CSL Intelligent Fixed Reader HTTP API V1.3

1. HTTP Protocol Support

1.1. Introduction

The following is a table of the HTTP query strings and the XML based response document layout. It consists of 9 main categories:

- 1. Users Management
- 2. System Management
- 3. Network Management
- 4. Time and Timer Management
- 5. Version Management
- 6. Capture Point Management
- 7. Tag and Tag Filter Management
- 8. IO Management
- 9. Events Management

This table of API is for Web Application 1.4.6.

1.2. HTTP API Format

(1) Server \rightarrow Reader

The format of High-level HTTP API query from server to reader is as follows:

http://<IP_address_of_Reader>/API?session_id=<session_id>&comma
nd=<command>[&<param1>=<param1_value>]

where:

Variable	Description
<ip_address_of_reader></ip_address_of_reader>	IP address of the CSL intelligent fixed reader
<session_id></session_id>	The session ID obtained in the XML response message from reader after user login (not necessary for some commands, e.g. login)
<command/>	High-level API command
<pre><param1></param1></pre>	Setting parameter for the corresponding command. It can be optional or more than one parameter
<param1_value></param1_value>	Value for the corresponding parameter setting

(2) Reader \rightarrow Server

The format of XML/HTTP response from reader to server is as follows:

Note:

- This document is applicable to CS463 web application 1.1.8 or above.

- All High-Level HTTP API query strings are *Case-Sensitive*.

1.3. HTTP Protocol Table

	query_string	Description	
Use	ers Management		
1	session_id= <login_session_id>&</login_session_id>	Adds a new user with name userna	me, password password, and
	command=addUser&	permission of accessing item.	
	username=username&	e.g.	
	password=password&	session_id= <login_session_id>&co</login_session_id>	ommand=addUser&usernam
	[desc=desc]	e=BruceLi&password=pw123&Sta	tus=1&LogFileConfiguratio
	[&item=permission]	n=1&DownloadLogFile=4&TagInv	ventory=1
	:		
		Valid attributes:	
		item	permission
		Status	1
		UserManagement	1, 2
		ForceLogout	4
		ReaderId	1, 2
		CompanyLabel	1, 2
		CapturePointName	1, 2
		AccessMode	1, 2
		CustomEmbeddedApplication	1, 2
		FrequencyConfiguration	1, 2
		OperationProfile	1, 2
		ConfigureLNAGain	1, 2
		MemoryInformation	1, 2
		PowerUpNotification	1, 2
		HeartBeatNotification	1, 2

	ReaderErrorNotification	1, 2
	GPIInterruptNotification	1, 2
	ConfigurationBackupRestore	1, 2
	LogFileConfiguration	1, 2
	DownloadLogFile	4
	ScheduledReboot	1, 2
	RebootSystem	4
	EthernetWiFiConfiguration	1, 2
	CloudServer	1, 2
	DataFormat	1, 2
	TimeSetting	1, 2
	TagGroup	1, 2
	TagDatabase	1, 2
	IOPortControl	1, 2, 4
	Trigger	1, 2
	ResultactionAction	1, 2
	Event	1, 2
	DisplayFormat	1, 2
	TagInventory	1
	FirmwareUpgrade	4
	SSLCertificate	1, 2
	TagFilter	1, 2
	Note:	
		execute or bitwise OR of
	result: xml version="1.0" ? <csl> <command/>addUserOK: </csl>	nmand>
2 session_id= <login_session_id>&</login_session_id>	Modify the <i>permission</i> of accessing	item of a user with name

command= <i>modUser</i> &	username.	
username= <i>username</i> &	e.g.	
[&item=permission]	session_id= <login_session_id>&c</login_session_id>	ommand=modUser&usern
:	me=BruceLi&Status=1&LogFileC	Configuration=3&Downloa
	ogFile=4&TagInventory=1&Even	t=3&DisplayFormat=3
	Valid attributes:	
	item	permission
	Status	1
	UserManagement	1, 2
	ForceLogout	4
	ReaderId	1, 2
	CompanyLabel	1, 2
	CapturePointName	1, 2
	AccessMode	1, 2
	CustomEmbeddedApplication	1, 2
	FrequencyConfiguration	1, 2
	OperationProfile	1, 2
	ConfigureLNAGain	1, 2
	MemoryInformation	1, 2
	PowerUpNotification	1, 2
	HeartBeatNotification	1, 2
	ReaderErrorNotification	1, 2
	GPIInterruptNotification	1, 2
	ConfigurationBackupRestore	1, 2
	LogFileConfiguration	1, 2
	DownloadLogFile	4
	ScheduledReboot	1, 2
	RebootSystem	4
	EthernetWiFiConfiguration	1, 2
	CloudServer	1, 2
	DataFormat	1, 2
	TimeSetting	1, 2

		TagGroup	1, 2
		TagDatabase	1, 2
		IOPortControl	1, 2, 4
		Trigger	1, 2
		ResultactionAction	1, 2
		Event	1, 2
		DisplayFormat	1, 2
		TagInventory	1
		FirmwareUpgrade	4
		SSLCertificate	1, 2
		TagFilter	1, 2
		Note:	
		•	execute or bitwise OR of
		these values	
		result:	
		xml version="1.0" ?	
		<csl> <command/>modUser</csl>	mmand
		<ack>OK:</ack>	Illinariu>
3	session_id= <login_session_id>&</login_session_id>	Removes the user with name <i>userna</i>	ите.
	command= <i>delUser</i> &	e.g.	
	username= <i>username</i>	session_id= <login_session_id>&co</login_session_id>	mmand=delUser&usernam
		e=Bruce Li	
		result:	
		xml version="1.0" ?	
		<csl> <command/>delUser<td>mand></td></csl>	mand>
		<ack>OK:</ack>	
4	session_id= <login_session_id>&</login_session_id>	Sets the user password for the user	with name username.
	command=setUserPassword&	Only "root" user can invoke this con	mmand.

```
username=username&
password=password
                              e.g.
                              session_id=<login_session_id>&command=setUserPassword&
                              username=Bruce Li&password=mod123
                              result:
                               <?xml version="1.0" ?>
                               <CSL>
                                 <Command>setUserPassword</Command>
                                 <Ack>OK:</Ack>
                               </CSL>
                              List all users information.
session id=<login session id>&
command=listUsers
                              e.g.
                              session_id=<login_session_id>&command=listUsers
                              result:
                               <?xml version="1.0" ?>
                               <CSL>
                                 <Command>listUsers</Command>
                                   <a href="top level administrator"</a>
                                     username="root"
                                     Status="1"
                                     UserManagement="3"
                                     ForceLogout="4"
                                     ReaderId="3"
                                     CompanyLabel="3"
                                     CapturePointName="3"
                                     AccessMode="3"
                                     CustomEmbeddedApplication="3"
                                     FrequencyConfiguration="3"
                                     OperationProfile="3"
                                     ConfigureLNAGain="3"
                                     MemoryInformation="3"
                                     PowerUpNotification="3"
                                     HeartBeatNotification="3"
                                     ReaderErrorNotification="3"
                                     GPIInterruptNotification="3"
                                     ConfigurationBackupRestore="3"
                                     LogFileConfiguration="3"
                                     DownloadLogFile="4"
                                     ScheduledReboot="3"
                                     RebootSystem="4"
                                     EthernetWiFiConfiguration="3"
                                     CloudServer="3"
                                     DataFormat="3"
```

```
TimeSetting="3"
                                       TagGroup="3"
                                       TagDatabase="3"
                                       IOPortControl="7"
                                       Trigger="3"
                                       ResultantAction="3"
                                       Event="3"
                                       DisplayFormat="3"
                                       TagInventory="1"
                                       FirmwareUpgrade="4"
                                       SSLCertificate="3"
                                       ReaderErrorNotification="3"
                                    TagFilter="3" /> <Account desc=""
                                       username="BruceLi"
                                       Status="1"
                                       LogFileConfiguration="3"
                                       DownloadLogFile="4"
                                       Event="3"
                                       DisplayFormat="3"
                                       TagInventory="1"/>
                                </CSL>
command=login&
                                Login is required for access to the reader.
                                Login is successful if password for the user is correct.
username=username&
password=password
                               e.g.
                               command=login&username=root&password=csl
                                result 1:
                                (Login successfully)
                                <?xml version="1.0" ?>
                                <CSL>
                                  <Command>login</Command>
                                  <Ack>OK: session_id=768f32f8</Ack>
                                </CSL>
                               result 2:
                                (If other user has already logged-in)
                                <?xml version="1.0" ?>
                                <CSL>
                                  <Command>login</Command>
                                  <Error alreadyLoginIP="192.168.25.124"</pre>
                                       alreadyLoginUser="root" code="-10"
                                       msg="Error: Only one user can login
                                      the system at the same
```

		time!Another User root has already logged-in the system (by browser or API command) at location 192.168.25.124.Please logout the other user and retry login." />
7	session_id= <login_session_id>&</login_session_id>	Log out is recommended to ensure the security and integrity of
	command= <i>logout</i>	the system.
		e.g.
		session_id= <login_session_id>&command=logout</login_session_id>
		result:
		xml version="1.0" ?
		<csl> <command/>logout</csl>
		<ack>OK:</ack>
		1/632
8	command=forceLogout&	Force logout the system and intend to login another session for
	username=username&	operation.
	password=password	
		e.g.
		command=forceLogout&username=root&password= <passwor< th=""></passwor<>
		d>
		Remark: username must be "root"
		result:
		xml version="1.0" ? <csl></csl>
		<command/> forceLogout <ack>OK:</ack>
9	session_id= <login_session_id>&</login_session_id>	Set the auto logout time to the Edge Server such that the it will
	command=setAutoLogoutTime&	automatically be logout after the idle time, time.
	time=time	
		e.g.
		session_id= <login_session_id>&command=setAutoLogoutTim</login_session_id>

```
e&time=30
                                   Valid attributes:
                                   time : unit = minute, 0 = login session never expire
                                   result:
                                   <?xml version="1.0" ?>
                                   <CSL>
                                     <Command>setAutoLogoutTime</Command>
                                     <Ack>OK:</Ack>
                                   </CSL>
10 | session_id=<login_session_id>&
                                   Get the auto logout time.
   command=getAutoLogoutTime
                                   e.g.
                                   session_id=<login_session_id>&command=getAutoLogoutTi
                                   me
                                   result:
                                   <?xml version="1.0" ?>
                                   <CSL>
                                     <Command>getAutoLogoutTime</Command>
                                     <Logout time="30" unit="minute" />
                                   </CSL>
System Management
11 | session_id=<login_session_id>&
                                  Set Reader ID.
   command=setReaderID&
                                  e.g.
   reader_id=reader_id&desc=desc
                                   session_id=<login_session_id>&command=setReaderID&read
                                   er_id=CS463 Demo Reader&desc=Demo Reader
                                   result:
                                   <?xml version="1.0" ?>
                                   <CSL>
                                     <Command>setReaderID</Command>
                                     <Ack>OK:</Ack>
                                   </CSL>
```

12	session_id= <login_session_id>&</login_session_id>	Get Reader ID.	
	command=getReaderID	e.g.	
		session_id= <login_session_id>&command=getReaderID</login_session_id>	
		result:	
		<csl></csl>	
13	session_id= <login_session_id>&</login_session_id>	To set Access Mode of the reader.	
	command=setAccessMode&		
	mode=mode	e.g.	
		session_id= <login_session_id>&command=setAccessMode&</login_session_id>	
		mode=http	
		Valid attributes:	
		mode: high <i>or</i> http = High Level HTTP API Mode	
		low = Low Level Mach1 API Mode	
		cslapi = CSL Unified API High Level Mode	
		cslapilow = CSL Unified API Low Level Mode	
		llrp = LLRP API Mode	
		bluetooth = CS108 Bluetooth API Mode	
		customembedded = Custom Embedded RFID HTTP	
		API Mode	
		cslapirs232 = CSL Unified API Mode via RS232	
		Control Serial Port	
		cslapilowrs232 = CSL Unified API Low Level	
		Mode via RS232 Control Serial Port	
		result:	
		xml version="1.0" ? <csl></csl>	
		<command/> setAccessMode <ack>OK:</ack>	

```
</CSL>
14 | session id=<login session id>&
                                To get Access Mode of the reader.
   command=getAccessMode
                                e.g.
                                session_id=<login_session_id>&command=getAccessMode
                                result 1:
                                <?xml version="1.0" ?>
                                <CSL>
                                  <Command>getAccessMode</Command>
                                  <Access mode="1" name=" HTTP/XML" />
                                </CSL>
                                result 2:
                                <?xml version="1.0" ?>
                                <CSL>
                                  <Command>getAccessMode</Command>
                                  <Access mode="2" name=" CS461 Low
                                      Level API (MACH1)" />
                                </CSL>
                                result 3:
                                <?xml version="1.0" ?>
                                <CSL>
                                  <Command>getAccessMode</Command>
                                  <Access mode="3" name="CSL Unified
                                      API/High Level" />
                                </CSL>
                                result 4:
                                <?xml version="1.0" ?>
                                <CSL>
                                  <Command>getAccessMode</Command>
                                  <Access mode="4" name="LLRP" />
                                </CSL>
                                result 5:
                                <?xml version="1.0" ?>
                                <CSL>
                                  <Command>getAccessMode</Command>
                                  <Access mode="5" name="CS108
                                      Bluetooth API" />
                                </CSL>
```

```
result 6:
                                <?xml version="1.0" ?>
                                <CSL>
                                  <Command>getAccessMode</Command>
                                  <Access mode="6" name="Custom
                                      Embedded RFID HTTP" />
                                </CSL>
                                result 7:
                                <?xml version="1.0" ?>
                                <CSL>
                                  <Command>getAccessMode</Command>
                                  <Access mode="7" name="CSL Unified
                                      API/High Level via RS232 Control Serial
                                      Port" />
                                </CSL>
                                result 8:
                                <?xml version="1.0" ?>
                                <CSL>
                                  <Command>getAccessMode</Command>
                                  <Access mode="8" name="CSL Unified
                                      API/Low Level "/>
                                </CSL>
                                result 9:
                                <?xml version="1.0" ?>
                                <CSL>
                                  <Command>getAccessMode</Command>
                                  <Access mode="9" name="CSL Unified
                                      API/Low Level via RS232 Control Serial
                                      Port" />
                                </CSL>
15 | session_id=<login_session_id>&
                                Set Custom Embedded RFID Application.
   command=setEmbeddedRFIDApp& e.g.
   path=path&
                                session_id=<login_session_id>&command=setEmbeddedRFI
   cmd=cmd
                                DApp&path=%2Fopt%2Fcsl_embedded_rfid_example_2.6_20
                                190828&cmd=.%2Fexample+-conf+config_HK.txt
                                result:
                                <CSL>
                                  <Command>setEmbeddedRFIDApp</Command>
```

		<ack>OK:</ack>
16	session_id= <login_session_id>&</login_session_id>	Get Reader ID.
10		e.g.
	command-genzinoeuaeara 1271pp	session_id= <login_session_id>&command=getReaderID</login_session_id>
		session_id= \login_session_id> &command=getxeuderns
		result:
		<csl></csl>
		<pre><command/>getEmbeddedRFIDApp <embeddedrfidapp< pre=""></embeddedrfidapp<></pre>
		path="/opt/csl_embedded_rfid_example_2
		.6_20190828" cmd="./example -conf config_HK.txt" />
17	session_id= <login_session_id>&</login_session_id>	Configure Operation Profile.
,	command=setOperProfile&	comigure operation Frome.
	profile_id= <i>profile_id&</i>	e.g. 1 (same transmit power on all antenna)
	linkProfile= linkProfile&	http://192.168.25.160/API?session_id=75cf3f18&com
	populationEst= populationEst&	mand=setOperProfile&profile id=Default Profile&
	sessionNo=sessionNo&	linkProfile=1&populationEst=50&sessionNo=0⌖=
	target=target&	2&queryAlgorithm=DynamicQ&reflectedPowerThreshol
	queryAlgorithm=queryAlgorithm&	d=24&tagModel=ANY&antenna port=1,2,3,4&transmit
	reflectedPowerThreshold=reflectedP	Power=30.00&dwellTime1=2000&dwellTime2=2000&d
	owerThreshold&	wellTime3=2000&dwellTime4=2000
	tagModel=tagModel&	e.g. 2 (different transmit power on each antenna)
	antenna_port=antenna_port	http://192.168.25.160/API?session_id=75cf3f18&com
	[&transmitPower=transmitPower]	mand=setOperProfile&profile id=Default_
	[&transmitPower1=transmitPower1	Profile&linkProfile=1&populationEst=50&sessionNo=0&
	&transmitPower2=transmitPower2	target=0&queryAlgorithm=DynamicQ&reflectedPowerT
	&transmitPower3=transmitPower3	hreshold=24&tagModel=ANY&antenna port=1,2,3,4&tr
	&transmitPower4=transmitPower4	ansmitPower1=21.00&transmitPower2=22.00&transmit
	&transmitPower5=transmitPower5	Power3=23.00&transmitPower4=24.00&dwellTime1=2
	&transmitPower6=transmitPower6	000&dwellTime2=2000&dwellTime3=2000&dwellTime4
	&transmitPower7=transmitPower7	=2000
	&transmitPower8=transmitPower8	

&transmitPower9=transmitPower9	Valid attributes	:
&transmitPower10=transmitPower1	linkProfile:	0 = Multipath Interface Resistance
o		1 = Range/Dense Reader
&transmitPower11=transmitPower1		2 = Range/Throughput/Dense Reader
1		3 = Max Throughput
&transmitPower12=transmitPower1	populationEst :	1 - 8192
2	sessionNo : 0 =	S_{0} , $1 = S_{1}$, $2 = S_{2}$, $3 = S_{3}$
&transmitPower13=transmitPower1	target: $0 = A$, 1	= B, 2 = A/B Togggle
3	queryAlgorithm	n : FixedQ, DynamicQ
&transmitPower14=transmitPower1	reflectedPower'	Threshold: $1.0 - 32.0$ in step of 0.1 dBm
4	tagModel : AN	Y, Magnus_S2, Magnus_S3, Ctesius
&transmitPower15=transmitPower1	antenna_port :	1-16, any combinations with comma
5		separated, e.g. 1,2,3,4
&transmitPower16=transmitPower1	transmitPower	: 0.0 - 32.0 in step of 0.1 dBm
6]	transmitPower1	-16:0.0-32.0 in step of 0.1 dBm
[&dwellTime1= dwellTime1	dwellTime1 – 1	6 : unit=ms, >= 0ms
&dwellTime2= dwellTime2	retry : >= 0	
&dwellTime3= dwellTime3	tagFocus: true,	false (if it is true, sessionNo is set to 1 and
&dwellTime4= dwellTime4	targe	et is set to 0 automatically)
&dwellTime5 = dwellTime5		
&dwellTime6= dwellTime6	Optional attribu	ites:
&dwellTime7= dwellTime7	memoryBank1	: Bank0, Bank1, Bank2, Bank3
&dwellTime8= dwellTime8	memoryBank10	Offset: $\geq = 0$
&dwellTime9= dwellTime9	memoryBank1I	Length: unit=no. of words, >= 0
&dwellTime10= dwellTime10	memoryBank2	: Bank0, Bank1, Bank2, Bank3
&dwellTime11= dwellTime11	memoryBank20	Offset: $\geq = 0$
&dwellTime12= dwellTime12	memoryBank2I	Length: unit=no. of words, >= 0
&dwellTime13= dwellTime13	fastId : true, fal	se
&dwellTime14= dwellTime14	minOnChipRSS	SI: 0 – 31, unit=dBm
&dwellTime15= dwellTime15	maxOnChipRS	SI: 0-31, unit=dBm
&dwellTime16= dwellTime16]	moistAvgWind	ow: 1 – 50
[&memoryBank1=memoryBank1	tempAvgWindo	ow: 1 – 50
&memoryBank1Offset=memoryBan	reconfigAntenn	aPortError : true, false
k1Offset	retryErrorAnter	nnaPortTime : unit=second, >=0, 0=never retry

	&memoryBank1Length=memoryBa	preFilter1 – 7:	ID of Tag Filter, the type of the filter must be
	nk1Length]		PRE_FILTER
	[&memoryBank2=memoryBank2	postFilter:	ID of Tag Filter, the type of the filter must be
	&memoryBank2Offset=memoryBan		POST_FILTER
	k2Offset		
	&memoryBank2Length=memoryBa	Note: If tagModel is A	NY, there can be 7 pre-filters.
	nk2Length]	If tagModel is M	Magnus_S2, there can be 5 pre-filters.
	[&retry=retry]		Magnus_S3, there can be 4 pre-filters. Ctesius, there can be 6 pre-filters.
	[&tagFocus=tagFocus]		costus, there can be o pre interes.
	[&fastId=fastId]	result : xml version</td <td>="1 0" ?></td>	="1 0" ?>
	[&minOnChipRSSI=minOnChipRSS	<csl></csl>	
	<i>I</i>]	<command <ack=""/> OK: <	>setOperProfile
	[&maxOnChipRSSI=maxOnChipRS		
	SI]		
	[&moistAvgWindow=moistAvgWind		
	ow]		
	[&tempAvgWindow=tempAvgWindo		
	w]		
	[&reconfigAntennaPortError=reconf		
	igAntennaPortError]		
	[&retryErrorAntennaPortTime=retry		
	ErrorAntennaPortTime]		
	[&preFilter1=preFilter1]		
	[&preFilter2=preFilter2]		
	[&preFilter3=preFilter3]		
	[&preFilter4= <i>preFilter4</i>]		
	[&preFilter5= <i>preFilter5</i>]		
	[&preFilter6= <i>preFilter6</i>]		
	[&preFilter7= <i>preFilter7</i>]		
	[&postFilter=postFilter]		
18	session_id= <login_session_id>&</login_session_id>	Get Operation	Profile information.
	command=getOperProfile	e.g.	
		•	8.25.160/API?session_id=7C1286DE&co

```
mmand=getOperProfile
<?xml version="1.0" ?>
<CSL>
  <Command>getOperProfile</Command>
 <ProfileList>
    cprofile profile_id="Default Profile"
        active="true"
        linkProfile="1"
        populationEst="50"
        sessionNo="0"
        target="2"
        queryAlgorithm="DynamicQ"
        reflectedPowerThreshold="24.0"
        tagModel="ANY"
        retry="0"
        tagFocus="false"
        fastId="false"
        minOnChipRSSI="16"
        maxOnChipRSSI="21"
        moistAvgWindow="5"
        tempAvgWindow="5"
        reconfigAntennaPortError="false"
        retryErrorAntennaPortError="0"
        antenna_port="1,2,3,4"
        transmitPower="30.0"
        transmitPower1="30.0"
        transmitPower2="30.0"
        transmitPower3="30.0"
        transmitPower4="30.0"
        transmitPower5="30.0"
        transmitPower6="30.0"
        transmitPower7="30.0"
        transmitPower8="30.0"
        transmitPower9="30.0"
        transmitPower10="30.0"
        transmitPower11="30.0"
        transmitPower12="30.0"
        transmitPower13="30.0"
        transmitPower14="30.0"
        transmitPower15="30.0"
        transmitPower16="30.0"
        dwellTime1="2000"
        dwellTime2="2000"
        dwellTime3="2000"
        dwellTime4="2000"
        dwellTime5="2000"
        dwellTime6="2000"
        dwellTime7="2000"
        dwellTime8="2000"
        dwellTime9="2000"
```

```
dwellTime10="2000"
                                            dwellTime11="2000"
                                            dwellTime12="2000"
                                            dwellTime13="2000"
                                            dwellTime14="2000"
                                            dwellTime15="2000"
                                            dwellTime16="2000"
                                            memoryBank1="Bank2"
                                            memoryBank1Offset="0"
                                            memoryBank1Length="2"
                                            memoryBank2="Bank3"
                                            memoryBank2Offset="0"
                                            memoryBank2Length="2"/>
                                      </ProfileList>
                                    </CSL>
19. | session_id=<login_session_id>&
                                   Remove operation profile.
   command=delOperProfile&
   profile_id= profile_id
                                   e.g.
                                   session_id=<login_session_id>&command=delOperProfile&se
                                   rver_id=ExampleProfile
                                   result:
                                    <?xml version="1.0" ?>
                                      <Command>delOperProfile</Command>
                                      <Ack>OK:</Ack>
                                    </CSL>
20. | session_id=<login_session_id>&
                                   Set RF LNA Gain and IF LNA Gain settings.
   command=setRFLNAIFLNAGain& e.g.
   rf_lna_compression_mode=rf_lna_c | session_id=<login_session_id>&command=setRFLNAIFLNA
   ompression mode&
                                   Gain&rf lna compression mode=1&rf lna gain=1dB&if lna
   rf_lna_gain=rf_lna_gain&
                                    _gain=24dB&agc_gain=-6dB
   if_lna_gain=if_lna_gain&
   agc_gain=agc_gain
                                    Valid attributes:
                                   rf_lna_compression_mode: 0, 1 (this must be 0 if rf_lna_gain
                                                           is 13dB)
                                   rf_lna_gain:
                                                 1dB, 7dB, 13dB
                                   if_lna_gain:
                                                 24dB, 18dB, 12dB, 6dB
                                   agc_gain:
                                                 -12dB, -6dB, 0dB, 6dB
```

```
result:
                                  <?xml version="1.0" ?>
                                  <CSL>
                                   <Command>setRFLNAIFLNAGain</Command>
                                    <Ack>OK:</Ack>
                                  </CSL>
21. session_id=<login_session_id>&
                                 Get RF LNA Gain and IF LNA Gain settings.
   command=getRFLNAIFLNAGain
                                 e.g.
                                 session_id=<login_session_id>&command=getRFLNAIFLNA
                                 Gain
                                 result:
                                  <?xml version="1.0" ?>
                                  <CSL>
                                    <Command>getRFLNAIFLNAGain</Command>
                                    <Settings
                                        rf_Ina_compression_mode="1"
                                        rf_lna_gain="1dB"
                                       if_lna_gain="24dB"
                                        agc_gain="-6dB" />
                                  </CSL>
22. | session_id=<login_session_id>&
                                 Get system memory information.
   command=getRAMMemory
                                 session_id=<login_session_id>&command=getRAMMemory
                                 result:
                                 <?xml version="1.0" ?>
                                    <Command>getRAMMemory</Command>
                                   <SystemMemory>
                                      <Total>64638976</Total>
                                      <Used>50401280</Used>
                                      <Free>14237696</Free>
                                   </SystemMemory>
                                  </CSL>
23. session_id=<login_session_id>&
                                 Get flash memory information.
   command=getFlashMemory
                                 e.g.
                                 session_id=<login_session_id>&command=getFlashMemory
```

```
result:
                                 <?xml version="1.0" ?>
                                 <CSL>
                                   <Command>getFlashMemory</Command>
                                   <FlashMemory>
                                     <Total>4194304</Total>
                                     <Used>3212000</Used>
                                     <Free>982304</Free>
                                   </FlashMemory>
                                 </CSL>
24. command=getReaderStatus&
                                 Get the reader run-time status for inspection without login the
   username=username&
                                 reader.
   password=password
                                 e.g.
                                 http://192.168.25.160/API?command=getReaderStatu
                                 s&username=root&password=csl
                                 result:
                                 <?xml version="1.0" ?>
                                 <CSL>
                                   <Command>getReaderStatus</Command
                                   <Model name="CS463-2" protocol="EPC
                                       Class1 Gen 2"/>
                                   <Reader desc="CS463 Demo Reader"
                                       reader id="Demo Reader"
                                       reader serial number="ABC0123456789"
                                       pcb_serial_number="DEF9876543210024" />
                                   < Reader Version
                                       cs108 bluetooth api library="1.0.2"
                                       cs461_low_level_api_mach1_library=" 1.0.4"
                                       csl_unified_api_library="1.0.3"
                                       java=" 1.8.0_221"
                                       ini library="1.0.4"
                                       llrp library="1.0.7"
                                       os=" Linux
                                       v4.14.78-imx_4.14.78_1.0.0_ga+g
                                       94da7bd"
                                       pcb version="2.4"
                                       rfid firmware="2.6.29"
                                       web_application="1.1.9" />
                                   <Timezone daylight saving="0"
                                       tz="GMT+08:00" />
                                   <Logout time="30" unit="minute" />
                                   <UserStatus client_ip="192.168.25.126"</pre>
```

```
login status="yes"
                                         session_id="00000000"
                                         username="root" />
                                     <AccessMode mode="1" name="HTTP/XML" />
                                     <a href="#"><ActiveOperationProfile</a>
                                         antenna_power="1:30.0,2:30.0,3:
                                         30.0,4:30.0,"
                                         profile_id="Default Profile" />
                                     <ActiveEventList>
                                       <Event desc="Event Demo"
                                           event_id="DemoEvent" />
                                       <Event desc="" event_id="e45" />
                                     </ActiveEventList>
                                     <CurrentLocalTime day="9" hour="15"
                                         minute="41" month="3" second="9"
                                         year="2020" />
                                     <CurrentUTCTime day="9" hour="7"</pre>
                                         minute="41" month="3" second="9"
                                         year="2020" />
                                   </CSL>
25. command=healthCheck&
                                  Make a health check of the reader without login the reader first.
   username=username&
   password=password
                                  e.g.
                                  http://192.168.25.160/API?command=healthCheck&us
                                   ername=root&password=csl
                                  result:
                                   <?xml version="1.0" ?>
                                   <CSL>
                                     <Command>healthCheck</Command>
                                     <result checkTime="Mon Mar 9 15:43:20 2020"</pre>
                                         freeRAM="15416368" upTime="16.25" />
                                   </CSL>
26. session id=<login session id>&
                                  Get the unresolved errors from the reader.
   command=getReaderError
                                  e.g.
                                  http://192.168.25.160/API?session_id=f13b3074&com
                                   mand=getReaderError
                                   result:
                                   <?xml version="1.0" ?>
                                   <CSL>
```

```
<Command>getReaderError</Command>
                                    <readerError error_code="0309"
                                         desc=" Reverse Power Too High - may be
                                         antenna mismatch"
                                         antennaPort="2"
                                         reflected_power="26"
                                         reflected_power_threshold="24"
                                         upTime="16.25" />
                                    <readerError error_code="0309"
                                         desc=" Reverse Power Too High - may be
                                         antenna mismatch"
                                         antennaPort="3"
                                         reflected_power="27"
                                         reflected power threshold="24"
                                         upTime="16.25" />
                                  </CSL>
27. | session_id=<login_session_id>&
                                  Reboot the system.
   command=restartSystem
   [&type=type]
                                  e.g.
                                  http://192.168.25.160/API?session_id=f13b3074&com
                                  mand=restartSystem&type=WARM_REBOOT
                                  Valid attributes:
                                           USB_HUB_REBOOT = reboot usb hub controller
                                  type:
                                           USB_HUB_RFID_REBOOT = reboot usb hub
                                               controller and modem controller
                                           WARM_REBOOT = reboot reader without reset
                                               power
                                           COLD_REBOOT = reboot reader by reseting
                                               power
                                  result:
                                  <?xml version="1.0" ?>
                                    <Command>restartSystem</Command>
                                    <Ack>OK:</Ack>
                                  </CSL>
28. | session_id=<login_session_id>&
                                  To check if the reader is on-line.
   command=isOnline
                                  e.g.
```

```
http://192.168.25.160/API?session_id=f13b3074&com
                                   mand=isOnline
                                  result:
                                   <?xml version="1.0" ?>
                                   <CSL>
                                     <Command>isOnline</Command>
                                     <Ack>OK: Online,CS463.</Ack>
                                   </CSL>
29. | session_id=<login_session_id>&
                                   Get the scheduled restart settings
   command=getScheduledRestart
                                  e.g.
                                   http://192.168.25.160/API?session_id=f13b3074&com
                                   mand=getScheduledRestart
                                  result:
                                   <?xml version="1.0" ?>
                                   <CSL>
                                     <Command>getScheduledRestart</Command>
                                     <ScheduledRebootList>
                                       <scheduleReboot
                                            type="WARM_REBOOT"
                                            enable="false"
                                            mode="Monday"
                                            month=""
                                            day=""
                                           time1="17:47"
                                           time2=""/>
                                     </ScheduledRebootList>
                                   </CSL>
30. | session_id=<login_session_id>&
                                   The system may be scheduled to restart.
   command=setScheduledRestart
   &type=type
                                   Valid attributes:
   &mode=mode
                                           USB_HUB_REBOOT = reboot usb hub controller
                                  type:
                                            USB_HUB_RFID_REBOOT = reboot usb hub
   &enable=enable
   [&month=month]
                                                controller and modem controller
                                           WARM REBOOT = reboot reader without reset
   [\&day=day]
   &time1=time1
                                                power
   [&time2=time2]
                                            COLD REBOOT = reboot reader by reseting
```

```
power
mode:
        ANNUAL, SEMI_ANNUAL, QUARTERLY,
        BI_MONTHLY, MONTHLY, SUNDAY, MONDAY,
        TUESDAY, WEDNESDAY, THURSDAY, FRIDAY,
        SATURDAY, DAILY, TWICE_PER_DAY
enable:
        true, false
        1 – 12 for mode ANNUAL
month:
        1 - 6 for mode SEMI ANNUAL
        1 – 3 for mode QUARTERLY
        1 or 2 for mode BI_MONTHLY
        Not required for other mode
        1-31, only required for mode ANNUAL,
day:
        SEMI_ANNUAL, QUARTERLY, BI_MONTHLY,
        MONTHLY
time1:
        hh:mm (hh is hour in 24 hour format, mm
        is minute)
time2:
        hh:mm (hh is hour in 24 hour format, mm
        is minute), only required for mode
        TWICE_PER_DAY
e.g.
http://192.168.25.160/API?session_id=f13b3074&com
mand=setScheduledRestart&type=WARM REBOOT&mo
de=MONDAY&enable=true&time1=23:00
result:
if successful,
<?xml version="1.0" ?>
<CSL>
  <Command>setScheduledRestart</Command>
  <Ack>OK:</Ack>
</CSL>
if fail (the example is wrong mode),
<?xml version="1.0" ?>
<CSL>
  <Command>setScheduledRestart</Command>
```

		<error code="-10" msg="Error: mode is
not valid"></error>
31.	session_id= <login_session_id>&</login_session_id>	Add Power Up Notification to be sent to server.
	command=addPowerUpNotification	
	¬ification_id=notification_id	e.g.
	&type=type	session_id= <login_session_id>&command=addPowerUpNotifi</login_session_id>
	&server_id=server_id	cation¬ification_id=Example Power Up
	&data_format_id=data_format_id	Notification&type=HTTP POST&server_id=Example CSL
	&enable= <i>enable</i>	Demo Cloud Server&data_format_id=Example Power Up
		Notification Data Format&enable=true
		Valid attributes :
		type: HTTP POST, MQTT
		enable: true, false
		result:
		xml version="1.0" ? <csl></csl>
32.	session_id= <login_session_id>&</login_session_id>	Modify Power Up Notification to be sent to server.
	command=modPowerUpNotificatio	
	n	e.g.
	¬ification_id=notification_id	session_id= <login_session_id>&command=modPowerUpNoti</login_session_id>
	&type=type	fication¬ification_id=Example Power Up
	&server_id=server_id	Notification&type=HTTP POST&server_id=Example CSL
	&data_format_id=data_format_id	Demo Cloud Server&data_format_id=Example Power Up
	&enable= <i>enable</i>	Notification Data Format&enable=false
		Valid attributes : type : HTTP POST, MQTT

```
enable: true, false
                                   result:
                                   <?xml version="1.0" ?>
                                   <CSL>
                                     <Command>modPowerUpNotification
                                          </Command>
                                     <Ack>OK:</Ack>
                                   </CSL>
33. session_id=<login_session_id>&
                                  Remove Power Up Notification.
   command=delPowerUpNotification
   &notification_id=notification_id
                                  e.g.
                                   session_id=<login_session_id>&command=delPowerUpNotifi
                                   cation&notification_id=Example Power Up Notification
                                   result:
                                   <?xml version="1.0" ?>
                                   <CSL>
                                     <Command>delPowerUpNotification
                                          </Command>
                                     <Ack>OK:</Ack>
                                   </CSL>
34. session_id=<login_session_id>&
                                   List Power Up Notification.
   command=listPowerUpNotification
                                   e.g.
                                   session_id=<login_session_id>&command=listPowerUpNotifi
                                   cation
                                   result:
                                   <?xml version="1.0" ?>
                                   <CSL>
                                     <Command>listPowerUpNotification
                                          </Command>
                                     <PowerUpNotificationList>
                                       <notification data_format_id="Example Power
                                            Up Notification Data Format"
                                            enable="false"
                                            notification_id="Example Power Up
                                            Notification"
                                            server_id="Example CSL Demo Cloud
```

		Server" type="HTTP POST" /> <notification data_format_id="Example Power Up Notification Data Format" enable="false" notification_id="Example Power Up Notification to MQTT Broker" server_id="Example MQTT Broker" type="MQTT"></notification>
35.	session_id= <login_session_id>&</login_session_id>	Add Reader Error Notification to be sent to server.
	command=addReaderErrorNotifica	
	tion	e.g.
	¬ification_id=notification_id	session_id= <login_session_id>&command=addReaderErrorNo</login_session_id>
	&type=type	tification¬ification_id=Example Reader Error
	&server_id=server_id	Notification&type=HTTP POST&server_id=Example CSL
	&data_format_id=data_format_id	Demo Cloud Server&data_format_id=Example Reader Error
	&enable= <i>enable</i>	Notification Data Format&enable=true
		Valid attributes: type: HTTP POST, MQTT enable: true, false
		result: xml version="1.0" ? <csl> <command/>addReaderErrorNotification <ack>OK:</ack> </csl>
36.	session_id= <login_session_id>&</login_session_id>	Modify Reader Error Notification to be sent to server.
	command=modReaderErrorNotific	
	ation	e.g.
	¬ification_id=notification_id	session_id= <login_session_id>&command=modReaderErrorN</login_session_id>
	&type=type	otification¬ification_id=Example Reader Error
	&server_id=server_id	Notification&type=HTTP POST&server_id=Example CSL
	&data_format_id=data_format_id	Demo Cloud Server&data_format_id=Example Reader Error

	&enable=enable	Notification Data Format&enable=false
		Valid attributes :
		type: HTTP POST, MQTT
		enable : true, false
		result:
		xml version="1.0" ?
		<csl> <command/>modReaderErrorNotification</csl>
		<ack>OK:</ack>
37.	session_id= <login_session_id>&</login_session_id>	Remove Reader Error Notification.
	command=delReaderErrorNotificat	
	ion	e.g.
	¬ification_id=notification_id	session_id= <login_session_id>&command=delReaderErrorNo</login_session_id>
		tification¬ification_id=Example Reader Error Notification
		result:
		xml version="1.0" ?
		<csl> <command/>delReaderErrorNotification</csl>
		<ack>OK:</ack>
		C/CSL>
38.	session_id= <login_session_id>&</login_session_id>	List Reader Error Notification.
	command=listReaderErrorNotificat	
	ion	e.g.
		session_id= <login_session_id>&command=listReaderErrorNo</login_session_id>
		tification
		result:
		xml version="1.0" ?
		<csl> <command/>listReaderErrorNotification</csl>
		<readererrornotificationlist></readererrornotificationlist>

```
<notification data format id="Example Reader
                                             Error Notification Data Format"
                                              enable="false"
                                             notification_id="Example Reader Error
                                             Notification"
                                             server id="Example CSL Demo Cloud
                                             Server"
                                             type="HTTP POST" />
                                         <notification data_format_id="Example Reader
                                             Error Notification Data Format"
                                              enable="false"
                                             notification_id="Example Reader Error
                                             Notification to MQTT Broker"
                                             server_id="Example MQTT Broker"
                                             type="MQTT" />
                                       </ReaderErrorNotificationList>
                                     </CSL>
39. |session_id=<login_session_id>&
                                    Add GPI Interrupt Notification to be sent to server.
   command=addGPIInterruptNotifica
   tion
                                    e.g.
    &notification id=notification id
                                    session id=<login session id>&command=addGPIInterruptN
    &interrupt_type=interrupt_type
                                    otification&notification_id=Example GPI Interrupt
                                    Notification&interrupt_type=Rising Edge&gpi_port=1
    &gpi_port=gpi_port
    &type=type
                                    &type=HTTP POST&server_id=Example CSL Demo Cloud
    &server_id=server_id
                                    Server&data_format_id=Example GPI Interrupt Notification
                                    Data Format&enable=true
    &data_format_id=data_format_id
    &enable=enable
                                    Valid attributes:
                                    interrupt_type: Rising Edge, Falling Edge, Both
                                    gpi_port:
                                                  1, 2, 3 or 4
                                    type:
                                                  HTTP POST, MQTT
                                    enable:
                                                  true, false
                                    result:
                                     <?xml version="1.0" ?>
                                     <CSL>
                                       <Command>addGPIInterruptNotification
                                           </Command>
                                      <Ack>OK:</Ack>
                                     </CSL>
40. session id=<login session id>&
                                    Modify GPI Interrupt Notification to be sent to server.
```

	command=modGPIInterruptNotific	
	ation	e.g.
	¬ification_id=notification_id	session_id= <login_session_id>&command=modGPIInterruptN</login_session_id>
	&interrupt_type=interrupt_type	otification¬ification_id=Example GPI Interrupt
	&gpi_port=gpi_port	Notification&interrupt_type=Falling Edge&gpi_port=1
	&type=type	&type=HTTP POST&server_id=Example CSL Demo Cloud
	&server_id=server_id	Server&data_format_id=Example GPI Interrupt Notification
	&data_format_id=data_format_id	Data Format&enable=false
	&enable= <i>enable</i>	
		Valid attributes:
		interrupt_type: Rising Edge, Falling Edge, Both
		gpi_port: 1, 2, 3 or 4
		type: HTTP POST, MQTT
		enable: true, false
		result:
		xml version="1.0" ?
		<csl> <command/>modGPIInterruptNotification</csl>
		<ack>OK:</ack>
41.	session_id= <login_session_id>&</login_session_id>	Remove GPI Interrupt Notification.
	command=delGPIInterruptNotificat	
	ion	e.g.
	¬ification_id=notification_id	session_id= <login_session_id>&command=delGPIInterruptNo</login_session_id>
		tification¬ification_id=Example GPI Interrupt Notification
		result:
		xml version="1.0" ?
		<csl></csl>
		<command/> delGPIInterruptNotification
		<ack>OK:</ack>
42.	session_id= <login_session_id>&</login_session_id>	List GPI Interrupt Notification.
	command=listGPIInterruptNotificat	
	ion	e.g.

```
session id=<login session id>&command=listGPIInterruptNo
                                     tification
                                     result:
                                     <?xml version="1.0" ?>
                                     <CSL>
                                       <Command>listGPIInterruptNotification
                                            </Command>
                                       <GPIInterruptNotificationList>
                                          <notification data format id="Example GPI
                                              Interrupt Notification Data Format"
                                              enable="false"
                                              qpi port="1"
                                              interrupt_type="Falling Edge"
                                              notification id="Example GPI Interrupt
                                              Notification"
                                              server id="Example CSL Demo Cloud
                                              Server"
                                              type="HTTP POST" />
                                       </GPIInterruptNotificationList>
                                     </CSL>
43. |session_id=<login_session_id>&
                                     Add Heart Beat to be sent to server.
   command=addHeartBeat
    &heart_beat_id=heart_beat_id
                                     e.g.
    &type=type
                                     session_id=<login_session_id>&command=addHeartBeat&he
    &interval=interval
                                     art_beat_id=Heart Beat to Demo Cloud Server&type=HTTP
    &enable=enable
                                     POST&interval=60&server id=Example CSL Demo Cloud
    [&address=address]
                                     Server&data_format_id=Example Heart Beat Data
                                     Format&enableReset=true&resetPort=ethernet&tryBeforeRese
    [&server_id=server_id]
    [&data_format_id=data_format_id]
                                     t=5&enable=true
    [&enableReset=enableReset]
    [&resetPort=resetPort]
                                     Valid attributes:
   [&tryBeforeReset=tryBeforeReset]
                                              ICMP Ping, HTTP POST, MQTT, arp, arp Gateway
                                     type:
                                     interval: unit=second, 30 – 86400
                                     enable:
                                              true, false
                                     address: required if type is ICMP Ping
                                     server_id : required if type is HTTP POST or MQTT
                                     data_format_id : required if type is HTTP POST or MQTT
                                     enableReset: required if type is not arp
```

```
resetPort: ethernet, wifi or both, required if enableReset is true
                                      tryBeforeReset : 1 - 10, required if enableReset is true
                                      result:
                                      <?xml version="1.0" ?>
                                        <Command>addHeartBeat</Command>
                                        <Ack>OK:</Ack>
                                      </CSL>
44. | session_id=<login_session_id>&
                                      Modify Heart Beat to be sent to server.
    command=modHeartBeat
    &heart_beat_id=heart_beat_id
                                      e.g.
                                      session_id=<login_session_id>&command=modHeartBeat&he
    &type=type
    &interval=interval
                                      art_beat_id=Heart Beat to Demo Cloud Server&type=HTTP
    &enable=enable
                                      POST&interval=60&server_id=Example CSL Demo Cloud
    [&address=address]
                                      Server&data_format_id=Example Heart Beat Data
    [&server_id=server_id]
                                      Format&enableReset=true&resetPort=ethernet&tryBeforeRese
                                      t=5&enable=false
    [&data_format_id=data_format_id]
    [&enableReset=enableReset]
    [&resetPort=resetPort]
                                      Valid attributes:
    [&tryBeforeReset=tryBeforeReset]
                                      type:
                                                ICMP Ping, HTTP POST, MQTT, arp, arp Gateway
                                      interval: unit=second, 30 – 86400
                                      enable:
                                               true, false
                                      address: required if type is ICMP Ping
                                      server_id : required if type is HTTP POST or MQTT
                                      data_format_id : required if type is HTTP POST or MQTT
                                      enableReset: required if type is not arp
                                      resetPort: ethernet, wifi or both, required if enableReset is true
                                      tryBeforeReset : 1 - 10, required if enableReset is true
                                      result:
                                      <?xml version="1.0" ?>
                                      <CSL>
                                        <Command>modHeartBeat</Command>
                                         <Ack>OK:</Ack>
                                      </CSL>
```

45.	session_id= <login_session_id>&</login_session_id>	Remove Heart Beat.
	command=delHeartBeat	
	&heart_beat_id=heart_beat_id	e.g.
	concure_seat_ra nean_seat_sa	session_id= <login_session_id>&command=delHeartBeat&hea</login_session_id>
		rt_beat_id=Heart Beat to Demo Cloud Server
		result:
		xml version="1.0" ?
		<csl></csl>
		<command/> delHeartBeat
		<ack>OK:</ack>
		YCSE
46.	session_id= <login_session_id>&</login_session_id>	List Heart Beat.
	command=listHeartBeat	
		e.g.
		session_id= <login_session_id>&command=listHeartBeat</login_session_id>
		Seeston_to togin_seession_to, ect of managed in section to the
		result:
		xml version="1.0" ?
		<csl></csl>
		<command/> listHeartBeat <heartbeatlist></heartbeatlist>
		<heartbeat <="" address="" th=""></heartbeat>
		data_format_id="Example Heart Beat Data Format"
		enable=" true "
		enableReset=" false "
		heart_beat_id="Heart Beat to Demo Cloud
		Server" interval="60"
		resetPort="ethernet"
		server_id="Example CSL Demo Cloud
		Server" tryBeforeReset="5"
		type="HTTP POST" />
		<heartbeat <="" address="" th=""></heartbeat>
		data_format_id="" enable=" true "
		enableReset=" true "
		heart_beat_id="ARPING of Local Gateway"
		interval=" 30 " resetPort=" ethernet "
Ь		research Cultillet

	T	
		server_id="" tryBeforeReset=" 5 "
		type="arping Gateway" />
47.	session_id= <login_session_id>&</login_session_id>	Upload file to the reader.
	command= <i>uploadFile</i> &	
	fileName=fileName	Here below is an example showing how to upload the SSL
		certificate that will be used for Secure Web Access via HTTP
		POST protocol written in C# (printed in blue color).
		<pre>HttpClient client = new HttpClient();</pre>
		var stream = new
		FileStream("C:\\temp\\certificate.pem",
		FileMode.Open);
		var content = new StreamContent(stream);
		var requestUri =
		"http://192.168.25.160/API?session_id=a33219dc&co
		mmand=uploadFile&fileName=certificate.pem";
		var response = await client.PostAsync(requestUri,
		content);
		result:
		xml version="1.0" ?
		<csl> <command/>uploadFile <ack>OK:</ack> </csl>
48.	session_id= <login_session_id>&</login_session_id>	Configure to use HTTP or HTTPS for accessing the web
	command=setSecureWebAccess	interface of the reader and the SSL certificate to be used for
	&useSelfSignedCert=useSelfSigned	HTTPS.
	Cert	
	[&certFile=certFile]	e.g. 1 (use the ex-factory self-signed certificate and key for
	[&keyFile=keyFile]	HTTPS)
	[&keyPassword=keyPassword]	session_id= <login_session_id>&command=setSecureWebAcc</login_session_id>
		ess&useSelfSignedCert=true
-	•	

```
e.g. 2 (use the client provided certificate and private key for
                                           HTTPS, the certificate and key files must be uploaded
                                           to the reader first by using the uploadFile command,
                                           the certificate and key files must be in PEM format,
                                           keyPassword must be provided if the key file is
                                           encrypted)
                                     session_id=<login_session_id>&command=setSecureWebAcc
                                     ess&useSelfSignedCert=false&certFile=cert.pem&keyFile=key
                                     pem
                                    e.g. 3 (use HTTP for web access)
                                     session_id=<login_session_id>&command=setSecureWebAcc
                                    ess&useSelfSignedCert=false&certFile=&keyFile=
                                     result:
                                     <?xml version="1.0" ?>
                                     <CSL>
                                       <Command>setSecureWebAccess</Command>
                                       <Ack>OK:</Ack>
                                     </CSL>
49. session id=<login session id>&
                                     Get the configuration of Secure Web Access.
   command=getSecureWebAccess
                                     e.g.
                                     session_id=<login_session_id>&command=getSecureWebAcc
                                     ess
                                     result:
                                     <?xml version="1.0" ?>
                                       <Command>getSecureWebAccess</Command>
                                       <SecureWebAccess certFile=""
                                            keyFile=""
                                            keyPassword=""
                                            useSelfSignedCert="false" />
                                     </CSL>
```

50.	session_id= <login_session_id>&</login_session_id>	Configure the certificate, private key files and key password of			
	command=setTwoWayAuth	the reader to be used for HTTPS two way authentication.			
	&certFile=certFile				
	&keyFile= <i>keyFile</i>	e.g. 1 (the certificate and key files must be uploaded to the			
	&keyPassword=keyPassword	reader first by using the uploadFile command, the			
		certificate and key files must be in PEM format,			
		keyPassword must be provided if the key file is			
		encrypted)			
		session_id= <login_session_id>&command=setTwoWayAuth&</login_session_id>			
		certFile=cert.pem&keyFile=key.pem			
		e.g. 2 (remove the certificate and key files from the reader)			
		session_id= <login_session_id>&command=setTwoWayAuth&</login_session_id>			
		certFile= &keyFile=			
		result:			
		xml version="1.0" ?			
		<csl> <command/>setTwoWayAuth</csl>			
		<ack>OK:</ack>			
51.	session_id= <login_session_id>&</login_session_id>	Get the configuration of Two Way Authentication.			
	command=getTwoWayAuth				
		e.g.			
		session_id= <login_session_id>&command=getTwoWayAuth</login_session_id>			
		result:			
		xml version="1.0" ?			
		<csl> <command/>getTwoWayAuth</csl>			
		<securewebaccess <="" certfile="" td=""></securewebaccess>			
		keyFile="" keyPassword=""/>			
Net	Network Management				

52.	session_id= <login_session_id>&</login_session_id>	Set Network Properties of the following setting:	
	command=setNetworkConfig&	Type, DHCP Mode, IP Address, Subnet Mask, Default	
	type=type&	Gateway, DNS Server 1 and DNS Server 2	
	[enable=enable&]		
	dhcpmode=dhcpmode	e.g.1	
	[&ip=ip	http://192.168.25.160/API?session id=a33219dc&com	
	&mask= <i>mask</i>	mand=setNetworkConfig&type=ethernet&dhcpmode=1	
	&gateway=gateway]		
	[&dns_server1=dns_server1]	e.g.2	
	[&dns_server2=dns_server2]	http://192.168.25.160/API?session_id=a33219dc&com	
	[&security=security]	mand=setNetworkConfig&type=wifi&dhcpmode=0&ip=	
	[&ssid=ssid	192.168.25.102&mask=255.255.255.0&gateway=192.	
	&psk=psk]	168.25.1&dns server1=192.168.25.2&dns server2=8.	
		8.8.8&security=wpa-psk&ssid=TestAP&psk=password	
		Valid attributes:	
		type: ethernet, wifi	
		enable: true or false, only valid for type wifi	
		dhcpmode: $0 = \text{Static IP}$	
		1 = DHCP Mode	
		security: none or wpa-psk, required for type wifi	
		result:	
		xml version="1.0" ?	
		<csl> <command/>setNetworkConfig <ack>OK:</ack> </csl>	
53.	 session_id= <login_session_id>&</login_session_id>	Get Network Properties such as IP Address, Subnet Mask,	
	command=getNetworkConfig	Default Gateway, MAC Address.	
		e.g. session_id= <login_session_id>&command=getNetworkConfig</login_session_id>	

```
result:
<?xml version="1.0" ?>
<CSL>
  <Command>getNetworkConfig</Command>
  <NetworkConfigList>
   <NetworkConfig type="ethernet"
      dhcpmode="0"
      gateway="192.168.25.1"
     ip="192.168.25.248"
     mask="255.255.2"
     MAC="00:0D:60:A5:8F:E3" />
   <NetworkConfig type="wifi"
     enable="true"
     dhcpmode="1"
      gateway="192.168.25.1"
     ip="192.168.25.238"
      mask="255.255.255.0"
     MAC="00:0D:60:34:56:78"
     security="wpa-psk"
     ssid="TestAP" />
  </NetworkConfigList>
</CSL>
```

Time and Timer Management

54.	session_id= <login_session_id>&</login_session_id>	Set system UTC date time as the parameters pass. Time is in	
	command=setDateTime&	the 24-hours format.	
	Year=year&Month=month&		
	Day=day&Hour=hour&	e.g.	
	Minute=minute&Second=second	session_id= <login_session_id>&command=setDateTime&Yea</login_session_id>	
		r=2020&Month=5&Day=1&Hour=15&Minute=32&Second=5	
		8	
55.	session_id= <login_session_id>&</login_session_id>	Set Time Zone and Daylight Saving Time (DST).	
	command=setTimeZone&	The setting can be read by calling getDateTime command.	
	time_zone=time_zone&dst=dst		
		e.g.	
		session_id= <login_session_id>&command=setTimeZone&</login_session_id>	
		time_zone=08:00&dst=0	

```
Valid attributes:
                                 time zone=<in hh:mm or -hh:mm format where hh=hour,
                                 mm=minute>
                                dst=-1,0,1
                                 result:
                                 <?xml version="1.0" ?>
                                 <CSL>
                                   <Command>setTimeZone</Command>
                                   <Ack>OK:</Ack>
                                 </CSL>
56. | session_id=<login_session_id>&
                                 Get date/time in the format of asctime() (ANSI C).
   command=getDateTime
                                 session_id=<login_session_id>&command=getDateTime
                                 result:
                                 <?xml version="1.0" ?>
                                 <CSL>
                                   <Command>getDateTime</Command>
                                   <DateTime>Wed Nov 29 09:43:48
                                       2020</DateTime>
                                   <UTCDateTime>Wed Nov 29 01:43:48
                                       2020</UTCDateTime>
                                   <TimeZone>GMT+08:00</TimeZone>
                                   <DaylightSavingTime>
                                   <UpTime>325.23</UpTime>
                                 </CSL>
57. | session_id=<login_session_id>&
                                 Get NTP server information.
   command=getNTP
                                 e.g.
                                 http://192.168.25.160/API?session_id=12AC12DE&co
                                 mmand=getNTP
                                 result:
                                 <?xml version="1.0" ?>
                                 <CSL>
                                   <Command>getNTP</Command>
                                   <ntp enable="true" ip1="207.46.130.100"</pre>
                                       ip2="pool.ntp.org"
                                       mode="Saturday"
                                       time="00:00" />
```

5 0	assism id dasim assism ids 0.		
58.	session_id= <login_session_id>&</login_session_id>	Configure NTP server.	
	command=setNTP	e.g.	
	&ip1= <i>ip1</i>	http://192.168.25.160/API?session_id=12AC12DE&co	
	&ip2= <i>ip2</i>	mmand=setNTP&ip1=207.46.130.100&ip2=pool.ntp.or	
	&mode= <i>mode</i>	g&mode=Saturday&time=00:00&enable=true	
	&time= <i>time</i>		
	&enable= <i>enable</i>	Valid attributes:	
	[&immedidateUpdate=immediateUp	ip1, ip2 : NTP server address in form of dot-notation	
	date]	xxx.xxx.xxx or valid URL	
		mode: Every, Monday, Tuesday, Wednesday, Thursday,	
		Friday, Saturday, Sunday	
		time: in 24-hour form of hh:mm, e.g. 00:00, 23:59	
		enable : true,false	
		immedidateUpdate=true (false by default), synchronize the	
		date/time with time server immediately	
		result:	
		<pre><?xml version="1.0" ?></pre>	
		<pre></pre> <pre><</pre>	
		<command/> setNTP	
		<ack>OK:</ack>	
Vers	sion Management		
59.	session_id= <login_session_id>&</login_session_id>	Get version information of the Reader.	
	command=getReaderVersion	e.g.	
		session_id= <login_session_id>&command=getReaderVersion</login_session_id>	
		result:	
		<csl></csl>	
		<command/> getReaderVersion	
		<pre><readerversion <="" cs108_bluetooth_api_library="1.0.2" pre=""></readerversion></pre>	

Capture Point Management

60. session_id=<login_session_id>& command=setCapturePointName&c apturepoint_id=capturepoint_id&na me=name

Set Antenna Read Point Names.

e.g.1

http://192.168.25.160/API?session_id=75cf3f18&com_mand=setCapturePointName&capturepoint_id=Antenn_a1&name=Room1

e.g.2

http://192.168.25.160/API?session_id=75cf3f18&com mand=setCapturePointName&capturepoint_id=Antenn a2&name=Room2

e.g.3

http://192.168.25.160/API?session_id=75cf3f18&com_mand=setCapturePointName&capturepoint_id=Antenn_a3&name=Room3

e.g.4

http://192.168.25.160/API?session_id=75cf3f18&com mand=setCapturePointName&capturepoint_id=Antenn a4&name=Room4

Valid attributes:

capturepoint_id: Antenna1, Antenna2 Antenna16

61.	session_id= <login_session_id>&</login_session_id>	Get Capture Point Name (Antenna Name).				
	command=getCapturePointName					
		e.g.				
		http://192.168.25.160/API?session_id=75cf3f18&com				
		mand=getCapturePointName				
		xml version="1.0" ? <csl></csl>				
		<command/> getCapturePointName				
		<capturepoint <="" id="Antenna1" td=""></capturepoint>				
		name="Capture Point 1"				
		selected="true" />				
		<capturepoint <="" id="Antenna2" td=""></capturepoint>				
		name="Capture Point 2"				
		selected="true" />				
		<capturepoint <="" id="Antenna3" td=""></capturepoint>				
		name="Capture Point 3"				
		selected="true" />				
		<capturepoint <="" id="Antenna4" td=""></capturepoint>				
		name="Capture Point 4"				
		selected="true" />				
		<capturepoint <="" id="Antenna5" td=""></capturepoint>				
		name="Capture Point 5"				
		selected="true" />				
		<capturepoint <="" id="Antenna6" td=""></capturepoint>				
		name="Capture Point 6"				
		selected="true" />				
		<capturepoint <="" id="Antenna7" td=""></capturepoint>				
		name="Capture Point 7"				
		selected="true" />				
		<capturepoint <="" id="Antenna8" td=""></capturepoint>				
		name="Capture Point 8"				
		selected="true" />				
		<capturepoint <="" id="Antenna9" td=""></capturepoint>				
		name="Capture Point 9"				
		selected="true" />				
		<pre><capturepoint "conture="" 10"<="" id="Antenna10" page="" point="" pre=""></capturepoint></pre>				
		name="Capture Point 10"				
		selected="true" /> <capturepoint <="" id="Antenna11" td=""></capturepoint>				
		name="Capture Point 11"				
		selected="true" />				
		<pre><capturepoint <="" id="Antenna12" pre=""></capturepoint></pre>				
		name="Capture Point 12"				
		selected="true" />				
		<pre><capturepoint <="" id="Antenna13" pre=""></capturepoint></pre>				
		name="Capture Point 13"				
		selected="true" />				

```
<capturepoint id="Antenna14"</pre>
      name="Capture Point 14"
      selected="true" />
  <capturepoint id="Antenna15"</pre>
      name="Capture Point 15"
      selected="true" />
 <capturepoint id="Antenna16"</pre>
      name="Capture Point 16"
      selected="true" />
</CSL>
```

Cloud Server Management

62. | session_id=<login_session_id>& Set host notification url and port to be communicated with. command=setServerID&

server_id=server_id& e.g.1 Example CSL Demo Cloud Server desc=desc&

http://192.168.25.160/API?session_id=a33219dc&c type=type& ommand=setServerID&server_id=Example CSL server_ip=server_ip Demo Cloud Server&desc=Demo Http Cloud

[&server_port=server_port]

Server&type=HTTP&server ip=https://democloud.c [&client_id=client_id] onvergence.com.hk:29090/WebServiceRESTs/1.0/re [&username = *username*]

g/create-update-delete/update-entity/tagdata [&password=password]

[&enable ssl=enable ssl] e.g.2 Example TCP Server

[&ssl_version=ssl_version] http://192.168.25.160/API?session id=a33219 [&two_way_authentication=two_wa dc&command=setServerID&server_id=Demo

TCP Server&desc=Demo TCP y_authentication]

Server&type=TCP&server ip=192.168.25.100& [&topic=topic]

server port=9090

e.g.2 Example MQTT Server [&qos = qos]

[&clean_session=clean_session]

http://192.168.25.160/API?session id=a33219 dc&command=setServerID&server id=Demo MQTT Server&desc=Demo MQTT

Server&type=MQTT&server ip=

test.mosquitto.orq<u>&server_port=8883&enable_s</u> sl=true&ssl version=TLSv1.2&two way authen tication=false&topic=csl/tagdata&clean session

=true&gos=0

		Valid attributes:		
		type:	HTTP, TCP, MQTT	
		server_port :	required if type is TCP or MQTT	
		client_id:	optional depends on MQTT server	
		username :	optional depends on MQTT server	
		password :	optional depends on MQTT server	
		enable_ssl :	true, false, required if type is MQTT	
		ssl_version:	TLSv1.2, TLSv1.1, TLSv1, SSLv3, SSLv2,	
			required if enable_ssl is true	
		two_way_authe	entication: true, false, required if type is	
			enable_ssl is true	
		clean_session:	true, false, required if type is MQTT	
		qos :	0 (at most once), 1 (at least once), 2 (exactly	
			once), required if type is MQTT	
		result:		
		xml version</td <td>n="1.0" ?></td>	n="1.0" ?>	
		<command/> setServerID		
		<ack>OK:</ack>		
		C/CSL/		
63.	session_id= <login_session_id>&</login_session_id>	Modify host notification url and port to be		
	command= <i>modServerID</i> &	communicated	d with.	
	server_id=server_id&			
	desc=desc&	Valid attributes	:	
	type=type&	type:	HTTP, TCP, MQTT	
	server_ip= <i>server_ip</i>	server_port :	required if type is TCP	
	[&server_port=server_port]	client_id:	optional depends on MQTT server	
	[&client_id=client_id]	username:	optional depends on MQTT server	
	[&username = username]	password :	optional depends on MQTT server	
	[&password=password]	enable_ssl :	true, false, required if type is MQTT	
	[&enable_ssl=enable_ssl]	ssl_version:	TLSv1.2, TLSv1.1, TLSv1, SSLv3, SSLv2,	
	[&ssl_version=ssl_version]		required if enable_ssl is true	
	[&two_way_authentication=two_wa	two_way_authe	entication: true, false, required if type is	

```
y_authentication]
                                                          enable ssl is true
   [&topic=topic]
                                   clean_session: true, false, required if type is MQTT
    [&clean_session=clean_session]
                                                 0 (at most once), 1 (at least once), 2 (exactly
                                   qos:
   [\&qos = qos]
                                                 once), required if type is MQTT
                                    result:
                                    <?xml version="1.0" ?>
                                    <CSL>
                                      <Command>modServerID</Command>
                                      <Ack>OK:</Ack>
                                    </CSL>
64. session id=<login session id>&
                                   Remove server from the server list.
   command=delServerID&
    server id= server id
                                   e.g.
                                    session_id=<login_session_id>&command=delServerID&serv
                                   er_id=DemoServer
                                   result:
                                    <?xml version="1.0" ?>
                                      <Command>delServerID</Command>
                                      <Ack>OK:</Ack>
                                    </CSL>
65. session id=<login session id>&
                                   List server table.
    command=listServer
                                   e.g.
                                    session_id=<login_session_id>&command=listServer
                                   result:
                                    <?xml version="1.0" ?>
                                    <CSL>
                                      <Command>listServer</Command>
                                      <ServerList>
                                        <Server desc="Demo Http Cloud Server"</pre>
                                             server id="Examp CSL Demo Cloud Server"
                                             server_ip="https://democloud.convergen
                                             ce.com.hk:29090/WebServiceRESTs/1.0
                                             /req/create-update-delete/update-entit
                                             y/tagdata"
```

```
server_port=""
                                              type="HTTP" />
                                         <Server desc="Demo TCP Server"</pre>
                                              server_id="Demo TCP Server"
                                              server ip="192.168.25.100"
                                              server port="9090"
                                              type="TCP" />
                                       </ServerList>
                                     </CSL>
                                    Upload the SSL certificate of the specified server. The
66. session id=<login session id>&
                                    certificate must be in PEM format.
   command=setServerCertificate&
   server_id=server_id&
    serverCertFile=serverCertFile
                                    Here below is an example showing how to upload the SSL
                                    certificate via HTTP POST protocol written in C# (printed in
                                    blue color).
                                    HttpClient client = new HttpClient();
                                     var stream = new
                                    FileStream("C:\\temp\\certificate.pem",
                                    FileMode.Open);
                                    var content = new StreamContent(stream);
                                    var requestUri =
                                     "http://192.168.25.160/API?session_id=a33219dc&co
                                    mmand=setServerCertificate&server_id=Demo_MQTT
                                    Server&serverCertFile=certificate.pem";
                                    var response = await client.PostAsync(requestUri,
                                    content);
                                    result:
                                     <?xml version="1.0" ?>
                                       <Command>setServerCertificate</Command>
                                       <Ack>OK:</Ack>
                                     </CSL>
67. | session_id=<login_session_id>&
                                    Remove the SSL certificate of the specified server.
   command=delServerCertificate&
    server_id=server_id
                                    e.g.
```

		session_id= <login_session_id>&command=delServerCertificat</login_session_id>	
		e&server_id=Demo MQTT Server	
		result:	
		xml version="1.0" ?	
		<csl> <command/>delServerCertificate</csl>	
		<ack>OK:</ack>	
		\/CSL>	
68.	session_id= <login_session_id>&</login_session_id>	Add data format of packet to be sent to server.	
	command=addDataFormat&		
	data_format_id=data_format_id&	e.g.	
	desc=desc&	session_id= <login_session_id>&command=addDataFormat&d</login_session_id>	
	format=format	ata_format_id=ExampleDataFormat&desc=Example Data	
	&field{m}=field	Format&format=JSON&field1=RFIDReaderName&label1=rfi	
	&label{m}=label	dReaderName&field2=EthernetMACAddress&label2=ethernet	
	[&tagDataField{n}=tagDataField	MACAddress&field3=NumberOfTags&label3=numberOfTags	
	&tagDataLabel{n}=tagDataLabel]	&field4=TagDataList&label4=tags&tagDataField1=EPC&tag	
		DataLabel1=epc&tagDataField2=AntennaPort&tagDataLabel2	
		=antennaPort&tagDataField3=RSSI&tagDataLabel3=rssi&tag	
		DataField4=TimeOfRead&tagDataLabel4=time	
		Valid attributes:	
		format : JSON, XML, CSV	
		field: SequenceNumber,	
		NumberOfTags,	
		TagDataList,	
		RFIDReaderName,	
		RFIDReaderSerialNumber,	
		RFIDReaderInternalSerialNumber,	
		EthernetMACAddress,	
		WiFiMACAddress,	
		EthernetMACAddressWithColon,	

WiFiMACAddressWithColon, HeartBeatFlag, PowerUpFlag, ReaderErrorFlag, ReaderErrorCode, ReaderErrorDescription, ReaderErrorAntennaPort, ReaderErrorReflectedPower, ReaderErrorReflectedPowerThreshold, TimeOfHeartBeat, TimeOfPowerUp, TimeOfReaderError, TimeStampOfHeartBeat, TimeStampOfPowerUp, TimeStampOfReaderError, TimeZone tagDataField (effective only if field TagDataList exists): PC, EPC, TidBank, UserBank, TimeOfRead, TimeStampOfRead, TimeZone, AntennaPort, AntennaPort_Number, RSSI, RSSI_Number, Frequency, Phase, EventId, HeartBeatFlag, PowerUpFlag, ReaderErrorFlag,

		ReaderErrorCode,	
		ReaderErrorDescription,	
		ReaderErrorAntennaPort,	
		ReaderErrorReflectedPower,	
		ReaderErrorReflectedPowerThrehold,	
		TimeOfHeartBeat,	
		TimeOfPowerUp,	
		TimeOfReaderError,	
		TimeStampOfHeartBeat,	
		TimeStampOfPowerUp,	
		TimeStampOfReaderError	
		result:	
		xml version="1.0" ?	
		<csl> <command/>addDataFormat</csl>	
		<ack>OK:</ack>	
69.	session_id= <login_session_id>&</login_session_id>	Modify data format of packet to be sent to server.	
	command=modDataFormat&		
	data_format_id=data_format_id&	e.g.	
	[desc=desc&]	session_id= <login_session_id>&command=modDataFormat&</login_session_id>	
	format=format	data_format_id=ExampleDataFormat&desc=Example Data	
	&field{m}=field	Format&format=XML&field1=RFIDReaderName&label1=rfi	
	&label{m}=label	dReaderName&field2=EthernetMACAddress&label2=ethernet	
	[&tagDataField{n}=tagDataField	MACAddress&field3=NumberOfTags&label3=numberOfTags	
	&tagDataLabel{n}=tagDataLabel]	&field4=TagDataList&label4=tags&tagDataField1=EPC&tag	
		DataLabel1=epc&tagDataField2=AntennaPort&tagDataLabel2	
		=antennaPort&tagDataField3=RSSI&tagDataLabel3=rssi&tag	
		=antennaPort&tagDataField3=RSSI&tagDataLabel3=rssi&tag DataField4=TimeStampOfRead&tagDataLabel4=timeStamp	
		DataField4=TimeStampOfRead&tagDataLabel4=timeStamp	

NumberOfTags,

TagDataList,

RFIDReaderName,

RFIDReaderSerialNumber,

RFIDReaderInternalSerialNumber,

EthernetMACAddress,

WiFiMACAddress,

EthernetMACAddressWithColon,

WiFiMACAddressWithColon,

HeartBeatFlag,

PowerUpFlag,

ReaderErrorFlag,

ReaderErrorCode,

ReaderErrorDescription,

ReaderErrorAntennaPort,

ReaderErrorReflectedPower,

ReaderErrorReflectedPowerThreshold,

TimeOfHeartBeat,

TimeOfPowerUp,

TimeOfReaderError,

TimeStampOfHeartBeat,

TimeStampOfPowerUp,

TimeStampOfReaderError,

TimeZone

tagDataField (effective only if field TagDataList exists):

PC,

EPC,

TidBank,

UserBank,

TimeOfRead,

TimeStampOfRead,

TimeZone,

AntennaPort,

AntennaPort_Number,

		RSSI,	
		RSSI_Number,	
		Frequency,	
		Phase,	
		EventId,	
		HeartBeatFlag,	
		PowerUpFlag,	
		ReaderErrorFlag,	
		ReaderErrorCode,	
		ReaderErrorDescription,	
		ReaderErrorAntennaPort,	
		ReaderErrorReflectedPower,	
		Reader Error Reflected Power Threhold,	
		TimeOfHeartBeat,	
		TimeOfPowerUp,	
		TimeOfReaderError,	
		TimeStampOfHeartBeat,	
		TimeStampOfPowerUp,	
		TimeStampOfReaderError	
		result:	
		xml version="1.0" ? <csl></csl>	
		<command/> modDataFormat	
		<ack>OK:</ack>	
		7 552	
70.	session_id= <login_session_id>&</login_session_id>	Remove data format.	
	command=delDataFormat&		
	data_format_id=data_format_id	e.g.	
		session_id= <login_session_id>&command=delDataFormat&d</login_session_id>	
		ata_format_id=ExampleDataFormat	
		result:	
		xml version="1.0" ? <csl></csl>	

```
<Command>delDataFormat</Command>
                                   <Ack>OK:</Ack>
                                  </CSL>
71. session_id=<login_session_id>&
                                 List data format.
   command=listDataFormat
                                 e.g.
                                 session_id=<login_session_id>&command=listDataFormat
                                 result:
                                  <?xml version="1.0" ?>
                                  <CSL>
                                    <Command>listDataFormat</Command>
                                    <DataFormatList>
                                     <dataFormat
                                          data_format_id="ExampleDataFormat"
                                          desc="Example Data Format"
                                          field1="RFIDReaderName"
                                          field2="EthernetMACAddress"
                                          field3="NumberOfTags"
                                          field4="TagDataList"
                                          format="XML"
                                          label1="rfidReaderName"
                                          label2="ethernetMACAddress"
                                          label3="numberOfTags"
                                          label4="tags"
                                          tagDataField1="EPC"
                                          tagDataField2="AntennaPort"
                                          tagDataField3="RSSI"
                                          tagDataField4="TimeStampOfRead"
                                          tagDataLabel1="epc"
                                          tagDataLabel2="antennaPort"
                                          tagDataLabel3="rssi"
                                          tagDataLabel1="timeStamp" />
                                   </DataFormatList>
                                  </CSL>
I/O Management
72. | session_id=<login_session_id>&
                                 Set the output port port to the logic value oper_logic.
   command=runIO_output&
   port=port&
                                 e.g.
   oper_logic=oper_logic
                                 http://192.168.25.160/API?session_id=a33219dc&com
```

```
mand=runIO output&port=1&oper logic=1
                                   result:
                                    <?xml version="1.0" ?>
                                    <CSL>
                                      <Command>runIO_output</Command>
                                      <Ack>OK:</Ack>
                                    </CSL>
                                    Valid attributes:
                                   port
                                            : 1,2,3,4
                                   oper_logic: 0,1
73. |session_id=<login_session_id>&
                                   Set the output port to the logic value logic.
   command=runIO_output8bits&
   logic=logic
                                   e.g.
                                   http://192.168.25.245/API?session_id=f9125ad4&com
                                   mand=runIO output8bits&logic=0F
                                   result:
                                    <?xml version="1.0" ?>
                                    <CSL>
                                      <Command>runIO_output8bits</Command>
                                      <Ack>OK:</Ack>
                                    </CSL>
                                    Valid attributes:
                                   logic: 2 hex digits, i.e. 00 - 0F
74. |session_id=<login_session_id>&
                                   Get input status from I/O ports.
   command=runIO_input
                                   e.g.1 Synchronized mode
                                   http://192.168.25.160/API?session_id=a33219dc&com
                                   mand=runIO input
```

```
result:
                                       <?xml version="1.0" ?>
                                       <CSL>
                                         <Command>runIO_input</Command>
                                         <Input input_logic_list="0,1,0,0"</pre>
                                              port_list="1,2,3,4" />
                                       </CSL>
                                      i.e.
                                      port1 : logic '0'
                                      port2 : logic '1'
                                      port3: logic '0'
                                      port4: logic '0'
                                       The 0 or 1 in the input_logic_list represents the corresponding
                                      logic of port number (port 1-4) in the port_list.
75. | session_id=<login_session_id>&
                                      Set or reset the output port port according to the logic
    command=directIOOutput&
                                       oper_logic (without login).
    port=port&
    oper_logic=oper_logic&
                                      e.g.
                                      http://192.168.25.160/API?command=directIOOutput
    username=username&
                                       <u>&port=1&oper_logic=1&username=root&password=csl</u>
    passwor=password
                                      result:
                                       <?xml version="1.0" ?>
                                       <CSL>
                                         <Command>directIOOutput</Command>
                                         <Ack>OK:</Ack>
                                       </CSL>
                                       Valid attributes:
                                       port
                                                : 1,2,3,4
                                       oper_logic: 0,1
76. | session_id=<login_session_id>&
                                      Set or reset the output port port according to the logic
    command=directIOOutput8bits&
                                       oper_logic (without login).
```

	logic=logic&		
	username=username&	e.g.	
	passwor=password	http://192.168.25.245/API?command=directIOOutput	
		8bits&logic=0F&username=root&password=csl	
		result:	
		xml version="1.0" ?	
		<csl></csl>	
		<command/> directIOOutput8bits <ack>OK:</ack>	
		Valid attributes :	
		logic : 2 hex digits, i.e. 00 - 0F	
77.	session_id= <login_session_id>&</login_session_id>	Get input status from I/O ports (without login).	
	command=directIOInput&		
	username=username&	e.g.1 Synchronized mode	
	passwor=password	http://192.168.25.160/API?command=directIOInput&u	
		sername=root&password=csl	
		result:	
		xml version="1.0" ?	
		<csl> <command/>directIOInput</csl>	
		<input <="" input_logic_list="0,1,0,0" th=""/>	
		port_list=" 1,2,3,4 " />	
		Valid attributes:	
		The 0 or 1 in the input_logic_list represents the corresponding	
		logic of port number (port 1-4) in the port_list.	
Eve	nts Management		
78	session_id= <login_session_id>&</login_session_id>	Create a triggering logic in <triggeringlogic> table.</triggeringlogic>	
. 5.		The state of the s	

command=addTriggeringLogic&

logic_id=logic_id&

desc=desc&

mode=*mode*

[&logic=logic]

[&state_mode=state_mode]

[&capturePoint=capturePoint]

[&referenceTagId=ref_tag]

e.g.1 This mode is used for "InventoryEnablingTrigger" or

"Trigger Logic" in Event definition

http://192.168.25.160/API?session_id=a33219dc&com

mand=addTriggeringLogic&logic_id=DemoTrigger&desc

<u> =Demo Trigger&mode=Read Any Tags (any ID, 1 trigger</u>

<u>per tag)</u>

e.g.2

http://192.168.25.160/API?session_id=a33219dc&com

mand=addTriggeringLogic&logic_id=Sensor1&desc=Se

snor 1&mode=Input Sensor

State&logic=Sensor1:0&state mode=CHANGE

e.g.3 This mode is used for "InventoryDisablingTrigger" with

reference tag in Event definition

http://192.168.25.160/API?session_id=a33219dc&com

mand=addTriggeringLogic&logic_id=NoTagAndStop&de

sc=Stop Inventory if no tag read more than 2

seconds&mode=No Tag Read in Specified Time

Span&logic=2000&referenceTagId=012345678901234

56789ABCD

e.g.4

http://192.168.25.160/API?session_id=a33219dc&com

<u>mand=addTriggeringLogic&logic_id=TagTest&desc=Tag</u>

Read test&mode=Tag Group Filtering&logic=TagGroup1

Valid attributes:

mode: Read Any Tags (any ID, 1 trigger per tag),

Input Sensor State,

No Tag Read in Specified Time Span,

Trigger in Tag Group,

Trigger in Tag Database,

Trigger if RSSI larger than or equal to,
Trigger if Moisture is larger than or equal to,
Trigger if Moisture is less than or equal to,
Trigger if Temperature is larger than or equal to,
Trigger if Temperature is less than or equal to,
Specified Time Span elapsed,

If mode==Input Sensor State

logic=Sensor and input level in form of 'Sensor[n]:[0,1]' where n=1,2,3,4. eg. Sensor1:0 ==> Sensor1 with input in high level, Sensor2:1 ==> Sensor2 with input in low level

If mode==No Tag Read in Specified Time Span logic=<time span in which no tag read, unit:ms>, refer to e.g.3 referenceTagId=<EPC>, EPC of the reference tag, which is ignore in counting the time

If mode==Trigger in Tag Group logic=<Tag Group>, refer to e.g.4

If mode==Trigger if RSSI larger than or equal to,

Trigger if Moisture is larger than or equal to,

Trigger if Moisture is less than or equal to,

Trigger if Temperature is larger than or equal to,

Trigger if Temperature is less than or equal to
logic=<threshold value>

state_mode : CHANGE = sensor input is changed to the specified logic

LEVEL = sensor input meets the specified logic

capturePoint : 1 - 16, any combinations with comma

			separated
		result:	
			version="1.0" ?>
		<csl></csl>	nmand>addTriggeringLogic
		<ack>OK:</ack>	
79.	session_id= <login_session_id>&</login_session_id>	Modify a	an existing triggering logic in <triggeringlogic> table</triggeringlogic>
	command=modTriggeringLogic&	by logic_	_id.
	logic_id=logic_id		
	[&desc=desc]	e.g.1 mo	dify capture point
	[&mode=mode]	http://1	92.168.25.160/API?session_id=a33219dc&com
	[&logic=logic]	mand=i	modTriggeringLogic&logic_id=DemoTrigger∩
	[&state_mode=state_mode]	turePoir	nt=1,3
	[&capture_point=capture_point]		
	[&referenceTagId=ref_tag]	e.g.2 modify desc	
		http://192.168.25.160/API?session_id=a33219dc&com	
		mand=modTriggeringLogic&logic_id=DemoTrigger&des	
		<u>c=ModifiedDemoTrigger</u>	
		Valid att	ributes :
		mode :	Read Any Tags (any ID, 1 trigger per tag),
		mode.	Input Sensor State,
			No Tag Read in Specified Time Span,
			Trigger in Tag Group,
			Trigger in Tag Database,
			Trigger if RSSI larger than or equal to,
			Trigger if Moisture is larger than or equal to,
			Trigger if Moisture is less than or equal to,
			Trigger if Temperature is larger than or equal to,
			Trigger if Temperature is less than or equal to,
			Specified Time Span elapsed,
		If mode=	=Input Sensor State
			•

```
logic=Sensor and input level in form of 'Sensor[n]:[0,1]' where
                                        n=1,2,3,4. eg. Sensor1:0 ==> Sensor1 with input in high level,
                                        Sensor2:1 ==> Sensor2 with input in low level
                                        If mode==No Tag Read in Specified Time Span
                                        logic=<time span in which no tag read, unit:ms>, refer to e.g.3
                                        referenceTagId=<EPC>, EPC of the reference tag, which is
                                        ignore in counting the time
                                        If mode==Trigger in Tag Group
                                        logic=<Tag Group>, refer to e.g.5
                                        If mode==Trigger if RSSI larger than or equal to,
                                                  Trigger if Moisture is larger than or equal to,
                                                  Trigger if Moisture is less than or equal to,
                                                  Trigger if Temperature is larger than or equal to,
                                                  Trigger if Temperature is less than or equal to
                                        logic=<threshold value>
                                        state_mode:
                                                       CHANGE = sensor input is changed to the
                                                                  specified logic
                                                       LEVEL = sensor input meets the specified
                                                                  logic
                                        capturePoint : 1 - 16, any combinations with comma
                                                       separated
                                        result:
                                        <?xml version="1.0" ?>
                                        <CSL>
                                          <Command>modTriggeringLogic</Command>
                                          <Ack>OK:</Ack>
                                        </CSL>
80. | session_id=<login_session_id>&
                                        Remove a triggering logic from the <TriggeringLogic> table.
```

```
command=delTriggeringLogic&
   logic_id=logic_id
                                   e.g.
                                   session_id=<login_session_id>&command=delTriggeringLogi
                                   c&logic _id=logic1
                                   result:
                                   <?xml version="1.0" ?>
                                   <CSL>
                                     <Command>delTriggeringLogic</Command>
                                     <Ack>OK:</Ack>
                                   </CSL>
81. session_id=<login_session_id>&
                                   List Triggering Logic table.
   command=listTriggeringLogic
                                   e.g.
                                   session_id=<login_session_id>&command=listTriggeringLogi
                                   result:
                                   <?xml version="1.0" ?>
                                     <Command>listTriggeringLogic</Command>
                                     <TriggeringLogic>
                                       logic capture_point="1234"
                                            desc="Read Burn-in Trigger Logic"
                                            logic=""
                                            logic_id="ReadBurninTrigger"
                                            mode="Read Any Tags (any ID, 1 trigger
                                            per tag)"
                                            referenceTagId=""
                                            state_mode="" />
                                       <logic capture_point="1234"</pre>
                                            desc="Read Burn-in Disabling Trigger
                                            Logic"
                                            logic="15000"
                                            logic_id="ReadBurninDisableTrigger"
                                            mode="No Tag Read in Specified Time
                                            Span"
                                            referenceTagId="00000002009050509522
                                            7234"
                                            state_mode=""/>
                                     </TriggeringLogic>
                                   </CSL>
82. |session_id=<login_session_id>&
                                   Create a resultant action in <ResultantAction> table.
```

command=addResultantAction&

action_id=action_id&

e.g. http://192.168.25.160/API?session_id=a33219dc&com

desc=desc

mand=addResultantAction&action_id=DemoAction&des

[&condition=condition

c=Demo%20Action&action mode=Batch%20Alert%20t

&condition_logic=condition_logic]

&action_mode=action_mode

o%20Server&server_id=DemoServer

ro :1 :7

[&server_id=server_id

&pre_action_wait=pre_action_wait

&post_action_delay=post_action_de

lay

&action=action

&pulse_logic=pulse_logic

&pulse_mode=pulse_mode

&pulse_width=pulse_width

&dutycycle=*dutycycle*

&duration=duration

&transport=transport

&data_format_id=data_format_id

&display_format_id=display_format

_id

&display_time_factor_type=*display*

_time_factor_type

&display_time_factor=display_time | server_id :

_factor]

Valid attributes:

condition: None, Input Sensor State

condition_logic: If condition is Input Sensor State, it

represents Sensor and input level in

form of 'Sensor[n]:[0,1]' where

n=1,2,3,4. eg. Sensor1:0 ==> Sensor1

with input in high level, Sensor2:1

==> Sensor2 with input in low level

action_mode: Do Nothing (Only Show on Screen),

Batch Alert to Server, Instant Alert to Server,

Low Latency Alert to Server,

Output Port,

Display Tag Database Record,

Display Tag Group Record,

server_id : <Cloud Server>

pre_acton_wait : unit = ms
post acton delay : unit = ms

•

action: Port[n]:[0,1,Pulse] where n=1,2,3,4;

0 ==> Open switch, 1 ==> Close switch

pulse_logic : Positive ==> Open, Close, Open

Negative ==> Close, Open, Close

(for action=Pulse only)

pulse_mode : One Shot Pulse, Pulse Train

pulse_width : unit = ms

dutycycle: unit = % (for Pulse Train only)
duration: unit = ms (for Pulse Train only)

		transport : TCP, HTTP POST, MQTT
		display_time_factor_type : Additive, Multiplicative
		Note:
		If action_mode is Display Tag Database Record or Display Tag
		Group Record, the display time is determined by
		display_time_factor_type and display_time_factor.
		e.g.1 display_time_factor_type = Additive
		display time = Tag Duplicate Elimination Window +
		display_time_factor (ms)
		e.g.2 display_time_factor_type = Multiplicative
		display time = Tag Duplicate Elimination Window ×
		display_time_factor (ms)
		If display_time_factor = 0, record shown on the page forever
		until next record comes in.
		result:
		xml version="1.0" ?
		<csl> <command/>addResultantAction</csl>
		<ack>OK:</ack>
		\/CSL>
83.	session_id= <login_session_id>&</login_session_id>	Modify an existing resultant action in <resultantaction> table</resultantaction>
	command=modResultantAction&	by action_id.
	action_id=actiont_id	
	[&desc=desc]	e.g.1
	[&condition=condition]	http://192.168.25.160/API?session_id=a33219dc&com
	[&condition_logic=condition_logic]	mand=modResultantAction&action_id=DemoAction&de
	[&action_mode=action_mode]	sc=Demo%20Action
	[&server_id=server_id]	
	[⪯_action_wait=pre_action_wait	e.g.2
]	http://192.168.25.160/API?session_id=a33219dc&com
	[&post_action_delay=post_action_d	mand=modResultantAction&action id=DemoAction&se
	elay]	rver id=DemoServer

[&action=action] Valid attributes: None, Input Sensor State [&pulse_logic=pulse_logic] condition: [&pulse_mode=pulse_mode] condition_logic: If condition is Input Sensor State, it [&pulse_width=pulse_width] represents Sensor and input level in [&dutycycle=*dutycycle*] form of 'Sensor[n]:[0,1]' where [&duration=duration] n=1,2,3,4. eg. Sensor1:0 ==> Sensor1 [&transport=transport] with input in high level, Sensor2:1 [&data_format_id=data_format_id] ==> Sensor2 with input in low level [&display_format_id=display_forma action mode: Do Nothing (Only Show on Screen), t_id Batch Alert to Server, [&display_time_factor_type=display Instant Alert to Server, _time_factor_type] Low Latency Alert to Server, [&display_time_factor=display_time Output Port, *factor*] Display Tag Database Record, Display Tag Group Record, <Cloud Server> server_id: unit = mspre_acton_wait: post_acton_delay: unit = msPort[n]:[0,1,Pulse] where n=1,2,3,4; action: 0 = >Open switch, 1 = >Close switch pulse_logic: Positive ==> Open, Close, Open Negative ==> Close, Open, Close (for action=Pulse only) pulse_mode: One Shot Pulse, Pulse Train pulse_width: unit = msdutycycle: unit = % (for Pulse Train only) duration: unit = ms (for Pulse Train only) transport: TCP, HTTP POST, MQTT display_time_factor_type : Additive, Multiplicative Note: If action_mode is Display Tag Database Record or Display Tag Group Record, the display time is determined by

```
display_time_factor_type and display_time_factor.
                                      e.g.1 display_time_factor_type = Additive
                                           display time = Tag Duplicate Elimination Window +
                                                          display_time_factor (ms)
                                      e.g.2 display time factor type = Multiplicative
                                           display time = Tag Duplicate Elimination Window \times
                                                          display_time_factor (ms)
                                      If display_time_factor = 0, record shown on the page forever
                                      until next record comes in.
                                      result:
                                      <?xml version="1.0" ?>
                                      <CSL>
                                        <Command>modResultantAction</Command>
                                        <Ack>OK:</Ack>
                                      </CSL>
84. | session_id=<login_session_id>&
                                      Remove an events action from <ResultantActionList> table.
    command=delResultantAction&
    action_id=action_id
                                      e.g.
                                      session id=<login session id>&command=delResultantActio
                                      n&action _id=DemoAction
                                      result:
                                      <?xml version="1.0" ?>
                                      <CSL>
                                        <Command>delResultantAction</Command>
                                        <Ack>OK:</Ack>
                                      </CSL>
                                      List all events action from <ResultantActionList> table.
85. | session_id=<login_session_id>&
    command=listResultantAction
                                      e.g.
                                      session_id=<login_session_id>&command=listResultantActio
                                      result:
                                      <?xml version="1.0" ?>
```

```
<CSL>
                                      <Command>listResultantAction</Command>
                                      <ResultantActionList>
                                         <resultantaction action=""
                                             action id="DemoAction"
                                             action mode="Instant Alert to Server"
                                             condition="None"
                                             condition logic=" "
                                             data_format_id="Example Tag Format"
                                             desc="Demo Action"
                                             display time factor="0"
                                             display_time_factor_type=""
                                             duration="0"
                                             dutycycle="0"
                                             post_action_delay="0"
                                             pre_action_wait="0"
                                             pulse_logic=""
                                             pulse_mode=""
                                             pulse_width="0"
                                             server_id="DemoServer"
                                             transport="HTTP POST" />
                                      </ResultantActionList>
                                    </CSL>
86. | session_id=<login_session_id>&
                                    Create event definition.
   command=addEvent&
                                    e.g. 1
   event_id=event_id&
                                    http://192.168.25.160/API?session_id=a33219dc&com
                                    mand=addEvent&event id=DemoEvent&desc=Demo%
   desc=desc&
                                    20Event&operProfile id=Default%20Profile&exclusivity
   operProfile id=operProfile id&
                                    =Non-exclusive&duplicateEliminationWindow=10000&a
   exclusivity=exclusivity&
                                    ntennaDifferentiation=false&triggering logic=DemoTrig
   duplicateEliminationWindow=
                                    ger&resultant action=DemoAction&enable=true
   duplicateEliminationWindow&
   antennaDifferentiation=
                                    Valid attributes:
   antennaDifferentiation&
                                    exclusivity:
                                                                Exclusive, Non-exclusive
   triggering_logic=triggering_logic&
                                    duplicateEliminationWindow: unit = ms
   resultant action=resultant action&
                                    antennaDifferentiation:
                                                                true, false
   enable=enable
                                    enable:
                                                                true, false
   [&inventoryEnablingTrigger=
                                    triggering logic:
                                                                < Read Any Tags (any ID, 1
   inventoryEnablingTrigger
                                                                trigger per tag)>,
   &inventoryDisablingTrigger=
                                                                <Trigger in Tag Group>,
   inventoryDisablingTrigger]
                                                                <Trigger in Tag Database>,
```

[&inventoryEnablingAction= < Trigger if RSSI larger than or equal to>, inventoryEnablingAction < Specified Time Span &inventoryDisablingAction= elapsed> inventoryDisablingAction] NONE, resultant action: <Do Nothing (Only Show on</pre> Screen)>, <Batch Alert to Server>, <Instant Alert to Server>, <Low Latency Alert to Server>, <Output Port>, < Display Tag Database Record>, <Display Tag Group Record>, inventoryEnablingTrigger: Always On, <Input Sensor State> inventoryDisablingTrigger: Never Stop, <Input Sensor State>, <No Tag Read in Specified Time Span>, < Specified Time Span elapsed> inventoryEnablingAction: NONE, <Output Port> inventoryDisablingAction: NONE, <Output Port> The valid resultant action operation can be used in the inventoryEnablingAction, inventoryDisablingAction and resultant_action attributes are as follows: AND, THEN

		result: xml version="1.0" ? <csl> <command/>addEvent<</csl>	/Command>	
		<ack>OK:</ack>		
87.	session_id= <login_session_id>&</login_session_id>	Modify event definition.		
	command=modEvent&	e.g. 1		
	event_id=event_id&	http://192.168.25.160/API3	?session id=a33219dc&com	
	desc=desc&	mand=modEvent&event_id	=DemoEvent&desc=Demo	
	operProfile_id=operProfile_id&	%20Event&operProfile id=[
	exclusivity= <i>exclusivity</i> &		tant action=DemoAction&e	
	duplicateEliminationWindow=	vent log=false&enable=tru	<u>e</u>	
	duplicateEliminationWindow&			
	antennaDifferentiation=	Valid attributes:		
	antennaDifferentiation&	exclusivity:	Exclusive, Non-exclusive	
	triggering_logic=triggering_logic&	duplicateEliminationWindow:		
	resultant_action=resultant_action&	antennaDifferentiation:	true, false	
	enable= <i>enable</i>	enable:	true, false	
	[&inventoryEnablingTrigger=	triggering_logic :	<read (any="" 1<="" any="" id,="" tags="" td=""></read>	
	inventoryEnablingTrigger]		trigger per tag)>,	
	[&inventoryDisablingTrigger=		<trigger group="" in="" tag="">,</trigger>	
	inventoryDisablingTrigger]		<trigger database="" in="" tag="">,</trigger>	
	[&inventoryEnablingAction=		<trigger if="" larger="" or<="" rssi="" td="" than=""></trigger>	
	inventoryEnablingAction		equal to>,	
	&inventoryDisablingAction=		<specified span<="" td="" time=""></specified>	
	inventoryDisablingAction]		elapsed>	
	invenioryDisablingAction	resultant_action:	NONE,	
			<pre><do (only="" nothing="" on<="" pre="" show=""></do></pre>	
			Screen)>,	
			<batch alert="" server="" to="">,</batch>	
			<instant alert="" server="" to="">,</instant>	
			<low alert="" latency="" td="" to<=""></low>	
			Server>,	
			<output port="">,</output>	

			<display database<="" tag="" th=""></display>
			Record>,
			<display group="" record="" tag="">,</display>
		inventoryEnablingTrigger:	Always On,
			<input sensor="" state=""/>
		inventoryDisablingTrigger:	Never Stop,
			<input sensor="" state=""/> ,
			<no in="" read="" specified<="" tag="" td=""></no>
			Time Span>,
			<specified span<="" td="" time=""></specified>
			elapsed>
		inventoryEnablingAction:	NONE,
			<output port=""></output>
		inventoryDisablingAction:	NONE,
			<output port=""></output>
		The valid resultant action operation can be used in the	
		inventoryEnablingAction, inventoryDisablingAction and	
		resultant_action attributes are as follows:	
		AND, THEN	
		result:	
		xml version="1.0" ? <csl></csl>	
88.	session_id= <login_session_id>&</login_session_id>	Enable/disable an event to be active/inactive.	
	command=enableEvent&		
	event_id=event_id&		
	enable= <i>enable</i>		
		Volid attributes :	
		Valid attributes :	

```
enable:
                                            true => enable an event to be active
                                            false => disable an event to be inactive
                                   result:
                                   <?xml version="1.0" ?>
                                     <Command>enableEvent</Command>
                                     <Ack>OK:</Ack>
                                   </CSL>
89. | session_id=<login_session_id>&
                                   Remove an event definition from the table.
   command=delEvent&
   event_id=event_id
                                   e.g.
                                   session_id=<login_session_id>&command=delEvent&event
                                   id=DemoEvent
                                   result:
                                   <?xml version="1.0" ?>
                                   <CSL>
                                     <Command>delEvent</Command>
                                     <Ack>OK:</Ack>
                                   </CSL>
                                   List Event definition table.
90. | session_id=<login_session_id>&
   command=listEvent
                                   e.g.
                                   session id=<login session id>&command=listEvent
                                   result:
                                   <?xml version="1.0" ?>
                                   <CSL>
                                     <Command>listEvent</Command>
                                     <EventMode mode="0" />
                                     <EventList>
                                        <event antennaDifferentiation="false"</pre>
                                            desc="Event Demo"
                                            duplicateEliminationWindow="10000"
                                            enable="true"
                                            event id="DemoEvent"
                                            exclusivity="Non-exclusive"
                                            inventoryDisablingAction="NONE"
                                            inventoryDisablingTrigger="Never Stop"
                                            inventoryEnablingAction="NONE"
                                            inventoryEnablingTrigger="Always On"
```

```
operProfile id="Default Profile"
                                            resultant_action="DemoAction"
                                            triggering_logic="DemoTrigger" />
                                        <event antennaDifferentiation="false"</pre>
                                            desc=""
                                            duplicateEliminationWindow="10000"
                                            enable="true"
                                            event_id="EventTest"
                                            exclusivity="Non-exclusive"
                                            inventoryDisablingAction="NONE"
                                            inventoryDisablingTrigger="Never Stop"
                                            inventoryEnablingAction="NONE"
                                            inventoryEnablingTrigger="Always On"
                                            operProfile id="Default Profile"
                                            resultant action="NONE"
                                            triggering logic="DemoTrigger" />
                                     </EventList>
                                    </CSL>
91. http://<ip>/importTagGroupCSV
                                   Import Tag Group from CSV file.
   session id=<login session id>&
                                   Here below is an example showing how to import CSV file via
    tagGroupFilename=tagGroupFilena
                                   HTTP POST protocol written in C# (printed in blue color).
    me&
   tagGroupContent=tagGroupContent
                                   HttpClient client = new HttpClient();
                                   var str =
                                   File.ReadAllText("C:\\temp\\DemoGroup.csv");
                                   var map = new Dictionary<string, string>
                                        { "session_id", "a33219dc" },
                                        { "tagGroupFilename", "DemoGroup.csv" },
                                        { "tagGroupContent", str }
                                   var content = new FormUrlEncodedContent(map);
                                   var requestUri =
                                    "http://192.168.25.160/importTagGroupCSV";
                                   var response = await client.PostAsync(requestUri,
                                   content);
                                   result:
                                    <?xml version="1.0" ?>
                                    <CSL>
                                     <Command>importTagGroupCSV</Command>
```

```
<Ack>OK:</Ack>
                                  </CSL>
                                  or
                                  <?xml version="1.0" ?>
                                  <CSL>
                                    <Command>importTagGroupCSV</Command>
                                    <Ack>Error: Tag Group already existed.
                                        (Remark: Tag Group = <filename>)</Ack>
                                  </CSL>
92. | session_id=<login_session_id>&
                                  Remove a tag group.
   command=delTagGroup&
                                 e.g.
   group_id=group_id
                                  session_id=<login_session_id>&command=delTagGroup&gro
                                  up_id=DemoGroup
                                  result:
                                  <?xml version="1.0" ?>
                                  <CSL>
                                    <Command>delTagGroup</Command>
                                    <Ack>OK:</Ack>
                                  </CSL>
93. | session_id=<login_session_id>&
                                 List tag group.
   command=listTagGroup
                                 e.g.
                                  session_id=<login_session_id>&command=listEvent
                                  result:
                                  <?xml version="1.0" ?>
                                    <Command>listTagGroup</Command>
                                    <TagGroupList>
                                      <tagGroup group_id="DemoGroup">
                                        <tag id="01234567890123456789ABCD" />
                                        <tag id="000000020090505095227234" />
                                      </tagGroup>
                                    </ TagGroupList>
                                  </CSL>
94. session id=<login session id>&
                                 Backup reader configuration.
   command=configurationBackup
                                 e.g.
                                  session_id=<login_session_id>&command=configurationBack
```

```
up
                                   result:
                                   <?xml version="1.0" ?>
                                     <Command>configurationBackup</Command>
                                     <Configuration>{configuration in Json
                                          format}</Configuration>
                                   </CSL>
95. session id=<login session id>&
                                   Backup reader configuration.
   command=configurationRestore&
                                   e.g.
   configuration=configuration
                                   session_id=<login_session_id>&command=configurationRest
                                   ore&configuration={configuration in Json format}
                                   Valid attributes:
                                   configuration: configuration of reader in Json format
                                   result:
                                   <?xml version="1.0" ?>
                                   <CSL>
                                      <Command>configurationRestore</Command>
                                     <Ack>OK: Please wait a moment... System
                                          restarting...</Ack>
                                   </CSL>
96. |session_id=<login_session_id>&
                                   Set database configuration.
   command=setDatabaseConfiguratio e.g.
   n&
                                   http://192.168.25.160/API?session_id=a33219dc&com
                                   mand=setDatabaseConfiguration&databasePath=%2Fr
   databasePath=databasePath
                                   un%2Fmedia%2Fmmcblk2p5%2Fmysql
                                   result:
                                   <?xml version="1.0" ?>
                                   <CSL>
                                     <Command>setDatabaseConfiguration
                                     </Command>
                                     <Ack>OK: </Ack>
                                   </CSL>
97. |session_id=<login_session_id>&
                                   Set database configuration.
```

	command=getDatabaseConfigurati	e.g.	
	on	session_id= <login_session_id>&command=getDatabaseConfi</login_session_id>	
		guration	
		result:	
		xml version="1.0" ?	
		<csl></csl>	
		<command/> getDatabaseConfiguration	
		<pre><database path="/run /modia /mmobils2nF /mysql"></database></pre>	
		<pre>path="/run/media/mmcblk2p5/mysql" /> </pre>	
98.	session_id= <login_session_id>&</login_session_id>	Add database.	
	command=addDatabase&	e.g.	
	databaseName=databaseName	http://192.168.25.160/API?session_id=a33219dc&com	
	[&field{n}=field	mand=addDatabase&databaseName=ProductDB&field1	
	&dataType{n}=datatype]	=EPC&dataType1=STRING&field2=ProductId&dataType	
		2=STRING&field3=ProductName&dataType3=STRING&	
		field4=ProductPrice&dataType4=NUMBER	
		Valid attributes:	
		dataType : STRING, NUMBER, IMAGE	
		Note:	
		A STRING dateType EPC field is always used as the primary	
		key field in the database. If the HTTP message does not	
		contain a EPC field, the system adds it.	
		result:	
		xml version="1.0" ?	
		<csl></csl>	
		<command/> addDatabase <ack>OK: </ack>	
99.	session_id= <login_session_id>&</login_session_id>	Modify database.	
	command= <i>modDatabase</i> &	e.g.	

databaseName=databaseName	http://192.168.25.160/API?session_id=a33219dc&com
[&field{n}=field	mand=modDatabase&databaseName=ProductDB&field
&dataType{n}=datatype]	1=EPC&dataType1=STRING&field2=ProductId&dataTyp
	e2=STRING&field3=ProductName&dataType3=STRING
	&field4=ProductImage&dataType4=IMAGE&field5=Prod
	uctPrice&dataType5=NUMBER
	Valid attributes :
	dataType : STRING, NUMBER, IMAGE
	Note:
	A STRING dateType EPC field is always used as the primary
	key field in the database. The field should not be modified.
	result:
	xml version="1.0" ?
	<csl> <command/>modDatabase</csl>
	<ack>OK: </ack>
100 session_id= <login_session_id>&</login_session_id>	Delete database.
command=delDatabase&	e.g.
databaseName=databaseName	session_id= <login_session_id>&command=delDatabase&data</login_session_id>
databaser varie—aarabaser varie	baseName=ProductDB
	buservaline—FroductibB
	result:
	xml version="1.0" ?
	<csl></csl>
	<command/> delDatabase <ack>OK: </ack>
101 session_id= <login_session_id>&</login_session_id>	List the field names and the data types of all databases.
command=listDatabase	e.g.
	session_id= <login_session_id>&command=listDatabase</login_session_id>

```
result:
                                  <?xml version="1.0" ?>
                                  <CSL>
                                    <Command>listDatabase</Command>
                                    <DatabaseList>
                                      <database dataType1="STRING"</pre>
                                          dataType2="STRING"
                                          dataType3="IMAGE"
                                          dataType4="NUMBER"
                                          databaseName="ProductDB"
                                          field1="ProductId"
                                          field2="ProductName"
                                          field3="ProductImage"
                                          field4="ProductPrice"
                                          keyDataType="STRING"
                                          keyField="EPC"/>
                                    </DatabaseList>
                                  </CSL>
102 session_id=<login_session_id>&
                                  Add tag record to the database.
   command=addTagDatabaseRecord
                                  e.g.
   &databaseName=databaseName
                                  http://192.168.25.160/API?session_id=a33219dc&com
                                  mand=addTagDatabaseRecord&databaseName=Produc
   &EPC=EPC
                                  tDB&EPC=01234567890123456789ABCD&ProductId=1
   [&fieldname=value]
                                  234&ProductName=Orange%20Juice&ProductPrice=1.
                                  Note:
                                  EPC field is always used as the primary key field in the
                                  database. It must be included in the addTagDatabaseRecord
                                  message and must not be null or empty.
                                  Use setTagDatabaseRecordImage command to set the image
                                  field.
                                  result:
                                  <?xml version="1.0" ?>
                                  <CSL>
                                    <Command>addTagDatabaseRecord/Command
                                    <Ack>OK: </Ack>
                                  </CSL>
```

103 session_id= <login_session_id>&</login_session_id>	Modify tag record in the database.		
command=modTagDatabaseRecord	e.g.		
&databaseName=databaseName	http://192.168.25.160/API?session_id=a33219dc&com		
&EPC= <i>EPC</i>	mand=modTagDatabaseRecord&databaseName=Produ		
[&fieldname=value]	ctDB&EPC=01234567890123456789ABCD&ProductId=		
	1234&ProductName=Orange%20Juice&ProductPrice=1		
	<u>.75</u>		
	Note:		
	EPC field is always used as the primary key field in the		
	database. It must be included in the modTagDatabaseRecord		
	message and must not be null or empty.		
	Use setTagDatabaseRecordImage command to set the image		
	field.		
	result:		
	xml version="1.0" ?		
	<csl> <command/>modTagDatabaseRecord</csl>		
	>		
	<ack>OK: </ack> 		
104 session_id= <login_session_id>&</login_session_id>	Delete tag record from the database.		
command=delTagDatabaseRecord	e.g.		
&databaseName=databaseName	http://192.168.25.160/API?session_id=a33219dc&com		
&EPC= <i>EPC</i>	mand=delTagDatabaseRecord&databaseName=Product		
	DB&EPC=01234567890123456789ABCD		
	Note:		
	EPC field is always used as the primary key field in the		
	database. It must be included in the delTagDatabaseRecord		
	message.		
	result:		
	xml version="1.0" ? <csl></csl>		

		<command/> delTagDatabaseRecord <ack>OK: </ack>
105	session_id= <login_session_id>&</login_session_id>	List tag records in the database.
	command=listTagDatabaseRecord	e.g.
	&databaseName=databaseName	http://192.168.25.160/API?session_id=a33219dc&com
		mand=listTagDatabaseRecord&databaseName=Product
		<u>DB</u>
		Note:
		Use getTagDatabaseRecordImage command to get the image
		field.
		result:
		xml version="1.0" ?
		<csl> <command/>listTagDatabaseRecord</csl>
		<tagdatabaserecordlist> <tagdatabaserecord< th=""></tagdatabaserecord<></tagdatabaserecordlist>
		EPC="01234567890123456789ABCD"
		ProductId="1234" ProductName="Orange Juice"
		ProductImage=""
		ProductPrice=" 1.75 " /> <tagdatabaserecord< th=""></tagdatabaserecord<>
		EPC="000000020090505095227234"
		ProductId="2345" ProductName="Apple Juice"
		ProductImage=""
		ProductPrice=" 1.65 " />
106	session_id= <login_session_id>&</login_session_id>	Set an image to the image field of a record in the database.
	command=setTagDatabaseRecordI	
	mage	Here below is an example showing how to set an image to the
	&databaseName=databaseName	image field of a record in the database via HTTP POST
	&EPC= <i>EPC</i>	protocol written in C# (printed in blue color).
	&imageFieldName=imageFieldNam	
	e	HttpClient client = new HttpClient();

```
&imageFileExtension=imageFileExt var stream = new FileStream("C:\\temp\\image.jpg",
   ension
                                  FileMode.Open);
                                  var content = new StreamContent(stream);
                                  var requestUri =
                                  "http://192.168.25.160/API?session_id=a33219dc&co
                                  mmand=setTagDatabaseRecordImage&databaseName
                                   =ProductDB&EPC=01234567890123456789ABCD&ima
                                  geFieldName=ProductImage&imageFileExtension=jpg";
                                  var response = await client.PostAsync(requestUri,
                                  content);
                                  result:
                                   <?xml version="1.0" ?>
                                   <CSL>
                                     <Command>setTagDatabaseRecordImage
                                         </Command>
                                     <Ack>OK:</Ack>
                                   </CSL>
107 session id=<login session id>&
                                  Get the image from a record in the database.
   command=getTagDatabaseRecordI
                                  Here below is an example showing how to get the image from
   mage
   &databaseName=databaseName
                                  a record in the database via HTTP POST protocol written in C#
   &EPC=EPC
                                  (printed in blue color).
   &imageFieldName=imageFieldNam
                                  HttpClient client = new HttpClient();
                                  var requestUri =
                                   "http://192.168.25.160/API?session_id=a33219dc&co
                                  mmand=getTagDatabaseRecordImage&databaseName
                                   =ProductDB&EPC=01234567890123456789ABCD&ima
                                  geFieldName=ProductImage";
                                  var response = await client.GetAsync(requestUri);
                                  var content = response.Content;
                                  if(content.Headers.ContentType.MediaType.Contains("i
                                  mage"))
                                       var fileName =
```

```
content.Headers.ContentDisposition.FileName;
                                         var stream =
                                    content.ReadAsStreamAsync().Result;
                                         using (var fileStream = new
                                    FileStream("C:\\temp\\" + fileName, FileMode.Create,
                                    FileAccess.Write, FileShare.None))
                                         {
                                             Await stream.CopyToAsync(fileStream);
                                         }
108 session_id=<login_session_id>&
                                    Backup the database and get the backup zip file.
   command=databaseBackup
                                    e.g.
    &databaseName=databaseName
                                    http://192.168.25.160/API?session_id=a33219dc&com
                                    mand=databaseBackup&databaseName=ProductDB
                                    Here below is an example, written in C# (printed in blue color),
                                    showing how to backup database ProductDB and save it as
                                    Backup.zip in C:\temp.
                                    HttpClient client = new HttpClient();
                                    var requestUri =
                                    "http://192.168.25.160/API?session=a33219dc&comm
                                    and=databaseBackup&databaseName=ProductDB";
                                    var response = await client.GetAsync(requestUri);
                                    var stream =
                                    response.Content.ReadAsStreamAsync().Result;
                                    using (var fileStream = new
                                    FileStream("C:\\temp\\Backup.zip", FileMode.Create,
                                    FileAccess.Write, FileShare.None))
                                        await stream.CopyToAsync(fileStream);
109 session_id=<login_session_id>&
                                    Restore the database by loading the backup file to the reader.
```

	command= <i>databaseRestore</i>		
		Here below is an example showing how to restore the database	
		via HTTP POST protocol written in C# (printed in blue color).	
		HttpClient client = new HttpClient();	
		<pre>var stream = new FileStream("C:\\temp\\Backup.zip",</pre>	
		FileMode.Open);	
		var content = new StreamContent(stream);	
		var requestUri =	
		"http://192.168.25.160/API?session=a33219dc&comm	
		and=databaseRestore";	
		var response = await client.PostAsync(requestUri,	
		content);	
110	session_id= <login_session_id>&</login_session_id>	Add display format of tag data for displaying on web browser.	
	command=addDisplayFormat&		
	display_format_id=display_format_i	e.g.	
	d	session_id= <login_session_id>&command=addDisplayFormar</login_session_id>	
	[&databaseName=databaseName]	&display_format_id=DBDisplayFormat&databaseName=Prod	
	&fieldName{n}= <i>fieldName</i>	uctDB&fieldName1=ProductId&displayLabel1=ID&topPositi	
	[&displayLabel{n}=displayLabel]	on1=10&leftPosition1=10&fontSize1=16&fontColor1=%2300	
	&topPosition{n}=topPosition	0000&fieldName2=ProductName&displayLabel2=Name⊤	
	&leftPosition{n}=leftPosition	Position2=30&leftPosition2=10&fontSize2=16&fontColor2=	
	[&fontSize{n}=fontSize	%23000000&fieldName3=ProductPrice&displayLabel3=Price	
	&fontColor{n}=fontColor]	&topPosition3=50&leftPosition3=10&fontSize3=16&fontColo	
	[&imageHeight{n}=imageHeight	r3=%23000000&fieldName4=ProductImage&topPosition4=10	
	&imageWidth{n}=imageWidth]	&leftPosition4=200&imageHeight4=0&imageWidth4=0	
		Valid attributes :	
		For displaying database record data:	
		databaseName: must be included	
		fieldName: DatabaseName, Time or the name of	
		field in the database	
		displayLabel: label displayed before data	

```
topPosition:
                                                   unit = px
                                     leftPosition:
                                                   unit = px
                                     fontSize:
                                                   unit = px
                                     fontColor:
                                                   color name (like "red") or hex code (like
                                                   "#ff0000")
                                     imageHeight: unit = px, or 0 means auto
                                     imageWidth:
                                                   unit = px, or 0 means auto
                                     For displaying tag group data:
                                     databaseName: must be ignored
                                     fieldName:
                                                   TagGroupId, Time, TagId
                                     displayLabel: label displayed before data
                                     topPosition : unit = px
                                     leftPosition : unit = px
                                     fontSize:
                                                   unit = px
                                     fontColor:
                                                   color name (like "red") or hex code (like
                                                    "#ff0000")
                                     result:
                                     <?xml version="1.0" ?>
                                        <Command>addDisplayFormat</Command>
                                        <Ack>OK:</Ack>
                                     </CSL>
111 session_id=<login_session_id>&
                                     Modify display format of tag data for displaying on web
   command=modDisplayFormat&
                                     browser.
   display_format_id=display_format_i
                                     e.g.
   [&databaseName=databaseName]
                                     session_id=<login_session_id>&command=modDisplayForma
                                     t&display_format_id=DBDisplayFormat&databaseName=Prod
    &fieldName{n}=fieldName
    [&displayLabel{n}=displayLabel]
                                     uctDB&fieldName1=Time&displayLabel1=Time&topPosition
                                     1=10&leftPosition1=10&fontSize1=16&fontColor1=%230000
    &topPosition{n}=topPosition
    &leftPosition{n}=leftPosition
                                     00&fieldName2=ProductId&displayLabel2=ID&topPosition2=
                                     30&leftPosition2=10&fontSize2=16&fontColor2=%23000000
    [&fontSize{n}=fontSize
```

&fontColor{n}=fontColor &fieldName3=ProductName&displayLabel3=Name&topPositi [&imageHeight{n}=imageHeight on3=50&leftPosition3=10&fontSize3=16&fontColor3=%2300 &imageWidth{n}=imageWidth] 0000&fieldName4=ProductPrice&displayLabel4=Price&topPo sition4=70&leftPosition4=10&fontSize4=16&fontColor4=%23 000000&fieldName5=ProductImage&topPosition5=10&leftPo sition5=200&imageHeight5=0&imageWidth5=200 Valid attributes: For displaying database record data: databaseName: must be included fieldName: DatabaseName, Time or the name of field in the database displayLabel: label displayed before data topPosition: unit = pxleftPosition: unit = pxfontSize: unit = pxfontColor: color name (like "red") or hex code (like "#ff0000") imageHeight: unit = px, or 0 means auto imageWidth: unit = px, or 0 means auto For displaying tag group data: databaseName: must be ignored fieldName: TagGroupId, Time, TagId displayLabel: label displayed before data topPosition: unit = pxleftPosition : unit = pxfontSize: unit = pxfontColor: color name (like "red") or hex code (like "#ff0000") result: <?xml version="1.0" ?> <CSL>

```
<Command>addDisplayFormat</Command>
                                    <Ack>OK:</Ack>
                                  </CSL>
112 session_id=<login_session_id>&
                                 Remove display format.
   command=delDisplayFormat&
   display_format_id=display_format_i e.g.
   d
                                 session_id=<login_session_id>&command=delDisplayFormat
                                 &display format id=DBDisplayFormatFormat
                                 result:
                                  <?xml version="1.0" ?>
                                  <CSL>
                                    <Command>delDisplayFormat</Command>
                                    <Ack>OK:</Ack>
                                  </CSL>
113 session id=<login session id>&
                                 List display format.
   command=listDisplayFormat
                                 e.g.
                                 session_id=<login_session_id>&command=listDisplayFormat
                                 result:
                                  <?xml version="1.0" ?>
                                  <CSL>
                                    <Command>listDisplayFormat</Command>
                                    <DisplayFormatList>
                                     <displayFormat
                                          databaseName="ProductDB"
                                          display format id="DBDisplayFormat"
                                          displayLabel1="Time"
                                          displayLabel2="ID"
                                          displayLabel3="Name"
                                          displayLabel4="Price"
                                          displayLabel5=""
                                          fieldName1="Time"
                                          fieldName2="ProductId"
                                          fieldName3="ProductName"
                                          fieldName4="ProductPrice"
                                          fieldName5="ProductImage"
                                          fontColor1="#000000"
                                          fontColor2="#000000"
                                          fontColor3="#000000"
                                          fontColor4="#000000"
```

```
fontColor5="#000000"
                                            fontSize1="16.0"
                                            fontSize1="16.0"
                                            fontSize1="16.0"
                                            fontSize1="16.0"
                                            fontSize1="16.0"
                                            imageHeight1="0.0"
                                            imageHeight2="0.0"
                                            imageHeight3="0.0"
                                            imageHeight4="0.0"
                                            imageHeight5="0.0"
                                            imageWidth1="0.0"
                                            imageWidth2="0.0"
                                            imageWidth3="0.0"
                                            imageWidth4="0.0"
                                            imageWidth5="200.0"
                                            leftPosition1="10.0"
                                            leftPosition2="10.0"
                                            leftPosition3="10.0"
                                            leftPosition4="10.0"
                                            leftPosition5="200.0"
                                            topPosition1="10.0"
                                            topPosition2="30.0"
                                            topPosition3="50.0"
                                            topPosition4="70.0"
                                            topPosition5="10.0" />
                                      </DataFormatList>
                                    </CSL>
114 session_id=<login_session_id>&
                                   Read Temperature in degree Celsius from FM13DT160 tag.
   command=fm13dt160ReadTemperat
   ure
                                   e.g.
    &linkProfile=linkProfile
                                    session_id=<login_session_id>&command=fm13dt160ReadTe
    &antennaPort=antennaPort
                                   mperature&linkProfile=1&antennaPort=1&transmitPower=30
    &transmitPower=transmitPower
                                    &dwellTime=2000&reflectedPowerThreshold=24&maskBank
    &dwellTime=dwellTime
                                    Bank1&mask=E2827001000000000000001&accessPasswor
    &reflectedPowerThreshold=reflecte
                                   d=00000000
    dPowerThreshold
    &maskBank=maskBank
                                    Valid attributes:
    &mask=mask
                                    linkProfile:
                                                 0 = Multipath Interface Resistance
    &accessPassword=accessPassword
                                                 1 = Range/Dense Reader
                                                 2 = Range/Throughput/Dense Reader
                                                 3 = Max Throughput
                                                 1 - 16
                                   antennaPort:
```

```
transmitPower: 0.0 - 32.0 in step of 0.1 dBm
                                    dwellTime:
                                                  unit=ms, \geq 0ms
                                    reflectedPowerThreshold: 1.0 – 32.0 in step of 0.1 dBm
                                    maskBank:
                                                  Bank0, Bank1, Bank2, Bank3
                                    result:
                                    <?xml version="1.0" ?>
                                    <CSL>
                                      <Command>fm13dt160ReadTemperature
                                           </Command>
                                      <EPC>E2827001000000000000001</EPC>
                                      <RSSI>-44.00</RSSI>
                                      <Temperature>26.25</Temperature>
                                    </CSL>
                                    Read Battery Voltage in volt from FM13DT160 tag.
115 session_id=<login_session_id>&
   command=fm13dt160ReadBatteryV
   oltage
                                    e.g.
    &linkProfile=linkProfile
                                    session id=<login session id>&command=fm13dt160ReadBa
                                    tteryVoltage&linkProfile=1&antennaPort=1&transmitPower=3
    &antennaPort=antennaPort
    &transmitPower=transmitPower
                                    0&dwellTime=2000&reflectedPowerThreshold=24&maskBan
    &dwellTime=dwellTime
                                    k=Bank1&mask=E2827001000000000000001&accessPassw
    &reflectedPowerThreshold=reflecte
                                   ord=00000000
   dPowerThreshold
    &maskBank=maskBank
                                    Valid attributes:
    &mask=mask
                                    linkProfile:
                                                  0 = Multipath Interface Resistance
    &accessPassword=accessPassword
                                                  1 = Range/Dense Reader
                                                  2 = Range/Throughput/Dense Reader
                                                  3 = Max Throughput
                                    antennaPort:
                                                 1 - 16
                                    transmitPower: 0.0 - 32.0 in step of 0.1 dBm
                                    dwellTime:
                                                  unit=ms, \geq 0ms
                                    reflectedPowerThreshold: 1.0 – 32.0 in step of 0.1 dBm
                                    maskBank:
                                                  Bank0, Bank1, Bank2, Bank3
                                    result:
                                    <?xml version="1.0" ?>
                                    <CSL>
                                      <Command>fm13dt160ReadBatteryVoltage
```

```
</Command>
                                     <BatteryVoltage>1.47</BatteryVoltage>
                                     <EPC>E282700100000000000001</EPC>
                                     <RSSI>-44.00</RSSI>
                                   </CSL>
116 session_id=<login_session_id>&
                                   Read External Voltage in volt from FM13DT160 tag.
   command=fm13dt160ReadExtVolta
   ge
                                   e.g.
   &linkProfile=linkProfile
                                   session id=<login session id>&command=fm13dt160ReadEx
   &antennaPort=antennaPort
                                   tVoltage&linkProfile=1&antennaPort=1&transmitPower=30&d
   &transmitPower=transmitPower
                                   wellTime=2000&reflectedPowerThreshold=24&maskBank=Ba
                                   nk1&mask=E282700100000000000001&accessPassword=0
   &dwellTime=dwellTime
   &reflectedPowerThreshold=reflecte
                                   0000000
   dPowerThreshold
   &maskBank=maskBank
                                   Valid attributes:
   &mask=mask
                                   linkProfile:
                                                 0 = Multipath Interface Resistance
   &accessPassword=accessPassword
                                                 1 = Range/Dense Reader
                                                 2 = Range/Throughput/Dense Reader
                                                 3 = Max Throughput
                                   antennaPort:
                                                1 - 16
                                   transmitPower: 0.0 - 32.0 in step of 0.1 dBm
                                   dwellTime:
                                                 unit=ms, \geq 0ms
                                   reflectedPowerThreshold: 1.0 – 32.0 in step of 0.1 dBm
                                                 Bank0, Bank1, Bank2, Bank3
                                   maskBank:
                                   result:
                                   <?xml version="1.0" ?>
                                   <CSL>
                                     <Command>fm13dt160ReadExtVoltage
                                          </Command>
                                     <EPC>E2827001000000000000001</EPC>
                                     <ExtVoltage>1.47</ExtVoltage>
                                     <RSSI>-44.00</RSSI>
                                   </CSL>
117 session id=<login session id>&
                                   Read External Sensor Voltage in volt from FM13DT160 tag.
   command=fm13dt160ReadExtSens
```

	or Voltage	e.g.		
	&linkProfile=linkProfile	session_id= <login_session_id>&command=fm13dt160ReadEx</login_session_id>		
	&antennaPort=antennaPort	tSensorVoltage	tSensorVoltage&linkProfile=1&antennaPort=1&transmitPower	
	&transmitPower=transmitPower	=30&dwellTime=2000&reflectedPowerThreshold=24&maskB		
	&dwellTime=dwellTime	ank=Bank1&mask=E2827001000000000000001&accessPass		
	&reflectedPowerThreshold=reflecte	word=00000000		
	dPowerThreshold			
	&maskBank= <i>maskBank</i>	Valid attributes	:	
	&mask= <i>mask</i>	linkProfile:	0 = Multipath Interface Resistance	
	&accessPassword=accessPassword		1 = Range/Dense Reader	
			2 = Range/Throughput/Dense Reader	
			3 = Max Throughput	
		antennaPort:	1 – 16	
		transmitPower	: 0.0 – 32.0 in step of 0.1 dBm	
		dwellTime:	unit=ms, >= 0ms	
		reflectedPower'	Threshold: 1.0 – 32.0 in step of 0.1 dBm	
		maskBank :	Bank0, Bank1, Bank2, Bank3	
		, , , , , , , , , , , , , , , , , , , ,		
		result:		
		xml version="1.0" ?		
		<csl> <command/>fm13dt160ReadExtSensorVoltage</csl>		
			2700100000000000001 rVoltage> 1.47	
		<rssi>-44</rssi>	4.00	
118	session_id= <login_session_id>&</login_session_id>	Read data from	memory in FM13DT160 tag.	
	command=fm13dt160ReadMemory			
	&linkProfile=linkProfile	e.g.		
	&antennaPort=antennaPort		gin_session_id>&command=fm13dt160ReadM	
	&transmitPower=transmitPower	emory&linkPro	ofile=1&antennaPort=1&transmitPower=30&dw	
	&dwellTime=dwellTime	ellTime=2000&	reflectedPowerThreshold=24&maskBank=Ban	
	&reflectedPowerThreshold=reflecte	k1&mask=E282	270010000000000000001&accessPassword=00	
	dPowerThreshold	000000&address=b040&length=4		

	&maskBank= <i>maskBank</i>		
	&mask= <i>mask</i>	Valid attributes:	
	&accessPassword=accessPassword	linkProfile:	0 = Multipath Interface Resistance
	&address=address		1 = Range/Dense Reader
	&length=length		2 = Range/Throughput/Dense Reader
			3 = Max Throughput
		antennaPort:	1 – 16
		transmitPower	: 0.0 - 32.0 in step of 0.1 dBm
		dwellTime:	unit=ms, >= 0ms
		reflectedPower	Threshold: $1.0 - 32.0$ in step of 0.1 dBm
		maskBank :	Bank0, Bank1, Bank2, Bank3
		address:	0000 – C1FC, hex value and must be divisible
		by	4
		length:	0-500, must be a multiple of 4
		result:	
		xml version="1.0" ? <csl></csl>	
		<command/> fm13dt160ReadMemory	
		<data>4FB030CF</data>	
		<epc>E282700100000000000001</epc>	
		<rssi>-44.00</rssi> 	
119	session_id= <login_session_id>&</login_session_id>	Write data to m	nemory in FM13DT160 tag.
	command=fm13dt160WriteMemory		
	&linkProfile=linkProfile	e.g.	
	&antennaPort=antennaPort		gin_session_id>&command=fm13dt160WriteM
	&transmitPower=transmitPower	·	ofile=1&antennaPort=1&transmitPower=30&dw
	&dwellTime=dwellTime		&reflectedPowerThreshold=24&maskBank=Ban
	&reflectedPowerThreshold=reflecte	ccte k1&mask=E2827001000000000000001&accessPasswor	
	dPowerThreshold	000000&address=0&data=8C9F7E60	
	&maskBank= <i>maskBank</i>		
	&mask= <i>mask</i>	Valid attributes	
	&accessPassword=accessPassword	linkProfile:	0 = Multipath Interface Resistance

&length=length 2 = Range/Throughput/Dense Reader 3 = Max Throughput antennaPort: 1 - 16 transmitPower: 0.0 - 32.0 in step of 0.1 dBm dwellTime: unit=ms, >= 0ms reflectedPowerThreshold: 1.0 - 32.0 in step of 0.1 dBm maskBank: Bank0, Bank1, Bank2, Bank3 address: 0000 - C1FF, hex value data: hex string, at most 4 bytes result: xml version="1.0" ? <csl> <command/>fm13dt160WriteMemory <ack>OK:</ack> <epc-e28270010000000000000001< epc=""> <rssi>-44.00</rssi> </epc-e28270010000000000000001<></csl> 120 session_id= <login_session_id>& command=fm13dt160ReadUserCfg &linkProfile=linkProfile &antennaPort=antennaPort &transmitPower=ransmitPower &dwellTime=dwellTime &treflectedPowerThreshold=reflected dPowerThreshold &maskBank=maskBank &mask=mask &accessPassword=accessPassword &user_cfg=user_cfg 1 = Range/Dense Reader 2 = Range/Throughput/Dense Reader 3 = Max Throughput antennaPort: 1 - 16</login_session_id>		&address=address		1 = Range/Dense Reader
antennaPort: 1 – 16 transmitPower: 0.0 – 32.0 in step of 0.1 dBm dwellTime: unit=ms, >= 0ms reflectedPowerThreshold: 1.0 – 32.0 in step of 0.1 dBm maskBank: Bank0, Bank1, Bank2, Bank3 address: 0000 – C1FF, hex value data: hex string, at most 4 bytes result: <pre></pre>		&length=length		2 = Range/Throughput/Dense Reader
transmitPower: 0.0 – 32.0 in step of 0.1 dBm dwellTime: unit=ms, >= 0ms reflectedPowerThreshold: 1.0 – 32.0 in step of 0.1 dBm maskBank: Bank0, Bank1, Bank2, Bank3 address: 0000 – C1FF, hex value data: hex string, at most 4 bytes result: <pre></pre>				3 = Max Throughput
dwellTime: unit=ms, >= 0ms reflectedPowerThreshold: 1.0 - 32.0 in step of 0.1 dBm maskBank: Bank0, Bank1, Bank2, Bank3 address: 0000 - C1FF, hex value data: hex string, at most 4 bytes result: <pre></pre>			antennaPort:	1 – 16
reflectedPowerThreshold: 1.0 – 32.0 in step of 0.1 dBm maskBank: Bank0, Bank1, Bank2, Bank3 address: 0000 – C1FF, hex value data: hex string, at most 4 bytes result: <pre></pre>			transmitPower	: 0.0 – 32.0 in step of 0.1 dBm
maskBank: Bank0, Bank1, Bank2, Bank3 address: 0000 – C1FF, hex value data: hex string, at most 4 bytes result: ?xml version="1.0" ? CSL command fm13dt160WriteMemory			dwellTime :	unit=ms, >= 0ms
address: 0000 – C1FF, hex value data: hex string, at most 4 bytes result: ?xml version="1.0" ? <csl> <command/> fm13dt160WriteMemory </csl> <command/> Ack> OK: 〈Ack> <epc> E2827001000000000000000000000000000000000</epc>			reflectedPower	Threshold: $1.0 - 32.0$ in step of 0.1 dBm
data: hex string, at most 4 bytes result: <pre> <pre> <pre> cSL></pre></pre></pre>			maskBank :	Bank0, Bank1, Bank2, Bank3
result: <pre> <pre> <pre> <pre> <pre> contemporary</pre></pre></pre></pre></pre>			address:	0000 – C1FF, hex value
<pre></pre>			data :	hex string, at most 4 bytes
<pre></pre>				
CSL>			result:	
<pre></pre>			xml version</td <td>n="1.0" ?></td>	n="1.0" ?>
<pre></pre>				d. 6 4.2 da4 6.0 W. ib a Managara
<pre></pre>				
<pre></pre>				
CSL>				
command=fm13dt160ReadUserCfg &linkProfile=linkProfile &antennaPort=antennaPort &transmitPower=transmitPower &dwellTime=dwellTime &reflectedPowerThreshold=reflecte dPowerThreshold &maskBank=maskBank &mask=mask &mask=mask &transmitPower=transmitPower=30&dw ellTime=2000&reflectedPowerThreshold=24&maskBank=Ban &t = 2827001000000000000000000000000000000000				
command=fm13dt160ReadUserCfg &linkProfile=linkProfile &antennaPort=antennaPort &transmitPower=transmitPower &dwellTime=dwellTime &reflectedPowerThreshold=reflecte dPowerThreshold &maskBank=maskBank &mask=mask &mask=mask &transmitPower=transmitPower=30&dw ellTime=2000&reflectedPowerThreshold=24&maskBank=Ban &t = 2827001000000000000000000000000000000000	120	assism id design assism ids 0-	Dand was afa	Jaka from FM12DT170 to a
&linkProfile=linkProfile &antennaPort=antennaPort &transmitPower=transmitPower &dwellTime=dwellTime &reflectedPowerThreshold=reflecte dPowerThreshold &maskBank=mask &mask=mask &ccessPassword=accessPassword &user_cfg=user_cfg &linkProfile=1&antennaPort=1&transmitPower=30&dw ellTime=2000&reflectedPowerThreshold=24&maskBank=Ban k1&mask=E282700100000000000000000001&accessPassword=00 00000&user_cfg=0 Valid attributes: linkProfile: 0 = Multipath Interface Resistance 1 = Range/Dense Reader 2 = Range/Throughput/Dense Reader 3 = Max Throughput		C	Read user_cig	data from FW13D1160 tag.
&antennaPort=antennaPort &transmitPower=transmitPower &dwellTime=dwellTime &reflectedPowerThreshold=reflecte dPowerThreshold &maskBank=maskBank &mask=mask &mask=mask &ccessPassword=accessPassword &user_cfg=user_cfg bession_id= <login_session_id>&command=fm13dt160ReadUs erCfg&linkProfile=1&antennaPort=1&transmitPower=30&dw ellTime=2000&reflectedPowerThreshold=24&maskBank=Ban k1&mask=E2827001000000000000000000000000000000000</login_session_id>				
&transmitPower=transmitPower &dwellTime=dwellTime &reflectedPowerThreshold=reflecte dPowerThreshold &maskBank=maskBank &mask=mask &accessPassword=accessPassword &user_cfg=user_cfg &transmitPower=1&transmitPower=30&dw ellTime=2000&reflectedPowerThreshold=24&maskBank=Ban k1&mask=E2827001000000000000000000000000000000000		-		
&dwellTime=dwellTime &reflectedPowerThreshold=reflecte &dPowerThreshold &maskBank=maskBank &mask=mask &accessPassword=accessPassword &user_cfg=user_cfg &linkProfile: 0 = Multipath Interface Resistance & 1 = Range/Dense Reader & 2 = Range/Throughput/Dense Reader & 3 = Max Throughput				
&reflectedPowerThreshold=reflecte				
dPowerThreshold &maskBank=maskBank &mask=mask &mask=mask &accessPassword=accessPassword &user_cfg=user_cfg 1 = Range/Dense Reader 2 = Range/Throughput/Dense Reader 3 = Max Throughput		&dwellTime=dwellTime	ellTime=2000&	&reflectedPowerThreshold=24&maskBank=Ban
&maskBank=maskBank &mask=mask &accessPassword=accessPassword &user_cfg=user_cfg 1 = Range/Dense Reader 2 = Range/Throughput/Dense Reader 3 = Max Throughput		&reflectedPowerThreshold=reflecte	k1&mask=E28	270010000000000000001&accessPassword=00
&mask=mask &accessPassword=accessPassword &user_cfg=user_cfg LinkProfile: 0 = Multipath Interface Resistance 1 = Range/Dense Reader 2 = Range/Throughput/Dense Reader 3 = Max Throughput		dPowerThreshold	000000&user_0	efg=0
&accessPassword=accessPassword linkProfile: 0 = Multipath Interface Resistance &user_cfg=user_cfg		&maskBank=maskBank		
&user_cfg=user_cfg 1 = Range/Dense Reader 2 = Range/Throughput/Dense Reader 3 = Max Throughput		&mask= <i>mask</i>	Valid attributes	:
2 = Range/Throughput/Dense Reader 3 = Max Throughput		&accessPassword=accessPassword	linkProfile:	0 = Multipath Interface Resistance
3 = Max Throughput		&user_cfg=user_cfg		1 = Range/Dense Reader
				2 = Range/Throughput/Dense Reader
antennaPort: 1 – 16				3 = Max Throughput
			antennaPort:	1 – 16

```
transmitPower: 0.0 - 32.0 in step of 0.1 dBm
                                    dwellTime:
                                                  unit=ms, \geq 0ms
                                    reflectedPowerThreshold: 1.0 – 32.0 in step of 0.1 dBm
                                                  Bank0, Bank1, Bank2, Bank3
                                    maskBank:
                                                  0 - 3
                                    user cfg:
                                    result:
                                    <?xml version="1.0" ?>
                                    <CSL>
                                      <Command>fm13dt160ReadUserCfg
                                           </Command>
                                      <Data>4F</Data>
                                      <EPC>E2827001000000000000001</EPC>
                                       <RSSI>-44.00</RSSI>
                                    </CSL>
121 session_id=<login_session_id>&
                                    Write user_cfg data to FM13DT160 tag.
   command=fm13dt160WriteUserCfg
   &linkProfile=linkProfile
                                    e.g.
   &antennaPort=antennaPort
                                    session_id=<login_session_id>&command=fm13dt160WriteU
   &transmitPower=transmitPower
                                    serCfg&linkProfile=1&antennaPort=1&transmitPower=30&d
   &dwellTime=dwellTime
                                    wellTime=2000&reflectedPowerThreshold=24&maskBank=Ba
   &reflectedPowerThreshold=reflecte
                                    nk1&mask=E2827001000000000000001&accessPassword=0
   dPowerThreshold
                                    0000000&user_cfg=1&data=30
   &maskBank=maskBank
   &mask=mask
                                    Valid attributes:
   &accessPassword=accessPassword
                                    linkProfile:
                                                  0 = Multipath Interface Resistance
   &user_cfg=user_cfg
                                                  1 = Range/Dense Reader
   &data =data
                                                  2 = Range/Throughput/Dense Reader
                                                  3 = Max Throughput
                                                  1 - 16
                                    antennaPort:
                                    transmitPower: 0.0 - 32.0 in step of 0.1 dBm
                                    dwellTime:
                                                  unit=ms, >= 0ms
                                    reflectedPowerThreshold: 1.0 – 32.0 in step of 0.1 dBm
                                    maskBank:
                                                  Bank0, Bank1, Bank2, Bank3
                                                  0 - 3
                                    user_cfg:
```

```
00 – FF, hex value, 1 byte only
                                   data:
                                   result:
                                    <?xml version="1.0" ?>
                                      <Command>fm13dt160WriteUserCfg
                                          </Command>
                                      <Ack>OK:</Ack>
                                      <EPC>E2827001000000000000001</EPC>
                                      <RSSI>-44.00</RSSI>
                                    </CSL>
122 session_id=<login_session_id>&
                                   Read data from register in FM13DT160 tag.
   command=fm13dt160ReadReg
   &linkProfile=linkProfile
                                   e.g.
   &antennaPort=antennaPort
                                   session_id=<login_session_id>&command=fm13dt160ReadRe
   &transmitPower=transmitPower
                                   g&linkProfile=1&antennaPort=1&transmitPower=30&dwellTi
   &dwellTime=dwellTime
                                   me=2000&reflectedPowerThreshold=24&maskBank=Bank1&
                                   mask=E2827001000000000000001&accessPassword=00000
   &reflectedPowerThreshold=reflecte
   dPowerThreshold
                                   000&address=c000
   &maskBank=maskBank
   &mask=mask
                                    Valid attributes:
   &accessPassword=accessPassword
                                   linkProfile:
                                                 0 = Multipath Interface Resistance
   &address=address
                                                 1 = Range/Dense Reader
                                                 2 = Range/Throughput/Dense Reader
                                                 3 = Max Throughput
                                                 1 - 16
                                   antennaPort:
                                   transmitPower: 0.0 - 32.0 in step of 0.1 dBm
                                   dwellTime:
                                                 unit=ms, >= 0ms
                                   reflectedPowerThreshold: 1.0 – 32.0 in step of 0.1 dBm
                                   maskBank:
                                                 Bank0, Bank1, Bank2, Bank3
                                                 c000 - c0ff, hex value
                                   address:
                                   result:
                                    <?xml version="1.0" ?>
                                    <CSL>
                                      <Command>fm13dt160ReadReg</Command>
                                      <Data>0600</Data>
```

		<epc>E282700100000000000001</epc> <rssi>-44.00</rssi>	
123	session_id= <login_session_id>&</login_session_id>	Write data to re	egister in FM13DT160 tag.
	command=fm13dt160WriteReg		
	&linkProfile=linkProfile	e.g.	
	&antennaPort=antennaPort	session_id= <lo< th=""><th>ogin_session_id>&command=fm13dt160WriteR</th></lo<>	ogin_session_id>&command=fm13dt160WriteR
	&transmitPower=transmitPower	eg&linkProfile	=1&antennaPort=1&transmitPower=30&dwellT
	&dwellTime=dwellTime	ime=2000&ref	lectedPowerThreshold=24&maskBank=Bank1&
	&reflectedPowerThreshold=reflecte	mask=E282700	0100000000000000001&accessPassword=00000
	dPowerThreshold	000&address=	c000&data=0600
	&maskBank= <i>maskBank</i>		
	&mask= <i>mask</i>	Valid attributes	3:
	&accessPassword=accessPassword	linkProfile:	0 = Multipath Interface Resistance
	&address=address		1 = Range/Dense Reader
	&data =data		2 = Range/Throughput/Dense Reader
			3 = Max Throughput
		antennaPort:	1 – 16
		transmitPower	: 0.0 – 32.0 in step of 0.1 dBm
		dwellTime:	unit=ms, >= 0ms
		reflectedPower	Threshold: 1.0 – 32.0 in step of 0.1 dBm
		maskBank:	Bank0, Bank1, Bank2, Bank3
		address:	c000 – c0ff, hex value
		data :	0000 – ffff, hex value
		result:	
		<pre><?xml version="1.0" ?> <csl></csl></pre>	
124	session_id= <login_session_id>&</login_session_id>	Send Deep Sleep command to FM13DT160 tag.	
	command=fm13dt160DeepSleep		

	&linkProfile=linkProfile	e.g.		
	&antennaPort=antennaPort	session_id= <login_session_id>&command=fm13dt160DeepS1</login_session_id>		
	&transmitPower=transmitPower	eep&linkProfil	eep&linkProfile=1&antennaPort=1&transmitPower=30&dwell	
	&dwellTime=dwellTime	Time=2000&re	Time=2000&reflectedPowerThreshold=24&maskBank=Bank1	
	&reflectedPowerThreshold=reflecte	&mask=E2827001000000000000001&accessPassword=0000		
	dPowerThreshold	0000&enable=true		
	&maskBank= <i>maskBank</i>			
	&mask= <i>mask</i>	Valid attributes	::	
	&accessPassword=accessPassword	linkProfile:	0 = Multipath Interface Resistance	
	&enable= <i>enable</i>		1 = Range/Dense Reader	
			2 = Range/Throughput/Dense Reader	
			3 = Max Throughput	
		antennaPort:	1 – 16	
		transmitPower: 0.0 – 32.0 in step of 0.1 dBm		
		dwellTime: unit=ms, >= 0ms		
		reflectedPower	Threshold: 1.0 – 32.0 in step of 0.1 dBm	
		maskBank:	Bank0, Bank1, Bank2, Bank3	
		enable:	true, false	
		result:		
		xml version="1.0" ?		
		<csl> <comman< th=""><th>d>fm13dt160DeepSleep</th></comman<></csl>	d>fm13dt160DeepSleep	
		<ack>OK:</ack>	32700100000000000001	
		<rssi>-4</rssi>	4.00	
125	session_id= <login_session_id>&</login_session_id>	Send Op Mode	e_Chk command to FM13DT160 tag.	
	command=fm13dt160OpModeChk	1 - 33	_	
	&linkProfile=linkProfile	e.g.		
	&antennaPort=antennaPort		ogin_session_id>&command=fm13dt160OpMod	
	&transmitPower=transmitPower		file=1&antennaPort=1&transmitPower=30&dwe	
	&dwellTime= <i>dwellTime</i>	llTime=2000&	reflectedPowerThreshold=24&maskBank=Bank	
	&reflectedPowerThreshold=reflecte			
	dPowerThreshold	00000&refresh	00000&refreshTempMeasurement=false	

	&maskBank= <i>maskBank</i>			
	&mask= <i>mask</i>	Valid attributes	:	
	&accessPassword=accessPassword	linkProfile:	0 = Multipath Interface Resistance	
	&refreshTempMeasurement=refresh		1 = Range/Dense Reader	
	TempMeasurement		2 = Range/Throughput/Dense Reader	
			3 = Max Throughput	
		antennaPort:	1 - 16	
		transmitPower	0.0 - 32.0 in step of 0.1 dBm	
		dwellTime:	unit=ms, >= 0ms	
		reflectedPower	Threshold: $1.0 - 32.0$ in step of 0.1 dBm	
		maskBank :	Bank0, Bank1, Bank2, Bank3	
		refreshTempMe	easurement: true, false	
		result:		
		xml version</td <td>n="1.0" ?></td>	n="1.0" ?>	
		<csl> <command/>fm13dt160OpModeChk</csl>		
		<epc>E282700100000000000001</epc>		
		<pre><opmode>user_access_en,vbat_pwr_flag </opmode></pre>		
		<rssi>-44 </rssi>	1.00	
		OpMode:		
		user_access_en	: user has valid access right	
		rtc_logging:	RTC logging in progress	
		vdet_process_fl	ag: instant temperature measurement is	
			interrupted	
		light_chk_flag	light strength over preset value	
		vbat_pwr_flag	battery voltage is higher than 0.9V	
126	session_id= <login_session_id>&</login_session_id>	Send Initial Reg	gfile command to FM13DT160 tag.	
	command=fm13dt160InitialRegfile			
	&linkProfile=linkProfile	e.g.		
	&antennaPort=antennaPort	session_id= <login_session_id>&command=fm13dt160InitialR</login_session_id>		
	&transmitPower=transmitPower	egfile&linkProf	egfile&linkProfile=1&antennaPort=1&transmitPower=30&dw	

	&dwellTime=dwellTime	ellTime=2000&	ellTime=2000&reflectedPowerThreshold=24&maskBank=Ban	
	&reflectedPowerThreshold=reflecte	k1&mask=E28	270010000000000000001&accessPassword=00	
	dPowerThreshold	000000		
	&maskBank= <i>maskBank</i>			
	&mask= <i>mask</i>	Valid attributes	:	
	&accessPassword=accessPassword	linkProfile:	0 = Multipath Interface Resistance	
			1 = Range/Dense Reader	
			2 = Range/Throughput/Dense Reader	
			3 = Max Throughput	
		antennaPort:	1 – 16	
		transmitPower	: 0.0 – 32.0 in step of 0.1 dBm	
		dwellTime :	unit=ms, >= 0ms	
		reflectedPower	reflectedPowerThreshold: 1.0 – 32.0 in step of 0.1 dBm	
		maskBank: Bank0, Bank1, Bank2, Bank3		
		result:		
		xml version="1.0" ?		
		<csl> <command/>fm13dt160InitialRegfile</csl>		
		<ack>OK:</ack> <epc>E28270010000000000001</epc>		
		<rssi>-44.00</rssi>		
127	session_id= <login_session_id>&</login_session_id>	Send Led Ctrl	command to FM13DT160 tag.	
	command=fm13dt160LedCtrl			
	&linkProfile=linkProfile	e.g.		
	&antennaPort=antennaPort	session_id= <lo< td=""><td>gin_session_id>&command=fm13dt160LedCtrl</td></lo<>	gin_session_id>&command=fm13dt160LedCtrl	
	&transmitPower=transmitPower	&linkProfile=1	&antennaPort=1&transmitPower=30&dwellTi	
	&dwellTime=dwellTime	me=2000&refle	ectedPowerThreshold=24&maskBank=Bank1&	
	&reflectedPowerThreshold=reflecte	mask=E282700	01000000000000000001&accessPassword=00000	
	dPowerThreshold	000&enable=tr	rue	
	&maskBank= <i>maskBank</i>			
	&mask= <i>mask</i>	Valid attributes	:	
	&accessPassword=accessPassword	linkProfile:	0 = Multipath Interface Resistance	

	&enable= <i>enable</i>		1 = Range/Dense Reader
			2 = Range/Throughput/Dense Reader
			3 = Max Throughput
		antennaPort:	1 - 16
		transmitPower	0.0 - 32.0 in step of 0.1 dBm
		dwellTime:	unit=ms, >= 0ms
		reflectedPower'	Threshold: 1.0 – 32.0 in step of 0.1 dBm
		maskBank :	Bank0, Bank1, Bank2, Bank3
		enable :	true, false
		result:	
		xml version</td <td>n="1.0" ?></td>	n="1.0" ?>
		<csl></csl>	d>fm13dt160LedCtrl
		<ack>OK:</ack>	
			2700100000000000001 4.00
			,
120	assism id dosin assism ids h	Cand Start Lagr	sing command to EM12DT160 tog
		Send Start Logging command to FM13DT160 tag.	
	command=fm13dt160StartLogging		
		e.g.	- ' '- '- '- '- '- '- '- '- '- '- '
	&antennaPort=antennaPort	•	gin_session_id>&command=fm13dt160StartLo
	&transmitPower=transmitPower		File=1&antennaPort=1&transmitPower=30&dw
			creflectedPowerThreshold=24&maskBank=Ban
	· ·		2700100000000000000001&accessPassword=00
			elay=5&timeStep=30&sampleNumber=5&sam
	&maskBank= <i>maskBank</i>		sampleFlashLength=1&outOfLimitFlash=true
	&mask= <i>mask</i>		ashLength=0.5&outOfLimitFlashNumber=3&1
			Out_Of_Range_Only&minLimit=0&maxLimit=
	&startDelay=startDelay	20	
	&timeStep=timeStep		
	&sampleNumber=sampleNumber	Valid attributes	:
	&sampleFlash=sampleFlash	linkProfile:	0 = Multipath Interface Resistance
	[&sampleFlashLength=sampleFlash		1 = Range/Dense Reader
	Length]		2 = Range/Throughput/Dense Reader

```
[&outOfLimitFlash=outOfLimitFlas
                                                 3 = Max Throughput
                                  antennaPort:
                                                 1 - 16
&outOfLimitFlashLength=outOfLim transmitPower: 0.0 – 32.0 in step of 0.1 dBm
itFlashLength
                                  dwellTime:
                                                 unit=ms, \geq 0ms
&outOfLimitFlashNumber=outOfLi
                                 reflectedPowerThreshold: 1.0 – 32.0 in step of 0.1 dBm
mitFlashNumber]
                                  maskBank:
                                                 Bank0, Bank1, Bank2, Bank3
&loggingMode=loggingMode
                                  startDelay:
                                                 unit minute, 0 - 65535, time delay to start
[&minLimit=minLimit]
                                                 logging after command is received
&maxLimit=maxLimit]
                                  timeStep:
                                                 unit second, 1 - 65535, time interval between
                                                 each temperature measurement sample
                                  sampleNumber:
                                                      1 - 4864, number of samples to be taken
                                  sampleFlash: true, false, LED flash after each sampling
                                  sampleFlashLength: unit second, 0.1 – 1.6, LED flash
                                                      time duration after each sampling
                                  outOfLimitFlash:
                                                     true, false, LED flash if temperature
                                                      sample is lower than the minLimit or
                                                      higher than the maxLimit
                                  outOfLimitFlash:
                                                      unit second, 0.1 - 1.6, LED flash time
                                                      duration if temperature sample is out of
                                                      the preset limit
                                  outOfLimitFlashNumber : 1 - 15, number of flashes if
                                                          temperature sample is out of the
                                                          preset limit
                                  loggingMode: Normal, Out_Of_Range_Only
                                  minLimit:
                                                 unit degree Celsius, -127.75 – 127.75
                                  maxLimit:
                                                 unit degree Celsius, -127.75 – 127.75
                                  result:
                                  <?xml version="1.0" ?>
                                  <CSL>
                                    <Command>fm13dt160StartLogging
                                         </Command>
                                    <Ack>OK:</Ack>
                                    <EPC>E282700100000000000001</EPC>
                                     <RSSI>-44.00</RSSI>
                                   </CSL>
```

129	session_id= <login_session_id>&</login_session_id>	Send Stop Logging command to FM13DT160 tag.		
	command=fm13dt160StopLogging			
	&linkProfile=linkProfile	e.g.		
	&antennaPort=antennaPort	session_id= <login_session_id>&command=fm13dt160StopLo</login_session_id>		
	&transmitPower=transmitPower	gging&linkProfile=1&antennaPort=1&transmitPower=30&dw		
	&dwellTime=dwellTime	ellTime=2000&reflectedPowerThreshold=24&maskBank=Ban		
	&reflectedPowerThreshold=reflecte	k1&mask=E282700100000000000001&accessPassword=00		
	dPowerThreshold	000000		
	&maskBank= <i>maskBank</i>			
	&mask= <i>mask</i>	Valid attributes	:	
	&accessPassword=accessPassword	linkProfile:	0 = Multipath Interface Resistance	
			1 = Range/Dense Reader	
			2 = Range/Throughput/Dense Reader	
			3 = Max Throughput	
		antennaPort:	1 – 16	
		transmitPower	: 0.0 - 32.0 in step of 0.1 dBm	
		dwellTime :	unit=ms, >= 0ms	
		reflectedPowerThreshold: 1.0 – 32.0 in step of 0.1 dBm		
		maskBank: Bank0, Bank1, Bank2, Bank3		
		result:		
		<pre><?xml version="1.0" ?></pre>		
		<csl></csl>		
		<command/> fm13dt160StopLogging		
		<ack>OK:</ack>	•	
			27001000000000000001 4.00	
130	session_id= <login_session_id>&</login_session_id>	Send Get Logo	ing command to FM13DT160 tag.	
	command=fm13dt160GetLogging	23112 201 2088		
	&linkProfile=linkProfile	e.g.		
	&antennaPort=antennaPort		gin_session_id>&command=fm13dt160GetLog	
	&transmitPower=transmitPower		le=1&antennaPort=1&transmitPower=30&dwel	
		5S		

```
lTime=2000&reflectedPowerThreshold=24&maskBank=Bank
&dwellTime=dwellTime
&reflectedPowerThreshold=reflecte
                              1&mask=E282700100000000000001&accessPassword=000
dPowerThreshold
                              00000
&maskBank=maskBank
                              Valid attributes:
&mask=mask
&accessPassword=accessPassword
                              linkProfile:
                                           0 = Multipath Interface Resistance
                                           1 = Range/Dense Reader
                                           2 = Range/Throughput/Dense Reader
                                           3 = Max Throughput
                              antennaPort:
                                           1 - 16
                              transmitPower: 0.0 - 32.0 in step of 0.1 dBm
                              dwellTime:
                                           unit=ms. \geq 0ms
                              reflectedPowerThreshold: 1.0 – 32.0 in step of 0.1 dBm
                                           Bank0, Bank1, Bank2, Bank3
                              maskBank:
                              result:
                              <?xml version="1.0" ?>
                              <CSL>
                                <Command>fm13dt160GetLogging</Command>
                                <EPC>E282700100000000000001</EPC>
                                <RSSI>-44.00</RSSI>
                                <LocalStartTime>Tue Apr 20 17:26:56
                                     2021</LocalStartTime>
                                <UTCStartTime>Tue Apr 20 09:26:56
                                     2021</UTCStartTime>
                                <StartDelay>5</StartDelay>
                                <StartDelayUnit>minute</StartDelayUnit>
                                <TimeStep>30</TimeStep>
                                <TimeStepUnit>second</TimeStepUnit>
                                <LoggingMode>Out_Of_Range_Only</Lo
                                    ggingMode>
                                <MinLimit>0</MinLimit>
                                <MaxLimit>20</MaxLimit>
                                <LogList>
                                  <log temperature="22.25"</li>
                                      localTime="Tue Apr 20 17:31:56 2021"
                                       utcTime="Tue Apr 20 09:31:56 2021" />
                                  <log temperature="22.50"</li>
                                      localTime="Tue Apr 20 17:32:26 2021"
                                      utcTime="Tue Apr 20 09:32:26 2021" />
                                  log temperature="22.50"
                                      localTime="Tue Apr 20 17:32:56 2021"
                                       utcTime="Tue Apr 20 09:32:56 2021" />
                                  <log temperature="22.25"</li>
```

```
localTime="Tue Apr 20 17:33:26 2021"
                                              utcTime="Tue Apr 20 09:33:26 2021" />
                                         <log temperature="22.50"</li>
                                              localTime="Tue Apr 20 17:33:56 2021"
                                              utcTime="Tue Apr 20 09:33:56 2021" />
                                     </CSL>
131 session_id=<login_session_id>&
                                    Read tag data.
   command=readTag
    &linkProfile=linkProfile
                                    e.g.
    &antennaPort=antennaPort
                                     session_id=<login_session_id>&command=readTag&linkProfi
    &transmitPower=transmitPower
                                     le=1&antennaPort=1&transmitPower=30&dwellTime=2000&r
    &dwellTime=dwellTime
                                    eflectedPowerThreshold=24&maskBank=Bank1&mask=E282
    &reflectedPowerThreshold=reflecte
                                    70010000000000000001&accessPassword=00000000&readAc
    dPowerThreshold
                                     cessPassword=true&readKillPassword=true&readTidBank=tru
    &maskBank=maskBank
                                    e&tidBankOffset=0&tidBankLength=2&readUserBank=true&
    &mask=mask
                                    userBankOffset=0&userBankLength=2
    &accessPassword=accessPassword
    &readAccessPassword=readAccess
                                     Valid attributes:
   Password
                                    linkProfile:
                                                   0 = Multipath Interface Resistance
    &readKillPassword=readKillPasswo
                                                   1 = Range/Dense Reader
   rd
                                                   2 = Range/Throughput/Dense Reader
   &readTidBank=readTidBank
                                                   3 = Max Throughput
    [&tidBankOffset=tidBankOffset
                                    antennaPort:
                                                   1 - 16
    &tidBankLength=tidBankLength]
                                    transmitPower: 0.0 - 32.0 in step of 0.1 dBm
    &readUserBank=readUserBank
                                                   unit=ms. \geq 0ms
                                    dwellTime:
    [&userBankOffset=userBankOffset
                                    reflectedPowerThreshold: 1.0 – 32.0 in step of 0.1 dBm
    &userBankLength=userBankLength||maskBank:
                                                   Bank0, Bank1, Bank2, Bank3
                                     readAccessPassword:
                                                            true, false
                                    readKillPassword:
                                                            true, false
                                    readTidBank:
                                                            true, false
                                     readUserBank:
                                                            true, false
                                     result:
                                     <?xml version="1.0" ?>
                                     <CSL>
```

		<pc>3000 <epc>E28 <accesspas <killpasswo <tidbank></tidbank></killpasswo </accesspas </epc></pc>	2700100 ssword>0 ord>0000 E282700	<pre>g 0000000000001 0000000 0000 1 61</pre>
132	session_id= <login_session_id>&</login_session_id>	Write tag data.		
	command=writeTag			
	&linkProfile=linkProfile	e.g.		
	&antennaPort=antennaPort	session_id= <log< td=""><td>gin_session</td><td>_id>&command=writeTag&linkProf</td></log<>	gin_session	_id>&command=writeTag&linkProf
	&transmitPower=transmitPower	ile=1&antennaF	Port=1&trai	nsmitPower=30&dwellTime=2000&
	&dwellTime=dwellTime	reflectedPower	Γhreshold=	24&maskBank=Bank1&mask=6161
	&reflectedPowerThreshold=reflecte	6161600000000	00000000&	accessPassword=00000000&writeP
	dPowerThreshold	C=true&newPC	C=3000&wi	riteEPC=true&newEPC=626263636
	&maskBank= <i>maskBank</i>	000000000000000000000000000000000000000	000&writeT	TidBank=true&newTidBankOffset=0
	&mask= <i>mask</i>	&newTidBank=E2801160		
	&accessPassword=accessPassword			
	[&writePC=writePC	Valid attributes	:	
	&newPC=newEPC]	linkProfile:	0 = Multip	eath Interface Resistance
	[&writeEPC=writeEPC		1 = Range	Dense Reader
	&newEPC=newEPC]		2 = Range	/Throughput/Dense Reader
	[&writeAccessPassword=writeAcces		3 = Max T	hroughput
	sPassword	antennaPort:	1 - 16	
	&newAccessPassword=newAccessP	transmitPower:	0.0 - 32.0	in step of 0.1 dBm
	assword]	dwellTime :	unit=ms, >	>= 0ms
	[&writeKillPassword=writeKillPass	reflectedPower	Threshold:	1.0 – 32.0 in step of 0.1 dBm
	word	maskBank:	Bank0, Ba	nk1, Bank2, Bank3
	&newKillPassword=newKillPasswo	writeAccessPas	sword:	true, false
	rd]	writeKillPasswo	ord:	true, false
	[&writeTidBank=writeTidBank	writeTidBank:		true, false
	&newTidBankOffset=newTidBankO	writeUserBank	:	true, false
	ffset			
	&newTidBank=newTidBank]	result:		
		xml version</td <td>n="1.0" ?</td> <td>></td>	n="1.0" ?	>

	[&writeUserBank=writeUserBank &newUserBankOffset=newBankOffs et &newUserBank=newUserBank]		<pc>3000 < <epc>6161 < <writepc>C < <writeepc></writeepc></writepc></epc></pc>	> writeTag < / Commar 61616000000000000000000000000000000000	00000	
133	session_id= <login_session_id>&</login_session_id>	A	dd Tag Filter.			
	command= <i>addTagFilter</i>					
	&tag_filter_id=tag_filter_id	e.g	3 .			
	&type=type	<u>ht</u>	tp://192.168	3.25.160/API?session	id=a33219dc&com	
	&bank=bank	m	<u>and=addTagl</u>	<u>Filter&tag filter id=Pr</u>	<u>e%20Filter%201&t</u>	
	&offset=offset	УP	e=PRE FILT	ER&bank=Bank1&offse	et=0&mask=6161	
	&mask= <i>mask</i>	<u>&</u> a	action=0			
	&action=action	Va	lid attributes :			
			Valid attributes:			
		-	type: PRE_FILTER, POST_FILTER bank: Bank0, Bank1, Bank2, Bank3			
			offset: unit = bits			
		action:				
		If type is PRE_FILTER:				
			3 , F =		Tag Not	
			Action	Tag Matching	Matching	
			0	assert SL or	deassert SL or	
				inventoried -> A	inventoried -> B	
			1	assert SL or inventoried -> A	do nothing	
			2	do nothing	deassert SL or inventoried -> B	
			3	negate SL or $(A \rightarrow B, B \rightarrow A)$	do nothing	
			4	deassert SL or inventoried -> B	assert SL or inventoried -> A	
			5	deassert SL or inventoried -> B	do nothing	
			6	do nothing	assert SL or inventoried -> A	

		7	do nothing	negate SL or (A -> B, B -> A)			
		If type is POST_FILTER:					
		action 0 = Match mask					
		action 0 = Natch mask					
		action 1 – Ivot maten mask					
	res	result:					
	</td <td>?xml version=</td> <td>="1.0" ?></td> <td></td>	?xml version=	="1.0" ?>				
	<(CSL>	addTagFilter <td>umand></td>	umand>			
		<ack>OK:<!--</td--><td></td><td>illiana z</td></ack>		illiana z			
	</td <td>CSL></td> <td></td> <td></td>	CSL>					
134 session_id= <login_sess< td=""><td>on_id>& Mo</td><td>odiy Tag Filter.</td><td></td><td></td></login_sess<>	on_id>& Mo	odiy Tag Filter.					
command=modTagFilte	r						
&tag_filter_id=tag_filte	<i>r_id</i> e.g	<u>.</u>					
&type=type	<u>ht</u>	http://192.168.25.160/API?session_id=a33219dc&com					
&bank=bank	<u>ma</u>	mand=modTagFilter&tag_filter_id=Pre%20Filter%201&					
&offset=offset	<u>ty</u>	type=PRE FILTER&bank=Bank1&offset=0&mask=6262					
&mask= <i>mask</i>	<u>&a</u>	&action=0					
&action=action	Va	Valid attributes :					
	tyŗ	type: PRE_FILTER, POST_FILTER					
	ba	bank: Bank0, Bank1, Bank2, Bank3					
	off	offset : unit = bits					
	act	action:					
		If type is PRE_FILTER:					
		Action	Tag Matching	Tag Not Matching			
		0	assert SL or	deassert SL or			
			inventoried -> A	inventoried -> B			
		1	assert SL or inventoried -> A	do nothing			
		2	do nothing	deassert SL or inventoried -> B			
		3	negate SL or (A -> B, B -> A)	do nothing			

	I					
	4	deassert SL or inventoried -> B	assert SL or inventoried -> A			
	5	deassert SL or inventoried -> B	do nothing			
	6	do nothing	assert SL or inventoried -> A			
	7	do nothing	negate SL or (A -> B, B -> A)			
	If type is POST_FILTER:					
	action $0 = 1$	action 0 = Match mask				
	action 1 = 1	Not match mask				
	result:					
	xml version</td <td>="1 0" 2></td> <td></td>	="1 0" 2>				
	<csl></csl>					
		<command/> modTagFilter <ack>OK:</ack>				
		/ACK>				
135 session_id= <login_session_id>&</login_session_id>	Delete Tag Filter	•				
command=delTagFilter						
&tag_filter_id=tag_filter_id	e.g.					
	http://192.168.25.160/API?session_id=a33219dc&com					
	mand=delTagFilter&tag_filter_id=Pre%20Filter%201					
	result:					
	xml version</th <th>="1.0" ?></th> <th></th>	="1.0" ?>				
	<csl></csl>					
	<ack>OK:<</ack>		manu>			
136 session_id= <login_session_id>&</login_session_id>	List Tag Filter					
	List rag ritter					
command= <i>listTagFilter</i>						
	e.g.					
		3.25.160/API?session				
	mand=delTagF	<u> ilter&tag filter id=Pr</u>	<u>e%20Filter%201</u>			

```
result:
<?xml version="1.0" ?>
<CSL>
  <Command>listTagFilter</Command>
  <TagFilterList>
  <tagfilter action="0"
        bank="Bank1"
        mask="6161"
        offset="0"
        tag_filter_id="Pre Filter 1"
        type="PRE_FILTER" />
  <tagfilter action="1"
        bank="Bank3"
        mask="3005"
        offset="0"
        tag_filter_id="Post Filter 1"
        type="POST_FILTER" />
  </TagFilterList>
</CSL>
```

2. CS461 TCP Format to TCP Server

If a resultant action has a TCP transport type defined, tag data will be sent to the server by TCP protocol. The following describes the CS461 TCP Format of tag data.

(1) Tag Data

cmd=evtNtf&evt_id=%s&src_ip=%s&ant=Antenna%d&cp_id=%s&idx=A%d&tag_id=
%s&rssi=%d&time=%s&cnt=%d&freq=%d&PC=%04X&usec=%d\n

cmd=evtNtf&evt_id=DemoEvent&src_ip=192.168.25.224&ant=Antenna1&
cp_id=CapturePoint1&tag_id=10000000000000000000000004&rssi=-35&ti
me=1587524604&freq=924.25&phase=22.5&PC=3000&usec=158000

```
cmd is the command type, and in this case is event notification;
evt_id is the event ID;
src_ip is the reader IP address;
ant is the antenna port where the tag is received and is of the
form ant=Antennal or ant=Antenna2 etc;
cp_id is the capture point (alias read point) name;
rssi is the tag rssi in unit of dBm;
time is the time of tag capture based on Linux epoch time.
freq is the frequency in Hz.
phase is the phase in degree.
PC is the protocol control bits.
usec is the micro-seconds part of the time of tag capture based on
Linux epoch time.
```

(2) End of batch message

This message is sent after the last tag data in each packet.

```
cmd=evtNtf&batchEnd=yes\n
```

cmd=evtNtf is the command type.

(3) Tag Data with additional bank (bank0, bank2, bank3)

```
If bank0 is selected in the Active Operation Profile,

cmd=evtNtf&evt_id=%s&src_ip=%s&ant=Antenna%d&cp_id=%s&tag_id=%s&rssi=
%d&time=%s&bank0=%s&freq=%s&phase=%s&PC=%04X&usec=%d\n
```

If bank2 is selected in the Active Operation Profile,

 $cmd = evtNtf&evt_id = ss&src_ip = ss&ant = Antenna \\ d&cp_id = ss&tag_id = ss&rssi \\ d&time = ss&bank2 = ss&freq = ss&phase = \\ s&PC = s04X&usec = sd \\ n$

If bank3 is selected in the Active Operation Profile, cmd=evtNtf&evt_id=%s&src_ip=%s&ant=Antenna%d&cp_id=%s&tag_id=%s&rssi=%d&time=%s&bank3=%s&freq=%s&phase=%s&PC=%04X&usec=%d\n