

LASC Compliance Report

Device: Eickhoff Shearer

Tester: Eickhoff Bergbautechnik


Date: 22.10.2010 10:43:11

EDS: C:/ Programme/ LASC/ LASCCompliance_1_4/ EDS/
sample_SHC.eds









Redirects: CoreRedirect.dll, version 1.04
EIPRedirect.dll, version 1.04



LASC Compliance Testing Summary








Project	Result	Skipped	Passed	Minor	Moderate	High	Critical	Total
LASC Level 1 Ver: Verified: 1.0.4		0	47	0	0	0	0	47




LASC Level 1 Suite Summary

Suite	Result	Skipped	Passed	Minor	Moderate	High	Critical	Total
Base Ethernet Communications		0	10	0	0	0	0	10
Testing Encapsulation Commands		0	13	0	0	0	0	13
Testing Session / Message Handling		0	10	0	0	0	0	10
Testing Implementation of common services		0	1	0	0	0	0	1
Testing Implementation of Core Objects		0	1	0	0	0	0	1
Testing Error Codes		0	5	0	0	0	0	5
Testing Timeouts		0	4	0	0	0	0	4
Re-test concurrent sessions		0	3	0	0	0	0	3







LASC Level 1







Test Details: Base Ethernet Communications


Test	Result	Details
4.1 Test Ping		<p>PASS: This test checks the device under test responds to a standard ping.</p> <p>If this test fails, the device may not be responding to a ping, or is not present on the network</p>
4.2 Test UDP request / response		<p>PASS: This test checks to make sure the device under test responds to a request list identity via UDP</p> <p>If this test fails, the device is not responding, or responding with an invalid response.</p>
4.3 Test TCP request / response		<p>PASS: This test checks to make sure the device under test responds to a request list identity via TCP</p> <p>If this test fails, the device is not responding, or responding with an invalid response.</p>
4.4.a Open concurrent TCP connections		<p>PASS: This test checks to make sure the device under test will allow concurrent TCP connections.</p> <p>If this test fails, 3 sessions were not able to be established.</p>
4.4.b Test concurrent TCP connections		<p>PASS: This test checks to make sure the concurrent sessions all respond as expected.</p> <p>If this test fails, the device under test is either not responding to any or all of the concurrent TCP connections.</p>
4.4.c Close concurrent TCP connections		<p>PASS: This test checks to make sure the concurrent sessions close correctly.</p> <p>If this test fails, the device under test has not closed the sessions correctly.</p>
4.5 Test Ethernet Fail 1		<p>PASS: This test checks to make sure that eip messages do not fail, if ethernet communications have been temporarily interrupted (device side)</p>

Test	Result	Details
4.6 Test Ethernet Fail 2		PASS: This test checks to make sure the eip session times out, if ethernet communications have been lost for a substantial period(device side)
4.7 Test Ethernet Fail 3		PASS: This test checks to make sure that eip messages do not fail, if ethernet communications have been temporarily interrupted (device side)
4.8 Test Ethernet Fail 4		PASS: This test checks to make sure the eip session times out, if ethernet communications have been lost for a substantial period(PC side)










Test Details: Testing Encapsulation Commands


Test	Result	Details
5.1.1 Test NOP No EIP Registered		<p>PASS: Successful if nothing received.</p> <p>If this test fails, the device has responded with a message to a NOP, under the specifications this is prohibited.</p>
5.1.2 Test NOP EIP Registered		<p>PASS: Successful if nothing received.</p> <p>The device has responded with a message to a NOP, under the specifications this is prohibited.</p>
5.2.1 Test ListIdentity TCP		<p>PASS: Successful if DUT responds with valid ListIdentity response. This message does not require an EIP session to be registered.</p> <p>The list identity packet contains items such as vendor id and ip address.</p> <p>If this test fails, check that a valid EDS has been loaded for the device.</p>
5.2.2 Test ListIdentity TCP - EIP registered		<p>PASS: Successful if DUT responds with valid ListIdentity response. This message does not require an EIP session to be registered.</p> <p>The list identity packet contains items such as vendor id and ip address.</p> <p>If this test fails, check that a valid EDS has been loaded for the device.</p>
5.2.3 Test ListIdentity UDP		<p>PASS: Successful if DUT responds with valid ListIdentity response. This message does not require an EIP session to be registered.</p> <p>The list identity packet contains items such as vendor id and ip address.</p> <p>If this test fails, check that a valid EDS has been loaded for the device.</p>
5.3.1 Test RegisterSession TCP		<p>PASS: The target shall send a RegisterSession reply to indicate that it has registered the originator.</p> <p>If this test fails, the device has either not responded with a valid session handle or not registered a valid session. The protocol version must response with 0x1 and the options must respond with 0x0.</p>

Test	Result	Details
5.3.2 Test RegisterSession TCP - Protocol 2		<p>PASS: The target shall send a RegisterSession reply to indicate that it has not registered the originator.</p> <p>Successful if error response is received with status 0x69 and the highest supported version in the protocol version field.</p> <p>If this test fails, the protocol version 2 should not yet be supported on any EIP devices, this register session test should fail with general status 0x69.</p>
5.4.1 Test UnRegisterSession - No EIP		<p>PASS: Successful if no response is received.</p> <p>If this test fails, the target may not have unregistered the session correctly, or the server is not handling session ID's correctly.</p>
5.4.2 Test UnRegisterSession - EIP		<p>PASS: Successful if no response is received.</p> <p>If this test fails, the target may not have unregistered the session correctly, or the server is not handling session ID's correctly.</p>
5.5.1 Test ListServices TCP - No EIP		<p>PASS: Successful if valid response is received from DUT. A registered EIP session is not required.</p> <p>The receiver shall reply with a standard encapsulation message, consisting of the header and data, as shown below. The data portion of the message shall provide the information on the services supported.</p> <p>Only one service class is defined, with type code 0x100 and name "Communications".</p>
5.5.2 Test ListService TCP - EIP		<p>PASS: Successful if valid response is received from DUT. A registered EIP session is not required.</p> <p>The receiver shall reply with a standard encapsulation message, consisting of the header and data, as shown below. The data portion of the message shall provide the information on the services supported.</p> <p>Only one service class is defined, with type code 0x100 and name "Communications".</p>
5.6.1 Test SendRRData - No EIP		<p>PASS: Successful if no message received from DUT</p> <p>If this test fails, the server may be responding to EIP messages without a valid session handle.</p>


Test	Result	Details
5.6.2 Test SendRRData - EIP		PASS: Successful if valid message is received from DUT. The SendRRData reply, shall contain data in response to the SendRRData request. The reply to the original encapsulated protocol request shall be contained in the data portion of the SendRRData reply.

Test Details: Testing Session / Message Handling


Test	Result	Details
6.1.1 Test Truncated message		PASS: This test checks to make sure the device under test can handle truncated messages gracefully.
6.1.2 Test Long messages (dual message)		<p>PASS: This test checks to make sure the device under test can handle long (dual) messages correctly.</p> <p>If this test fails, the device has not responded to each of the two messages as required.</p>
6.1.3 Test garbled message - short		<p>PASS: This test checks that the device can handle garbage messages, shorter in length than a valid data packet.</p> <p>If this test fails, the device has not responded correctly.</p>
6.1.4 Test garbled message - normal		<p>PASS: This test checks that the device can handle garbage messages, of equal length to a valid data packet</p> <p>If this test fails, the device has not responded correctly.</p>
6.1.5 Test garbled message - long		<p>PASS: This test checks that the device can handle garbage messages, of greater length than a valid data packet</p> <p>If this test fails, the device has not responded correctly.</p>
6.1.6 Test message flood		<p>PASS: This test checks to make sure that the device under test responds correctly to a message flood, of valid messages.</p> <p>If this test fails, the device has either responded with an error, or not responded to each of the messages that have been sent.</p>
6.1.7 Test Incorrect session identifier		<p>PASS: This test checks the devices response to incorrect session identifiers being sent.</p> <p>If this test fails, the device has not responded as expected.</p>
6.1.8 Test Incorrect sender context		<p>PASS: This test checks the devices response to incorrect sender context being sent.</p> <p>If this test fails, the device has not responded as expected.</p>
6.1.9 Test Incorrect session handle		<p>PASS: This test checks the devices response to incorrect session handler context being sent.</p> <p>If this test fails, the device has not responded as expected.</p>

Test	Result	Details
6.1.10 NOP Interference Test		<p>PASS: This test checks to make sure that NOP commands do not interfere with normal operations.</p> <p>If this test fails, NOP messages are interfering with the device under test.</p>






Test Details: Testing Implementation of common services

Test	Result	Details
7.1 Test Get Attribute Single		<p>PASS: This test checks that the device under test responds correctly to a get attribute.</p> <p>If this test fails, the device under test is either not responding to a get attribute command, or it is sending an invalid response.</p>





Test Details: Testing Implementation of Core Objects

Test	Result	Details
8.1 Test All EDS attributes		<p>PASS: This tests to make sure the device under test responds with the correct configuration details as per the supplied EDS file.</p> <p>If this test fails, the device has either not responded correctly, or responded with an invalid detail. check the EDS file supplied is correct.</p>

Test Details: Testing Error Codes

Test	Result	Details
9.1 Testing Invalid / Unsupported encapsulation command		PASS: This test sends and invalid or unsupported encapsulation command. If this test fails, the device has not responded with response 0x1.
9.2 Testing Path Destination Unknown		PASS: This test sends a message with path destination unknown. If this test fails, the device has not responded with response 0x5.
9.3 Testing Invalid Session Handle		PASS: This test sends a message with invalid session handle If this test fails, the device has not responded with response 0x64.
9.4 Testing Invalid Length Message		PASS: This test sends message with invalid length If this test fails, the device has not responded with response 0x65.
9.5 Testing Unsupported encapsulation protocol		PASS: This test sends a message with unsupported encapsulation protocol version. If this test fails, the device has not responded with response 0x69.

Test Details: Testing Timeouts

Test	Result	Details
10.1 List Identity Response / UDP		PASS: This test checks the timeout on List Identity response / UDP If this test fails, the device has not responded within 250ms
10.2 List Identity Response / TCP		PASS: This test checks the timeout on List Identity response / TCP If this test fails, the device has not responded within 250ms
10.3 Unconnected Explicit Response		PASS: This test checks the timeout on Unconnected explicit response. If this test fails, the device has not responded within 100ms
10.4 Testing Two Back to Back Explicit Message Requests		PASS: This test checks the timeout on two back to back explicit message requests. If this test fails, the device has not responded to both messages.

EDS file

```
<?xml version="1.0" encoding="UTF-8" ?>
<Device name="SHC">
  <class id="1" name="Identity Object" mandatory="1">
    <attribute id="1" name="Revision" mandatory="1">1</attribute>
    <instance id="1" name="" mandatory="1">
      <attribute id="1" name="Vendor ID" mandatory="1">0</attribute>
      <attribute id="2" name="Device Type" mandatory="1">0</attribute>
      <attribute id="3" name="Product Code" mandatory="1">0</attribute>
      <attribute id="4" name="Revision" mandatory="1">257</attribute>
      <attribute id="5" name="Status" mandatory="1">0</attribute>
      <attribute id="6" name="Serial Number" mandatory="1">1</attribute>
      <attribute id="7" name="Product Name" mandatory="1">EickShearer</attribute>
    </instance>
  </class>
  <class id="245" name="TCP/IP Interface Object" mandatory="1">
    <attribute id="1" name="Revision" mandatory="1">1</attribute>
    <instance id="1" name="" mandatory="1">
      <attribute id="1" name="Status" mandatory="1">0</attribute>
      <attribute id="2" name="Configuration Capability" mandatory="1">0</attribute>
      <attribute id="3" name="Configuration Control" mandatory="1">0</attribute>
      <attribute id="4" name="Physical Link Object" mandatory="1">0</attribute>
      <attribute id="5" name="Interface Configuration"
mandatory="1">172.16.10.66</attribute>
      <attribute id="6" name="Host Name" mandatory="1">0</attribute>
    </instance>
  </class>
  <class id="246" name="Ethernet Link Object" mandatory="1">
    <attribute id="1" name="Revision" mandatory="1">1</attribute>
    <instance id="1" name="" mandatory="1">
      <attribute id="1" name="Interface Speed" mandatory="1">100</attribute>
      <attribute id="2" name="Interface Flags" mandatory="1">3</attribute>
      <attribute id="3" name="Physical Address" mandatory="1">0</attribute>
    </instance>
  </class>
  <class id="103" name="Shearer Control System" mandatory="0">
    <attribute id="1" name="Revision" mandatory="0">1</attribute>
    <attribute id="8" name="Shearer Position" mandatory="0"></attribute>
    <attribute id="9" name="Shearer Speed" mandatory="0"></attribute>
    <attribute id="10" name="Shearer Direction" mandatory="0"></attribute>
    <attribute id="11" name="Cutter1 current" mandatory="0"></attribute>
    <attribute id="12" name="Cutter2 current" mandatory="0"></attribute>
    <attribute id="13" name="Haulage current" mandatory="0"></attribute>
    <attribute id="14" name="Shearer status" mandatory="0"></attribute>
    <instance id="1" name="" mandatory="0">
      <attribute id="1" name="Horizon Adjustment" mandatory="0"></attribute>
      <attribute id="2" name="Drum Data" mandatory="0"></attribute>
      <attribute id="3" name="Extracted Floor" mandatory="0"></attribute>
    </instance>
  </class>

```

```
    <attribute id="4" name="Floor Gradient" mandatory="0"></attribute>
  </instance>
</class>
<class id="4" name="Assembly Object" mandatory="0">
  <instance id="101" name="Horizon Adjustment" mandatory="0">
    <attribute id="3" name="Horizon Adjustment Vector" mandatory="0"></attribute>
  </instance>
  <instance id="102" name="Extracted Floor Heights" mandatory="0">
    <attribute id="3" name="Extracted Floor Vector" mandatory="0"></attribute>
  </instance>
  <instance id="103" name="Shearer Parameters" mandatory="0">
    <attribute id="3" name="Shearer Parameters" mandatory="0"></attribute>
  </instance>
  <instance id="104" name="Floor Gradient" mandatory="0">
    <attribute id="3" name="Floor Gradient Vector" mandatory="0"></attribute>
  </instance>
</class>
</Device>
```


==> Start Test 3 # 4.4.a Open concurrent TCP connections

Local Variable Filled: \$HANDLE\$ Value: 10

Timeout set: 60000 milliseconds

Running event: tcp:open with parameters:

172.16.10.66,44818,*10

Event Complete: SUCCESS

Timeout set: 60000 milliseconds

Running event: tcp:open with parameters:

172.16.10.66,44818,*11

Event Complete: SUCCESS

Timeout set: 60000 milliseconds

Running event: tcp:open with parameters:

172.16.10.66,44818,*12

Event Complete: SUCCESS

Timeout set: 60000 milliseconds

Running event: tcp:open with parameters:

172.16.10.66,44818,*13

Event Complete: SUCCESS

Timeout set: 60000 milliseconds

Running event: tcp:open with parameters:

172.16.10.66,44818,*14

Event Complete: SUCCESS

Timeout set: 60000 milliseconds

Running event: tcp:open with parameters:

172.16.10.66,44818,*15

Event Complete: SUCCESS

Timeout set: 60000 milliseconds

Running event: tcp:open with parameters:

172.16.10.66,44818,*16

Event Complete: SUCCESS

Timeout set: 60000 milliseconds

Running event: tcp:open with parameters:

172.16.10.66,44818,*17

Event Complete: SUCCESS

Timeout set: 60000 milliseconds

Running event: tcp:open with parameters:

172.16.10.66,44818,*18

Event Complete: SUCCESS

Timeout set: 60000 milliseconds

Running event: tcp:open with parameters:

172.16.10.66,44818,*19

Event Complete: SUCCESS

Timeout set: 60000 milliseconds

Running event: tcp:open with parameters:

172.16.10.66,44818,*20

Event Complete: SUCCESS

Timeout set: 60000 milliseconds

Running event: tcp:open with parameters:

172.16.10.66,44818,*21

Event Complete: SUCCESS

Timeout set: 60000 milliseconds

Running event: tcp:open with parameters:

172.16.10.66,44818,*22

Event Complete: SUCCESS

Timeout set: 60000 milliseconds

Running event: eds_check with parameters: *1
Testing: Class 1, Instance 1, Attribute 5

EDS Response OK:
expected (from EDS) 0
received 0000 (0)
Event Complete: SUCCESS
Timeout set: 300000 milliseconds
Running event: eds_check with parameters: *1
Testing: Class 1, Instance 1, Attribute 6

EDS Response OK:
expected (from EDS) 1
received 00000001 (1)
Event Complete: SUCCESS
Timeout set: 300000 milliseconds
Running event: eds_check with parameters: *1
Testing: Class 1, Instance 1, Attribute 7

EDS Response OK:
expected (from EDS) EickShearer
received EickShearer
Event Complete: SUCCESS
Timeout set: 300000 milliseconds
Running event: eds_check with parameters: *1
Testing: Class 245, Instance 0, Attribute 1

EDS Response OK:
expected (from EDS) 1
received 0001 (1)
Event Complete: SUCCESS
Timeout set: 300000 milliseconds
Running event: eds_check with parameters: *1
Testing: Class 245, Instance 1, Attribute 1

EDS Response OK:
expected (from EDS) 0
received 00000000 (0)
Event Complete: SUCCESS
Timeout set: 300000 milliseconds
Running event: eds_check with parameters: *1
Testing: Class 245, Instance 1, Attribute 2

EDS Response OK:
expected (from EDS) 0
received 00000000 (0)
Event Complete: SUCCESS
Timeout set: 300000 milliseconds
Running event: eds_check with parameters: *1
Testing: Class 245, Instance 1, Attribute 3

EDS Response OK:
expected (from EDS) 0
received 00000000 (0)
Event Complete: SUCCESS

Timeout set: 300000 milliseconds
Running event: eds_check with parameters: *1
Testing: Class 245, Instance 1, Attribute 4

EDS Response OK:
expected (from EDS) 0
received 00000000000000000000000000000000 (0)
Event Complete: SUCCESS
Timeout set: 300000 milliseconds
Running event: eds_check with parameters: *1
Testing: Class 245, Instance 1, Attribute 5

IP Address OK
Event Complete: SUCCESS
Timeout set: 300000 milliseconds
Running event: eds_check with parameters: *1
Testing: Class 245, Instance 1, Attribute 6

EDS Response OK:
expected (from EDS) 0
received 0000 (0)
Event Complete: SUCCESS
Timeout set: 300000 milliseconds
Running event: eds_check with parameters: *1
Testing: Class 246, Instance 0, Attribute 1

EDS Response OK:
expected (from EDS) 1
received 0001 (1)
Event Complete: SUCCESS
Timeout set: 300000 milliseconds
Running event: eds_check with parameters: *1
Testing: Class 246, Instance 1, Attribute 1

EDS Response OK:
expected (from EDS) 100
received 00000064 (100)
Event Complete: SUCCESS
Timeout set: 300000 milliseconds
Running event: eds_check with parameters: *1
Testing: Class 246, Instance 1, Attribute 2

EDS Response OK:
expected (from EDS) 3
received 00000003 (3)
Event Complete: SUCCESS
Timeout set: 300000 milliseconds
Running event: eds_check with parameters: *1
Testing: Class 246, Instance 1, Attribute 3

EDS Response OK:
expected (from EDS) 0
received 000000000000 (0)
Event Complete: SUCCESS
Timeout set: 300000 milliseconds

Running event: eds_check with parameters: *1
Testing: Class 103, Instance 0, Attribute 1

EDS Response OK:
expected (from EDS) 1
received 0001 (1)
Event Complete: SUCCESS
Timeout set: 300000 milliseconds
Running event: eds_check with parameters: *1
Testing: Class 103, Instance 0, Attribute 8

No value in EDS, value returned : 64000000
Event Complete: SUCCESS
Timeout set: 300000 milliseconds
Running event: eds_check with parameters: *1
Testing: Class 103, Instance 0, Attribute 9

No value in EDS, value returned : 0000
Event Complete: SUCCESS
Timeout set: 300000 milliseconds
Running event: eds_check with parameters: *1
Testing: Class 103, Instance 0, Attribute 10

No value in EDS, value returned : 00
Event Complete: SUCCESS
Timeout set: 300000 milliseconds
Running event: eds_check with parameters: *1
Testing: Class 103, Instance 0, Attribute 11

No value in EDS, value returned : 0000
Event Complete: SUCCESS
Timeout set: 300000 milliseconds
Running event: eds_check with parameters: *1
Testing: Class 103, Instance 0, Attribute 12

No value in EDS, value returned : 0000
Event Complete: SUCCESS
Timeout set: 300000 milliseconds
Running event: eds_check with parameters: *1
Testing: Class 103, Instance 0, Attribute 13

No value in EDS, value returned : 0000
Event Complete: SUCCESS
Timeout set: 300000 milliseconds
Running event: eds_check with parameters: *1
Testing: Class 103, Instance 0, Attribute 14

No value in EDS, value returned : 0200
Event Complete: SUCCESS
Timeout set: 300000 milliseconds
Running event: eds_check with parameters: *1
Testing: Class 103, Instance 1, Attribute 1

No value in EDS, value returned : 14000000000000000000
Event Complete: SUCCESS

Running event: tcp:close with parameters: *24
Event Complete: SUCCESS
Timeout set: 60000 milliseconds
Running event: tcp:close with parameters: *25
Event Complete: SUCCESS
<== End Test 3, RAN OK

<= End Suite 11

Summary

3 Tests

3 Success (100.00%)

0 Fail (0.00%)

0 Not Run (0.00%)

Suite Complete in 3846 ms