

# **Eevp.Control.SoaRoa**

EEVP\_VC

Exported on 07/15/2025

## Table of Contents

1 History .....	3
2 Eevp.Control.SoaRoa 서비스.....	4
3 Interface Download.....	5
4 Main.Control.SoaRoa Interface.....	6

# 1 History

Version	담당자 (업체)	주요 변경 사항
r1	모비스	초안 release
r2	모비스	SW 버전 Field 추가
r3	모비스	사용 인자 푸른색 음영처리

## 2 Eevp.Control.SoaRoa 서비스

항목	값
서비스 이름	Eevp.Control.SoaRoa
Version	1.00
설명	ROA 구동을 위한 서비스

### 3 Interface Download

마스터DB	버전	다운로드	BitBucket

## 4 Main.Control.SoaRoa Interface

- Instance
  - Single Instance
- Description
  - Interface designed to use

Service Interface	Element Type	Element Name	Args	Description	SOME/IP IDs	Permission	Change Log (Jira)	비고
Eevp.Control.SoaRoa	Method	StartRoa	No Arguments	ROA detect mode를 시작하는 함수	Method ID = 16385 (0x4001)	UDP		추후 Front Door, Rear Door 관련 시나리오가 추가될 경우에는 StartRoa, StopRoa가 필요없어질 수 있다. (Front Door와 Rear Door의 개폐 상황이 ROA mode 시작의 trigger가 된다)

Service Interface	Element Type	Element Name	Args	Description	SOME/IP IDs	Permission	Change Log (Jira)	비고
Eevp.Control.SoaRoa	Method	StopRoa	No Arguments	ROA detect mode를 종료하는 함수	Method ID = 16386 (0x4002)	UDP		추후 Front Door, Rear Door 관련 시나리오가 추가될 경우에는 StartRoa, StopRoa가 필요없어질 수 있다. (Front Door와 Rear Door의 개폐 상황이 ROA mode 시작의 trigger가 된다)



Service Interface	Element Type	Element Name	Args	Description	SOME/IP IDs	Permission	Change Log (Jira)	비고
Eevp.Control.SoaRoa	Method	ResetRoaDetectInfo	No Arguments	<p>ROA detect mode의 감지 결과를 reset하는 함수.</p> <p>ROA mode를 시작했을 때 (혹은 시작되었을 때) stop 안 시키고 detect 여부에 대한 각종 정보만을 reset하고 싶을 때 사용</p>	Method ID = 16387 (0x4003)	UDP		

Service Interface	Element Type	Element Name	Args	Description	SOME/IP IDs	Permission	Change Log (Jira)	비고
Eevp.Control.SoaRoa	Method	SetRoaMode	<pre> enumeration SoaRoaMode {     kOFF = 0x00     kNORMAL = 0x01     kADVANCED = 0x02 } </pre>	<p>ROA 동작 모드를 설정하는 모드.</p> <p>상용 ROA라면 위의 Start/Stop 이 아니라 Front Door/Rear Door의 개폐에 따라 동작이 시작되고, 그 Sensor 동작 여부는 Mode에 따라 달라진다.</p> <ul style="list-style-type: none"> <li>• kOFF : ROA 관련 동작을 전혀 안 한다</li> <li>• kNORMAL: Driver Door가 열렸다 닫혔을 때에만 Inform하는 가장 기본적인 기능</li> <li>• kADVANCED: Driver Door가 열렸다 닫힌 후 ROA 기능을 진행하는 모드</li> </ul>	Method ID = 16388 (0x4004)	UDP		<p>현재 시나리오에서는 사용되지 않을 예정이고 추후 Front Door, Rear Door 관련 시나리오가 추가될 경우에는 StartRoa, StopRoa가 필요없어지고 대신 이 함수가 동작 시나리오를 정해주는 것만으로 끝난다.</p> <p>(Front Door와 Rear Door의 개폐 상황이 ROA mode 시작의 trigger가 된다)</p>

Service Interface	Element Type	Element Name	Args	Description	SOME/IP IDs	Permission	Change Log (Jira)	비고
Eevp.Control.SoaRoa	Field	soaRoaDeviceNormal	<pre>enumeration SoaRoaDetectState {     kNORMAL = 0x00     kABNORMAL = 0x01 }</pre> <div> <pre>enumeration SoaDeviceIsNormal {     kNORMAL = 0x00     kABNORMAL = 0x01 }</pre> </div>	ROA Device가 정상 동작하는지 알려주는 Field  kNORMAL : 부품 정상상태 kABNORMAL = 부품 비정상상태	Getter ID = 1 (0x0001) Notifier ID = 49153 (0xC001) EventGroup ID = 8705 (0x2201)	UDP read only		
Eevp.Control.SoaRoa	Field	soaRoaRunningState	<div> <pre>enumeration SoaRoaRunningState {     kOFF = 0x00     kON = 0x01     kUNKNOWN = 0x02 }</pre> </div>	ROA 기능의 현재 동작 상태를 알려주는 Field  kUNKNOWN: 고장 등의 이유로 동작하지 않을때	Getter ID = 2 (0x0002) Notifier ID = 49154 (0xC002) EventGroup ID = 8706 (0x2202)	UDP read only		

Service Interface	Element Type	Element Name	Args	Description	SOME/IP IDs	Permission	Change Log (Jira)	비고
Eevp.Control.SoaRoa	Field	soaRoa Mode	<pre> enumeration SoaRoaMode {     kOFF = 0x00     kNORMAL = 0x01     kADVANCED = 0x02 } </pre>	<p>ROA mode를 알려주는 Field</p> <ul style="list-style-type: none"> <li>• kOFF : ROA 관련 동작을 전혀 안 한다</li> <li>• kNORMAL: Driver Door가 열렸다 닫혔을 때에만 Inform하는 가장 기본적인 기능</li> <li>• kADVANCE D: Driver Door가 열렸다 닫힌 후 ROA 기능을 진행하는 모드</li> </ul>	<p>Getter ID = 3 (0x0003)</p> <p>Notifier ID = 49155 (0xC003)</p> <p>EventGroup ID = 8707 (0x2203)</p>	UDP read only		현재 Start / Stop 으로 동작이 시작되는 시나리오에서는 항상 kOFF 만 존재하고, 나중에 Start / Stop 명령 외에 Front Door / Rear Door로 동작이 되어야 하는 시나리오가 추가 되면 그 때에 설정한 값에 따라 다른 값이 가게 될 것이다.

Service Interface	Element Type	Element Name	Args	Description	SOME/IP IDs	Permission	Change Log (Jira)	비고
Eevp.Control.SoaRoa	Field	soaRoaDetectState	<pre> <b>enumeration</b> <b>SoaRoaDetectState</b> {     <b>kEMPTY =</b> <b>0x00</b>      <b>kDETECTED_ONCE</b> <b>= 0x01</b>      kDETECTED_SEVERAL = 0x02     kUNKNOWN = 0x03 } </pre> <div> <pre> enumeration SoaRoaDetectState {     kEMPTY = 0x00      kDETECTED_ONCE = 0x01      kDETECTED_SEVERAL = 0x02     kUNKNOWN = 0x03 } </pre> </div>	<p>ROA Start/Reset 이후 현재 Detect State를 가리키는 Field</p> <ul style="list-style-type: none"> <li><b>kEMPTY:</b> (움직임 감지 x) + <b>Unknown</b> 상태</li> <li><b>kDETECTED_ONCE:</b> 움직임 감지</li> <li><b>kDETECTED_SEVERAL:</b> 여러번 감지</li> <li><b>kUNKNOWN:</b> 센서 고장 혹은 그 외의 이유로 알 수 없다</li> </ul>	<p>Getter ID = 4 (0x0004)</p> <p>Notifier ID = 49156 (0xC004)</p> <p>EventGroup ID = 8708 (0x2204)</p>	UDP read only		

Service Interface	Element Type	Element Name	Args	Description	SOME/IP IDs	Permission	Change Log (Jira)	비고
Eevp.Control.SoaRoa	Field	soaRoa DetectCount	UInt8	ROA Start/Reset 이후 현재까지 Detect된 회수를 가리키는 Field	Getter ID = 5 (0x0005) Notifier ID = 49157 (0xC005) EventGroup ID = 8709 (0x2205)	UDP read only		움직임이 감지되거나 reset/stop 이 될 때 이 값이 확실히 변하므로 움직임이 감지되는 Notify를 확인하고 싶으면 이 Field의 Notify에 걸면 된다. (값이 0일 경우: stop, reset 외엔 다 움직임이 감지되어 증가된 경우)

Service Interface	Element Type	Element Name	Args	Description	SOME/IP IDs	Permission	Change Log (Jira)	비고
Eevp.Control.SoaRoa	Field	soaRoaSensorError	<pre> enumeration SoaRoaSensorError {     kOK = 0x00      kERR_SNSR = 0x01      kERR_BLCKGE = 0x02 } </pre>	Sensor가 잘못되었는지를 좀 더 자세하게 알려주는 Field	Getter ID = 6 (0x0006) Notifier ID = 49158 (0xC006) EventGroup ID = 8710 (0x2206)	UD P read only		
Eevp.Control.SoaRoa	Field	soaRoaSwVersion	UInt8 soaRoaSwVersion	SOA ROA module의 현재 Software Version	Getter ID = 7 (0x0007) Setter ID = None (Read Only) Notifier ID = 49159 (0xC007) EventGroup ID = 8711 (0x2207)	UD P read only		