumeration	TCP = 1, UDP = 2
ServerIPAddress	서버 IP 주소	String	
Rate_MBS	대역폭(Mbps)	Int	

항목	설명	자료형	허용 값
Delay_S	테스트 시간(초)	Int	
Display_S	표시 주기(초)	Int	

5.4.2.7 동작: DNS 테스트

DNS 테스트 동작의 데이터는 다음과 같습니다.

항목	설명	자료형	허용 값
Host	원격 호스트	String	
Code	DNS 레코드 유형	Int/Enumeration	A = 1, AAAA = 28 (IANA codes)

8. 동작: 재부팅

해당 없음

9. 동작: 클라우드 서비스 재시작

해당 없음

10. 동작 요청에 대한 장비 응답

동작 응답은 다음 JSON 형식으로 전송합니다.

```
{
  "ProductId" : "product_id",
  "Id" : int_action_multiple_command_id, "ActionResults": [
    {'action_result_data'},
    .....
  ]
}
```

일괄 동작 명령에 대한 응답 데이터는 다음과 같습니다.

항목	설명	자료형	허용 값
ProductId	장비 고유 식별자	String	
Id	일괄 동작 명령 ID	Int	
ActionResults	각 동작 명령의 결과 목록	[{'action_result_1_data'}, {'action_result_2_data'}]	

각 동작 결과 « action_result_data »에는 다음 항목이 포함됩니다.

항목	설명	자료형	허용 값
Id	명령 ID	Int	
State	명령 상태	Int/Enumeration	Failed = 0, Success = 1
Data	동작 결과(성공 시)	동작 유형별 JSON 객체	

항목	설명	자료형	허용 값
ErrorMessage	오류 메시지(실패 시)	String	

1. 설정 파일 다운로드 동작의 응답 데이터

N.A

2. 장비 식별 LED 활성화/비활성화 동작의 응답 데이터

N.A

5.4.2.10.3 Wi-Fi 스캔 동작의 응답 데이터

Wi-Fi 스캔을 완료하면 다음 JSON 형식으로 스캔 데이터를 반환합니다.

```
'Data': [
    {'wifi_scan_data'},
    .....
]
```

Wi-Fi 스캔 결과에 포함되는 각 무선 항목의 데이터는 다음과 같습니다(« wifi_scan_data » 등).

항목	설명	자료형	허용 값
WirelessInterfaceName	Wi-Fi 스캔에 사용하는 물리 무선 인터페이스 이름	String	radio0, radio1, etc.
Role	무선 역할 "Monitor" 는 해당 버전에서 비활성화됩니다.	Int/Enumeration	Undefined = 0, Client = 1, AccessPoint = 2, Adhoc = 3, Monitor = 5, Mesh = 6, Repeater = 7 IsolatingAccessPoint = 8, RogueAP = 9, SRCC = 10
Ssid	서비스 세트 식별자(SSID)	String	
BSsid	기본 서비스 세트 식별자(BSSID)	String	
SecurityMode	보안 모드	Int/Enumeration	NONE = 0, WPA-PSK = 1, WPA2-PSK = 2, WPA3-PSK = 3, MIXED_WPA_WPA2_PSK = 4, MIXED_WPA2_WPA3_PSK = 5, WPA_EAP = 6, WPA2_EAP = 7, WPA3_EAP = 8, MIXED_WPA2_WPA3_EAP = 9, WPA3_OWE = 10, OSEN = 11, WEP_OPEN_SYSTEM = 12, WEP_SHARED_KEY = 13
Channel	사용 채널	Int	

항목	설명	자료형	허용 값
Frequency_GHz	채널 주파수	Double	
ChannelWidth	채널 대역폭	Int/Enumeration	20, 40, 80, 160
Signal_dBm	신호 세기(dBm)	Int	

5.4.2.10.4 Ping 동작의 응답 데이터

Ping 테스트를 완료하면 다음 JSON 형식으로 결과를 반환합니다.

'Data' : 'string_ping_result'

결과는 문자열로 반환합니다.

5. Traceroute 동작의 응답 데이터

Traceroute를 완료하면 다음 JSON 형식으로 결과를 반환합니다.

'Data' : 'string_traceroute_result'

결과는 문자열로 반환합니다.

6. 대역폭 테스트 동작의 응답 데이터

대역폭 테스트를 완료하면 다음 JSON 형식으로 결과를 반환합니다.

'Data' : 'string_bandwidth_result'

결과는 문자열로 반환합니다.

7. DNS 테스트 동작의 응답 데이터

DNS 테스트를 완료하면 다음 JSON 형식으로 결과를 반환합니다.

'Data' : 'string_dns_result'

결과는 문자열로 반환합니다.

8. 재부팅 동작의 응답 데이터

해당 없음

9. 클라우드 서비스 재시작 동작의 응답 데이터

해당 없음

5.5 MQTT 토픽

장비 설정 전송을 관리하는 MQTT 토픽은 다음과 같습니다.

토픽	설명
configuration/state/<wm_service_id>/<router_id>/request	WaveManager 서비스/MQTT 클라이언트가 장비의 설정 상태를 요청합니다.
configuration/state/<wm_service_id>/<router_id>	장비가 WaveManager 서비스/MQTT 클라이언트에 설정 상태를 전송합니다.
configuration/data/<wm_service_id>/<router_id>/request	WaveManager 서비스/MQTT 클라이언트가 하나 이상의 설정 구성 요소 데이터를 요청합니다.
configuration/data/<wm_service_id>/<router_id>	장비가 WaveManager/MQTT 클라이언트에 하나 이상의 설정 구성 요소 데이터를 전송합니다.
status/<wm_service_id>/<router_id>	장비가 WaveManager 서비스/MQTT 클라이언트에 기록된 통계를 전송합니다.
command/update/<wm_service_id>/<router_id>/request	WaveManager 서비스/MQTT 클라이언트가 장비 설정 변경을 요청합니다.
command/update/<wm_service_id>/<router_id>	장비가 WaveManager 서비스/MQTT 클라이언트에 설정 변경 상태를 전송합니다.
command/action/<wm_service_id>/<router_id>/request	WaveManager 서비스/MQTT 클라이언트가 장비에 동작을 요청합니다.
command/action/<wm_service_id>/<router_id>	장비가 WaveManager 서비스/MQTT 클라이언트에 동작 상태를 전송합니다.

각 식별자의 의미:

- **wm_service_id:** WaveManager 서비스(또는 MQTT 클라이언트)의 식별자입니다.
- **router_id:** 장비 식별자로, 장비의 메인보드 ID를 지정된 형식으로 변환한 값입니다.

MQTT 토픽에서 사용하는 장비 식별자의 예:

메인보드 ID	MQTT 토픽용 라우터 ID
00:00:11:6E:EA:17	0000116EEA17

장비가 전송하는 모든 메시지는 장비 식별자 « ProductId »가 포함됩니다.