# Secure Communication with Master and Outstation over DNP3.0 Protocol

### Introduction:

OpenDNP3 was released as open-source library. Originally OpenDNP3 was written in C++ which was a reasonable choice to write a platform independent library that needed to run efficiently on embedded Linux. Over time, .NET and Java bindings were added to integrate with more products and platforms. For our project we will be using the original C++ written OpenDNP3, the reason is that C++ is very old, complex, and error prone. Writing asynchronous C++ has always been extremely error prone. However, an extra layer of security with developer's involvement is always needed, so we use the TLS for this added layer of security.

Firstly, we will explain how the implementation of DNP3 open protocol from GitHub and TCP communication. Secondly, configuration changes according to the testbed which will be used will be explained, and in the third, the implementation of TLS – the creation of keys and certificate will be explained using the OpenSSL. In the fourth, the implementation of verifying the entity using a certificate and private key will be explained. Lastly, we will show the design and results of the implementation of the certificate and key. The evaluation of the design will show the screenshots of every step of outputs in the execution phase.

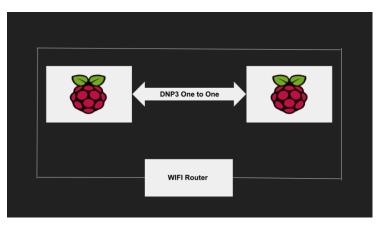
#### **Test Environment:**

We have made our environment using LINUX based operating system – Raspbian, running on Raspberry PI 3 with Wireless Area Network (WAN). Our hierarchical model consists of multiple master/outstations like architecture in SCADA systems, and Raspberry PI can run as an embedded system. The Raspberry Pi is a low cost, credit-card sized computer that can be plugged into a computer monitor or connected over SSL and Putty and uses a standard keyboard and mouse. It is compatible to manipulate the Pi using multiple libraries which are application specific and scripting languages like Python.

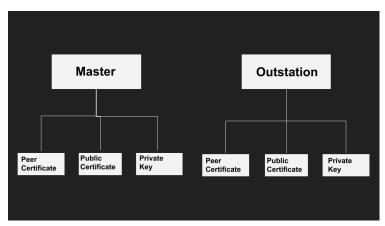


Fig 1.1: Raspberry Pi

For our implementation we have opted Opendnp3 which is a reference to DNP3 (IEEE-1815) protocol, and it is an open source from GitHub "". It uses Automatak from the open source DNP3. Firstly, we have used C++ version of DNP3 with GCC and G++ compilers and to build the DNP3 in the system we have used CMAKE. The next step after this is to use ASIO as a cross-platform for input and output communication for the Linux. First thing we did is the TCP architecture that are already available in te examples provided by the open source DNP3. TCP/IP connection is shown below:



For the next part in the architecture our plan is to generate the keys and digital certificates over TLS communication. PKI infrastructure is widely used on the internet for many applications. In our architecture we have a digital certificate and private key generated at master and slave. Each entity has a public certificate, private key, and a peer certificate to verify the file transfer.



#### **Configuration:**

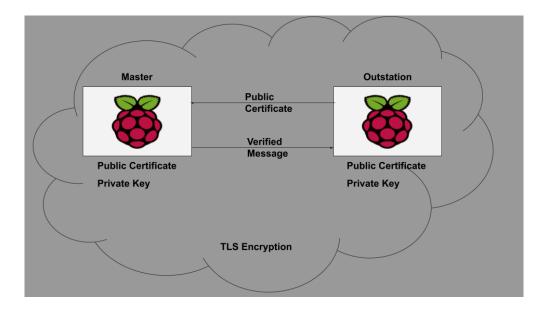
Configuration of DNP3 is the important process of changing the source code according to our implementation. Both the raspberry pi's will be connected over the same network. Firstly, the main.cpp program is manipulated by changing the IP addresses. Master's main.cpp consists of outstation's IP address and the port number. Outstation's main.cpp consists of master's IP address as the peer and outstation's IP address as the local device. The next configuration is to manipulate the TLSconfig.cpp program in the opensource OpenDNP3. The basic configuration in this is adding the path of the peer certificate, public certificate, and private key. Peer certificate for master will be Outstation's certificate and vice versa. Public certificate and private key are generated individually for each raspberry pi. Certificates and private key are stored in a folder called certificates.

#### Implementation:

Implementation of this architecture is with the use of openSSL on DNP3 Secure Authentication. The implementation is explained in the following architecture. The first part is in master's perspective, although both master and certificate have three files that are peer certificate, public certificate and private key, master has its own public certificate and private key. The peer certificate is the outstation's public certificate which is shared by outstation over wireless local area network. All these file paths are changed in the TLSConfig.cpp program in the OpenDNP3 opensource code.

After changing the code, the program is executed by CMAKE. After the make command, the demo files are created. The demo files are executed for our operation.

The second part is in outstation's perspective, which has the same architecture as master. After generation of the key's, the pem files are generated.



#### **Evaluation:**

In this section, we present the simulation results over the infrastructure. Using the architecture that we explained earlier, we show the master communicating with outstation and verifying the public certificate. After verifying, the password for the private key will be given for the communication. This TLS connection will be shown.

🐞 🌐 🛅 🗾 🔤 master	🤯 main.cpp - /home/pi/	🛜 🖘 23:30
File Edit Search View Documen	t Project Build Tools Help	
P • 🖻 • 🕌 🗊 🗌	$ \begin{array}{c c} \bullet & \bullet $	4 S A
Symbols Documents	main.cpp 🛪 TLSConfig.h 🕱	
<ul> <li>✓ Prunctions</li> <li>Ø main [32]</li> <li>✓ ⊕ Extern Variables</li> </ul>	37 return -1; 38 - } 39	rtificate> <local certificate=""> <private key="">" &lt;&lt; std::endl; •</private></local>
φ opendnp3 [30] φ std [29]	<pre>41 std::string localCertificate(argv[2]); 42 std::string privateKey(argv[3]); 43</pre>	
	44       std::cout << "Using peer cert: " << peerCertificat	cate << std::endl;
	48 // Specify what log levels to use. NORMAL is warni 49 // You can add all the comms logging by uncommenti 50 const auto logLevels = levels::NORMAL   levels::AL 51	ing below
	52 // This is the main point of interaction with the 53 // send log messages to the console 54 DNP3Manager manager(1, ConsoleLogger::Create()); 55	stack
	<pre>56 // Connect via a TCPClient socket to a outstation 57 auto channel = manager.AddTLSClient( "tls-client", logLevels, ChannelRetry::Default</pre>	t(), {IPEndpoint("192.168.1.96", 8888)}, "192.168.1.100", rivateKey), PrintingChannelListener::Create());
	<ul> <li>// The master config object for a master. The defa</li> <li>// useable, but understanding the options are impo</li> <li>MasterStackConfig stackConfig;</li> </ul>	
	<ul> <li>// you can override application layer settings for</li> <li>// in this example, we've change the application</li> <li>stackConfig.master.responseTimeout = TimeDuration:</li> <li>stackConfig.master.disableUnsolOnStartup = true;</li> </ul>	layer timeout to 2 seconds
	69       70     // You can override the default link layer setting       71     // in this example we'vUsing peer cert: e changed       72     stackConfig.link.localAddr = 1;       73     stackConfig.link.remoteAddr = 10;	
	74 75 76 76	
	77 // Create a new master on a previously declared po	vrt, with a →
Status	indentation mode for /home/pi/dnp3/cpp/examples/tls/master/main.cpp. indentation mode for /home/pi/dnp3/cpp/examples/tls/master/main.cpp.	
Compiler 23:29:57: File /home/pi/dr	<pre>interfaction mode for /nome/pi/sonps/cpp/soamples/ts/maslef/main.cpp. ng3/cpp/examples/tls/master/main.cpp opened(1). indentation mode for /home/pi/dnp3/cpp/lib/include/opendnp3/channel/TLSCC</pre>	onfig.h.
line: 37 / 155 col: 18 sel: 0 INS	SP mode: LF encoding: UTF-8 filetype: C++ scope: main	

Fig 1.2: IP Change in Master main.cpp

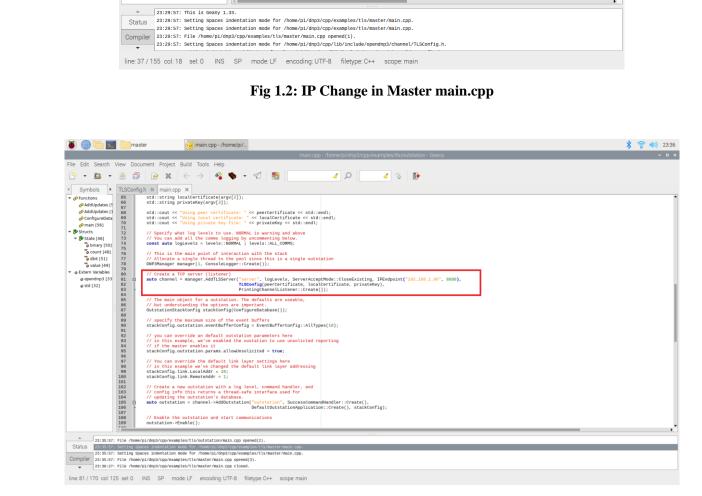


Fig 1.3: IP Change in Outstation main.cpp

🐞 🌗 🛅 🗾 🔤 master	😡 TLSConfig.h - /home/ 🔽 channel	* 🛜 📣 23:33
	TLSConfig.h - /home/pi/dnp3/cpp/lib/include/opendnp3/	íchannel - Geany 🗸 🗖 🗙
File Edit Search View Documen	t Project Build Tools Help	
🕒 • 🖻 • 🔮 🗊	$ \begin{array}{c c} \bullet & \bullet $	4 🔎 🚺 🗞 🚯
Symbols Documents	<pre>38 * provided by other party. 40 * provided by other party. 40 * genam alcolartFilePath File that contains the 41 * genam alcolartFilePath File that contains the 42 * genam alcolartFilePath File that contains the 43 * genam alcolartFilePath File that contains the 44 * genam alcolartFilePath File that contains the 44 * genam alcolartFilePath File that contains the 45 * genam cloberList Alcow TLS version 1.3 (defaul 46 * genam cloberList The opensal clober-list, defaul 47 * localCertFilePath and privateKeyFilePath can opt 49 * data. 51 * */ 52 U TLSconfig(const std::strings 53 const std::strings 54 context.std::strings 55 const std::strings 56 bool allowTLSV1 = false, 57 bool allowTLSV1 = false, 58 bool allowTLSV1 = true, 59 const std::strings cloberList, defaulty 59 const std::strings cloberList, defaulty 59 const std::strings cloberList = **) 50 const std::strings cloberList, 56 bool allowTLSV1 = false, 57 bool allowTLSV1 = true, 58 const std::strings cloberList, 59 const std::strings cloberList, 50 const std::strings cloberList, 50 const std::strings cloberList, 50 const std::strings cloberList, 51 conterFilePath(conclertFilePath), 52 const std::strings cloberList, 53 const std::strings cloberList, 54 const std::strings cloberList, 55 const std::strings cloberList, 56 allowTLSV13 = true, 57 bool allowTLSV13 = true, 58 const std::strings cloberList, 59 const std::strings cloberList, 50 const std::strings cloberList, 50 const std::strings cloberList, 51 const std::strings cloberList, 52 const std::strings cloberList, 53 const std::strings cloberList, 54 const std::strings cloberList, 55 const std::strings cloberList, 56 const std::strings cloberList, 57 const std::strings cloberList, 58 const std::strings cloberList, 59 const std::strings cloberList, 50 const std::strings cloberList, 50 const std::strings cloberList, 50 const std::strings cloberList, 51 const std::strings cloberList, 52 const std::strings cloberList, 53 const std::strings cloberList,</pre>	It false) It true) It true) ults to "" which does not modify the default cipher list tionally be the same file, i.e. a PEM that contains both pie
23:29:57: Setting Spaces i	.33. Indentation mode for /home/pi/dnp3/cpp/examples/tls/master/main.cpp. indentation mode for /home/pi/dnp3/cpp/examples/tls/master/main.cpp. p3/cpp/examples/tls/master/main.cpp opened(1).	
23:29:57: Setting Spaces i	Indentation mode for /home/pi/dnp3/cpp/lib/include/opendnp3/channel/TLSCo	
line: 1 / 101 col: 0 sel: 0 INS	SP mode: LF encoding: UTF-8 filetype: C++ scope: unknown	1

Fig 1.4: Setting TLS certs path in TLSConfig on Master Raspberry Pi

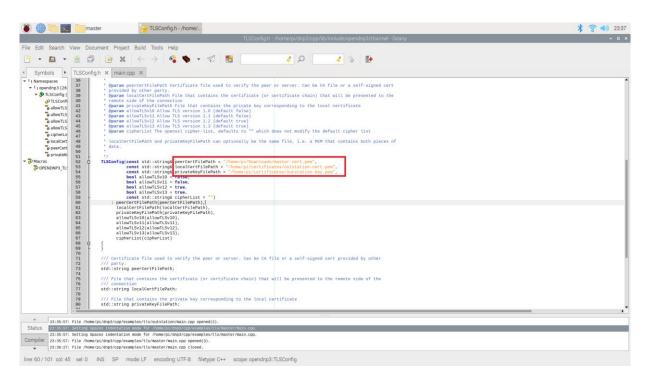


Fig 1.5: Setting TLS certs path in TLSConfig on Outstation Raspberry Pi

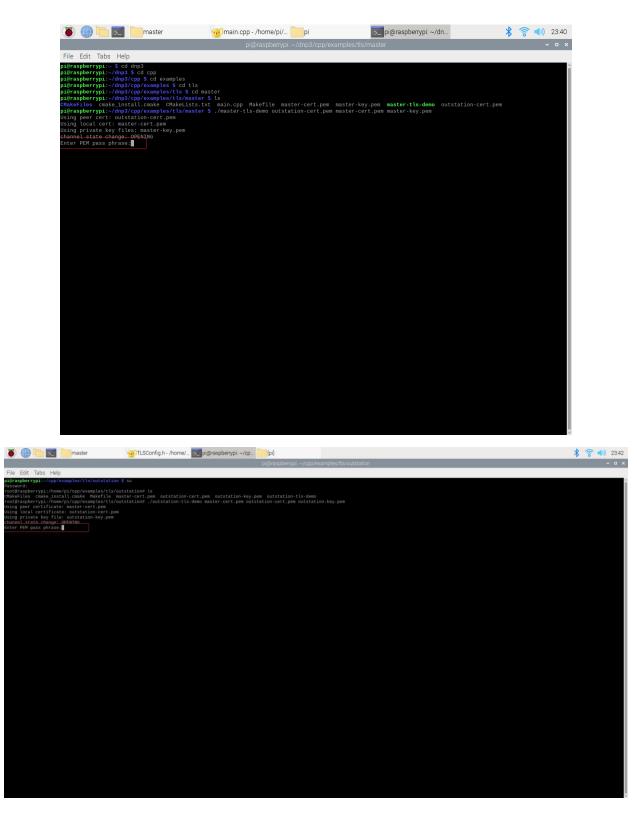


Fig 1.6: Entering cert password for master-tls-demo and outstation-tls-demo

	master	inain.cpp - /home/pi/		pi@raspberrypi: ~/dn	≱ 🛜 ◀)) 23:43
File Edit Tabs Help		pi@raspberry	pi: ~/dnp3/cpp/example	s/us/master	~ ¤ ×
Using peer cert: outst	ation-cert.pem	<pre>master \$ ./master-tls-demo outs</pre>	tation-cert.pem maste	r-cert.pem master-key.pem	
Using local cert: mast Using private key file channel state change:	s: master-key.pem				
Enter PEM pass phrase: ms(1623382980839) INFC Enter a command					
x - exits program a - performs and ad-ho	c range scan				
i - integrity demand s e - exception demand s d - diable unsolcited	can can				
r - cold restart					
ms(1623382981025) INFO ms(1623382981141) INFO channel state change:	tls-client - (	/erified certificate at depth: Connected to: 192.168.1.96, por		L/L=Mia/O=FIU/CN=outstation/emailAd	iress=pshab001@fiu.edu
ms(1623382981143) INFC ms(1623382981143)AL	-> master - Begin -> master - CO 15	ning task: Disable Unsolicited 5 3C 02 06 3C 03 06 3C 04 06			
ms(1623382981143)AL ms(1623382981144)AL ms(1623382981144)AL	-> master - FIR: -> master - 060,0 -> master - 060,0	1 FIN: 1 CON: 0 UNS: 0 SEQ: 0 002 - Class Data - Class 1 - al 003 - Class Data - Class 2 - al 004 - Class Data - Class 3 - al	FUNC: DISABLE_UNSOLIC .l objects .l objects	I I ED	
ms(1623382981144)AL ms(1623382981264) <-AL ms(1623382981264)AL	master - FIR:	1 FIN: 1 CON: 1 UNS: 1 SEQ: 0	l objects FUNC: UNSOLICITED_RES	PONSE IIN: [0x90, 0x00]	
ms(1623382981264)AL ms(1623382981379) <-AL	-> master - FIR: master - FIR:	1 FIN: 1 CON: 0 UNS: 1 SEQ: 0 1 FIN: 1 CON: 0 UNS: 0 SEQ: 0	FUNC: CONFIRM FUNC: RESPONSE IIN: [	9x90, 0x00]	
ms(1623382981380) INFO ms(1623382981380)AL ms(1623382981380)AL	<pre>master - Begir -&gt; master - C1 02 -&gt; master - FTR:</pre>	ning task: Clear Restart IIN 2 50 01 00 07 07 00 1 FIN: 1 CON: 0 UNS: 0 SEQ: 1	FUNC: WRITE		
ms(1623382981380)AL ms(1623382981431) <-AL	-> master - 080,0 master - FIR:	001 Internal Indications - Pack 1 FIN: 1 CON: 0 UNS: 0 SEQ: 1	ed Format, 8-bit star FUNC: RESPONSE IIN: [	t stop [7, 7] 9x10, 0x00]	
ms(1623382981432)AL	-> master - C2 01 -> master - FIR:	hing task: Startup Integrity Po L 3C 02 06 3C 03 06 3C 04 06 3C 1 FIN: 1 CON: 0 UNS: 0 SEQ: 2	01 06 FUNC: READ		
ms(1623382981432)AL ms(1623382981432)AL	-> master - 060,0 -> master - 060,0	002 - Class Data - Class 1 - al 003 - Class Data - Class 2 - al	l objects l objects		
ms(1623382981432)AL ms(1623382981563) <-AL	-> master - 060,0 master - FIR:	004 - Class Data - Class 3 - al 001 - Class Data - Class 0 - al 1 FIN: 1 CON: 0 UNS: 0 SEQ: 2	l objects FUNC: RESPONSE TIN: [	9x10, 0x00]	
ms(1623382981563) <-AL ms(1623382981563) <-AL	master - 001,0 master - 003,0	002 Binary Input - With Flags, 002 Double-bit Binary Input - W	8-bit start stop [0, /ith Flags, 8-bit star	9] t stop [0, 9] start stop [0, 0]	
ms(1623382981563) <-AL ms(1623382981563) <-AL ms(1623382981564) <-AL	master - 030,0 master - 030,0	102 Double-bit Binary Input - W 105 Analog Input - Single-preci 1061 Analog Input - 32-bit With 1061 Counter - 32-bit With Flag, 1061 Frozen Counter - 32-bit Wit 1062 Binary Output - 32-bit With 1061 Analog Output Status - 32-b 104 Time and Date - Endered abe 104 Time and Date - Endered abe	Flag, 8-bit start sto 8-bit start stop [0,	9]	
ms(1623382981564) <-AL ms(1623382981564) <-AL ms(1623382981564) <-AL	master - 021,0 master - 010,0	001 Frozen Counter - 32-bit Wit 002 Binary Output - Output Stat 001 Analog Output Status - 32-b	h Flag, 8-bit start s us With Flags, 8-bit it With Flag, 8-bit s	top [0, 9] start stop [0, 9] tart stop [0, 9]	
ms(1623382981564) <-AL	master 050,0	004 Time and Date - Indexed abs			
ms(1623382981564) <-AL	master - 110,0	001 Octet String - Sized by var	olute time and long i iation, 8-bit start s	hterval, 8-bit start stop [0, 9] top [0, 9]	
begin response: unsoli	master - 110,0	001 Octet String - Sized by var n: 1	olute time and long i iation, 8-bit start s	nterval, 8-bit start stop [0, 9] top [0, 9]	
ms(1623382981564) <-AL begin response: unsoli [0] : 0 : 2 : 0 [1] : 0 : 2 : 0 [2] : 0 : 2 : 0 [3] : 0 : 2 : 0	master - 110,0	001 Octet String - Sized by var n: 1	olute time and long i iation, 8-bit start s	terval, 8-bit start stop [0, 9] top [0, 9]	
begin response: unsoli [0] : 0 : 2 : 0 [1] : 0 : 2 : 0 [2] : 0 : 2 : 0	master - 110,0	001 Octet String - Sized by var in: 1	olute time and long i iation, 8-bit start s	terval, 8-bif start stop [0, 0] cop [0, 0]	
begin response: unsoli [0]: 0 : 2 : 0 [1]: 0 : 2 : 0 [2]: 0 : 2 : 0 [2]: 0 : 2 : 0 [3]: 0 : 2 : 0 [5]: 0 : 2 : 0	master - 100, cited: 0 fir: 1 fi	181 Octet String - Sized by var n: 1 	olute time and long i iation, 8-bit start s	terval, 8-bif start stop [0, 9] Cop [0, 9]	
begin response: unso 1 [0] : 0 : 2 : 0 [1] : 0 : 2 : 0 [2] : 0 : 2 : 0 [3] : 0 : 2 : 0 [5] : 0 : 2 : 0 [5] : 0 : 2 : 0 [6] master	master - 100, cited: 0 fir: 1 fi	n: 1	olute time and long i iation, 8-bit start s errypi.~/cpp/examples/fis/ou	terval, 8-bif start stop (0, 0) top (0, 0)	
begin response: unsol [0]:0:2:0 [1]:0:2:0 [2]:0:2:0 [3]:0:2:0 [3]:0:2:0 [5]:0:2:0 [5]:0:2:0 [6]:0	- master - 119,6 cited: 0 fir: 1 fi	n: 1	olute time and long i iation, 8-bit start s enym -/cpp/examples/Ms/ou	terval, 8-bif start stop (0, 0) top (0, 0)	
bogin, response: unsoli           0 : 0 : 2 : 0           1 : 0 : 2 : 0           2 : 0 : 2 : 0           3 : 0 : 2 : 0           4 : 0 : 2 : 0           5 : 0 : 2 : 0           6 : 0 : 2 : 0           1 : 0 : 2 : 0		n: 1	enypi ⇒kppicampicijis/ou	terval, 8-bif start stop (0, 0) top (0, 0)	
bogin, response: unsoli           0 : 0 : 2 : 0           1 : 0 : 2 : 0           2 : 0 : 2 : 0           3 : 0 : 2 : 0           4 : 0 : 2 : 0           5 : 0 : 2 : 0           6 : 0 : 2 : 0           1 : 0 : 2 : 0		n: 1	enypi ⇒kppicampicijis/ou	terval, 8-bif start stop [0, 9] top [0, 9]	
bogin, response: unsoli           0 : 0 : 2 : 0           1 : 0 : 2 : 0           2 : 0 : 2 : 0           3 : 0 : 2 : 0           4 : 0 : 2 : 0           5 : 0 : 2 : 0           6 : 0 : 2 : 0           1 : 0 : 2 : 0		n: 1	enypi ⇒kppicampicijis/ou	terval, 8-bif start stop [0, 9] top [0, 9]	
bogin, response: unsoli           0 : 0 : 2 : 0           1 : 0 : 2 : 0           2 : 0 : 2 : 0           3 : 0 : 2 : 0           4 : 0 : 2 : 0           5 : 0 : 2 : 0           6 : 0 : 2 : 0           1 : 0 : 2 : 0		n: 1	enypi ⇒kppicampicijis/ou	terval, 8-bif start stop (0, 0) top (0, 0)	
bogin, response: unsoli           0 : 0 : 2 : 0           1 : 0 : 2 : 0           2 : 0 : 2 : 0           3 : 0 : 2 : 0           6 : 0 : 2 : 0           6 : 0 : 2 : 0           7 : 0 : 2 : 0           6 : 0 : 2 : 0           7 : 0 : 2 : 0           9 : 0 : 2 : 0           10 : 0 : 2 : 0           11 : 0 : 2 : 0           12 : 0 : 2 : 0           13 : 0 : 2 : 0           14 : 0 : 2 : 0           15 : 0 : 2 : 0           16 : 0 : 2 : 0           17 : 0 : 0 : 1 : 0 : 2 : 0           18 : 0 : 1 : 0 : 2 : 0           19 : 0 : 2 : 0           10 : 0 : 2 : 0           10 : 0 : 2 : 0           10 : 0 : 2 : 0           10 : 0 : 2 : 0           10 : 0 : 2 : 0           10 : 0 : 2 : 0           10 : 0 : 2 : 0           10 : 0 : 2 : 0           10 : 0 : 2 : 0 : 0 : 2 : 0           10 : 0 : 0 : 0 : 0 : 0 : 0 : 0 : 0 : 0 :		n: 1	enypi ⇒kppicampicijis/ou	tstation	
bogin, response: unsoli           0 : 0 : 2 : 0           1 : 0 : 2 : 0           2 : 0 : 2 : 0           3 : 0 : 2 : 0           6 : 0 : 2 : 0           6 : 0 : 2 : 0           7 : 0 : 2 : 0           6 : 0 : 2 : 0           7 : 0 : 2 : 0           9 : 0 : 2 : 0           10 : 0 : 2 : 0           11 : 0 : 2 : 0           12 : 0 : 2 : 0           13 : 0 : 2 : 0           14 : 0 : 2 : 0           15 : 0 : 2 : 0           16 : 0 : 2 : 0           17 : 0 : 0 : 1 : 0 : 2 : 0           18 : 0 : 1 : 0 : 2 : 0           19 : 0 : 2 : 0           10 : 0 : 2 : 0           10 : 0 : 2 : 0           10 : 0 : 2 : 0           10 : 0 : 2 : 0           10 : 0 : 2 : 0           10 : 0 : 2 : 0           10 : 0 : 2 : 0           10 : 0 : 2 : 0           10 : 0 : 2 : 0 : 0 : 2 : 0           10 : 0 : 0 : 0 : 0 : 0 : 0 : 0 : 0 : 0 :		n: 1	enypi ⇒kppicampicijis/ou	terval, 8-bit start stop (0, 9) top (0, 9)	
bogin, response: unsoli           0 : 0 : 2 : 0           1 : 0 : 2 : 0           2 : 0 : 2 : 0           3 : 0 : 2 : 0           6 : 0 : 2 : 0           6 : 0 : 2 : 0           7 : 0 : 2 : 0           6 : 0 : 2 : 0           7 : 0 : 2 : 0           9 : 0 : 2 : 0           10 : 0 : 2 : 0           11 : 0 : 2 : 0           12 : 0 : 2 : 0           13 : 0 : 2 : 0           14 : 0 : 2 : 0           15 : 0 : 2 : 0           16 : 0 : 2 : 0           17 : 0 : 0 : 1 : 0 : 2 : 0           18 : 0 : 1 : 0 : 2 : 0           19 : 0 : 2 : 0           10 : 0 : 2 : 0           10 : 0 : 2 : 0           10 : 0 : 2 : 0           10 : 0 : 2 : 0           10 : 0 : 2 : 0           10 : 0 : 2 : 0           10 : 0 : 2 : 0           10 : 0 : 2 : 0           10 : 0 : 2 : 0 : 0 : 2 : 0           10 : 0 : 0 : 0 : 0 : 0 : 0 : 0 : 0 : 0 :		n: 1	enypi ⇒kppicampicijis/ou	terval, 8-bif start stop (0, 9) top (0, 9)	
begin response: unsold [0]: 0: 2: 0 [1]: 0: 2: 0 [2]: 0: 2: 0 [3]: 0: 2: 0 [3]: 0: 2: 0 [3]: 0: 2: 0 [4]: 0: 2: 0 [5]: 0: 2: 0 [6]: 0: 2: 0 [6]: 0: 2: 0 [6]: 0: 0: 0: 0: 0: 0: 0: 0: 0: 0: 0: 0: 0: 0 [6]: 0: 0: 0: 0: 0: 0: 0: 0: 0: 0 [6]: 0: 0: 0: 0: 0: 0: 0: 0: 0: 0 [6]: 0: 0: 0: 0: 0: 0: 0: 0: 0: 0: 0: 0: 0:	<pre>imaster : 110/0 cited: 0 fir: 1 fi //LSConfigh - /home/ //LSConfigh - /home/ /LSConfigh - /</pre>	n: 1 pi@raspberypi~/cp. [p] p@raspb p@raspb AfA Rest: 1 Source: 10 kerght: 5 Dest: 10 Source: 1 kerght: 5 0 House Source: 1 kerght: 5 0 House Source: 1 kerght: 5 0 House Source: 10 kerght: 6 Dest: 10 kerght: 10 kerght: 10 kerght: 10 kerght	enypi ⇒kppicampicijis/ou	terval, 8-bif start stop (0, 9) top (0, 9)	
begin response: unsoli [0]: 0: 2: 0 [1]: 0: 2: 0 [2]: 0: 2: 0 [3]: 0: 2: 0 [3]: 0: 2: 0 [5]: 0: 2: 0 [6]: 0	2011 2011	n: 1	enypi —/cpp/examples/fis/ou	terval, 8-bif start stop (0, 9) top (0, 9)	
begin response: unsoli [0]: 0: 2: 0 [1]: 0: 2: 0 [2]: 0: 2: 0 [3]: 0: 2: 0 [3]: 0: 2: 0 [5]: 0: 2: 0 [6]: 0	2011 2011	n: 1	enypi —/cpp/examples/fis/ou	Iterval, 8-bif start stop (0, 9)	
begin response: unsol [0]: 0: 2: 0 [1]: 0: 2: 0 [2]: 0: 2: 0 [3]: 0: 2: 0 [3]: 0: 2: 0 [5]: 0: 2: 0 [6]:	122/140.106:8880 (11)	n: 1	enypi —/cpp/examples/fis/ou	istation	
begin response: unsoli (0): 0: 2: 0 [1]: 0: 2: 0 [2]: 0: 2: 0 [3]: 0: 2: 0 [4]: 0 [5]: 0: 2: 0 [5]: 0: 2: 0 [6]: 0 [6	master - 010/ itted: 0 fir: 1 fi      ///////////////////////////////	n: 1	enypi —/cpp/examples/fis/ou	terval, 8-bif start stop (0, 9) top (0, 9)	
begin response: unsoli (0): 0: 2: 0 [1]: 0: 2: 0 [2]: 0: 2: 0 [3]: 0: 2: 0 [3]: 0: 2: 0 [3]: 0: 2: 0 [3]: 0: 2: 0 [4]: 0 [5]: 0: 2: 0 [5]: 0: 0: 0: 0: 0 [5]: 0: 0: 0: 0: 0: 0 [5]: 0: 0: 0: 0: 0: 0 [5]: 0: 0: 0: 0: 0: 0: 0 [5]: 0: 0: 0: 0: 0: 0: 0 [5]: 0: 0: 0: 0: 0: 0: 0: 0 [5]: 0: 0: 0: 0: 0: 0: 0: 0: 0 [5]: 0: 0: 0: 0: 0: 0: 0: 0: 0: 0 [5]: 0: 0: 0: 0: 0: 0: 0: 0: 0: 0: 0: 0: 0:	192,168.1.06:8888 192,168.1.06:8888 192,168.1.06:8888 192,168.1.06:8888 192,168.1.06:8888 192,168.1.06:8888 192,108.1.06:8888 192,108.1.08.1.88 194,108.1.88 194,	n: 1	enypi —/cpp/examples/fis/ou	tetral, 8-bif start stop (0, 9) top (0, 9)	
begin response: unsoli (0): 0: 2: 0 [1]: 0: 2: 0 [2]: 0: 2: 0 [3]: 0: 2: 0 [3]: 0: 2: 0 [3]: 0: 2: 0 [3]: 0: 2: 0 [5]: 0: 0: 2: 0 [5]: 0: 0: 2: 0 [5]: 0: 0: 2: 0 [5]: 0: 0: 0: 0: 0: 0: 0: 0: 0: 0: 0: 0: 0:	master - 110(C)     cited: 0 fir: 1 fi     file: 1 file:	n: 1  Compare the set of the set	enypi —/cpp/examples/fis/ou	tstation	
begin response: unsoli (0): 0: 2: 0 [1]: 0: 2: 0 [2]: 0: 2: 0 [3]: 0: 2: 0 [3]: 0: 2: 0 [3]: 0: 2: 0 [5]: 0: 2: 0 [6]: 0 [6]: 0: 0: 2: 0 [6]: 0: 0: 2: 0 [6]: 0: 0: 2: 0 [6]: 0: 0: 0: 0: 0: 0: 0: 0 [6]: 0: 0: 0: 0: 0: 0: 0: 0 [6]: 0: 0: 0: 0: 0: 0: 0: 0: 0 [6]: 0: 0: 0: 0: 0: 0: 0: 0: 0 [6]: 0: 0: 0: 0: 0: 0: 0: 0: 0: 0: 0 [6]: 0: 0: 0: 0: 0: 0: 0: 0: 0: 0: 0: 0: 0:	222 100 100:8880 2011.SConfig h - /home/ 2011.SConfig h - /home/ 201	n: 1	enypi —/cpp/examples/fis/ou	telation	
begin response: unsoli [0]: 0: 2: 0 [1]: 0: 2: 0 [2]: 0: 2: 0 [3]: 0: 2: 0 [3]: 0: 2: 0 [3]: 0: 2: 0 [5]: 0: 2: 0 [6]: 0: 0: 2: 0 [6]: 0: 0: 0: 0: 0: 0: 0: 0: 0: 0: 0: 0: 0: 0: 0 [6]: 0: 0: 0: 0: 0: 0: 0: 0: 0: 0 [6]: 0: 0: 0: 0: 0: 0: 0: 0: 0: 0: 0 [6]: 0: 0: 0: 0: 0: 0: 0: 0: 0: 0: 0: 0: 0:	Inster - 910/ Inster - 910/ Ited: 0 fir: 1 fi Ited: 0 fir: 0 fi Ited: 0 fir: 1 fi Ited: 0 fir: 0 fi Ited: 0 fi Ited: 0 fir: 0 fi Ited: 0 f	n: 1  Composition of the set of t	enypi —/cpp/examples/fis/ou	tetation	
begin response: unsoli (0): 0: 2: 0 [1]: 0: 2: 0 [2]: 0: 2: 0 [3]: 0: 2: 0 [3]: 0: 2: 0 [3]: 0: 2: 0 [3]: 0: 2: 0 [4]: 0 [5]: 0: 2: 0 [5]: 0: 0: 0: 0: 0 [5]: 0: 0: 0: 0: 0: 0 [5]: 0: 0: 0: 0: 0: 0 [5]: 0: 0: 0: 0: 0: 0: 0 [5]: 0: 0: 0: 0: 0: 0: 0 [5]: 0: 0: 0: 0: 0: 0: 0: 0: 0 [5]: 0: 0: 0: 0: 0: 0: 0: 0: 0: 0 [5]: 0: 0: 0: 0: 0: 0: 0: 0: 0: 0: 0: 0: 0:	master - 110(C     idea () fir: 1 fi     idea () fir: 1 fir     idea () fir     ide	n: 1  (0)  (0)  (0)  (0)  (0)  (0)  (0)  (0	enypi —/cpp/examples/fis/ou	tetation	
begin response: unsol [0]: 0: 2: 0 [1]: 0: 2: 0 [2]: 0: 2: 0 [3]: 0: 2: 0 [3]: 0: 2: 0 [5]: 0: 2: 0 [5]: 0: 2: 0 [5]: 0: 2: 0 [5]: 0: 2: 0 [6]: 0: 0: 0: 0: 0: 0: 0: 0 [6]: 0: 0: 0: 0: 0: 0: 0: 0 [6]: 0: 0: 0: 0: 0: 0: 0: 0: 0 [6]: 0: 0: 0: 0: 0: 0: 0: 0: 0 [6]: 0: 0: 0: 0: 0: 0: 0: 0: 0: 0: 0: 0: 0:	IDE 106: 1-06:888     IDE 106:888     IDE 106:88     IDE 106:8	n: 1  • • Func: UNSOLICITED_RESPONSE IIN: [0]  • • FUNC: UNSOLICITED_RESPONSE IIN: [0]  • • FUNC: UNSOLICITED_RESPONSE IIN: [0]  • • • • • • • • • • • • • • • • • • •	enypi —/cpp/examples/fis/ou	tetation	

Fig 1.7: Secure TLS connection between Master and Outstation Raspberry Pi

**Research continuity:** 

After establishing the secure communication between two raspberry pi's, data transfer from sensors is essential. Sensor data should be transferred from slave raspberry pi to master raspberry pi over secured DNP3 protocol. After sending sensor data over the raspberry pi's, the next step is to compare DNP3 with MODBUS to check for security and data integrity.

## Additional work:

- Worked on research paper(Anomaly detection in smart grid) and submitted the paper to IEEE IECON conference.
- Migrated and activated dspace license.
- Updated all the computer's hardware and software to windows 10.
- Looking into the storage devices and hard drives available in labs.