ALLSEEN ALLIANCE

AllJoyn™ Software Framework: enabling the Internet of Everything

Debugging/Testing

Narayanan Subramanian
Qualcomm Connected Experiences, Inc.

Agenda

- Challenges for testers
- Test areas for code contributors and application developers
- Unit testing
- System testing
- Debugging
- Logging
- Use of ‘sessions’ test app.
- Debugging discovery and connectivity related problems
- Popular tools

Challenges for testers

- **AllJoyn is a diverse Software framework.**
- **Implemented for different bindings: C++, Java, JS, Objective C, C, Unity, Thin Library**
- **Implemented for different platforms: Linux, Android, Windows, iOS, OSX, OpenWRT, Embedded systems**
- **Transport agnostic - Has to work for several transports viz. TCP, ICE, etc.**

Types of testing

- Testing for source code contributors:
  - Unit testing
  - System testing
  - Stress testing
  - Backward compatibility testing
- Testing for application developers:
  - Application testing
  - Backward compatibility testing

Unit testing

- Testing the APIs
- Platform independent tests
- Binding specific tests
- Tests can be built using scons command
- Any contributions pertaining to AllJoyn source code must run the unit tests
- Any new API added later must have a unit test

How to run google unit and JUnit tests

- Build google unit tests:

```
scons BINDINGS=cpp GTEST_DIR=<gtest>
```

- Run google unit tests

```
ajtest.exe
```

- Run Junit tests

```
ant test -DOS=win7 -DCPU=x86_64 -DVARIANT=debug
```

System testing

- Platform and network dependent
- Currently supports only TCP/IP transport and local transport
- Tests discovery, connectivity and message passing

Standard Library basic system test

```
bbservice.exe –n  allseen.alliance
bbclient.exe –n allseen.alliance –d –c 10
```

Thin Library basic system test

```
SampleDaemon.exe
svclite.exe
bbclient.exe –n allseen.alliance –d –c 10
```

System testing

- Bindings independent
- Mostly device to device
- Multiple devices can be simulated on a single machine
- Applications have the routing node bundled in them
- AllJoyn, now, works over loopback interface too
- To write a new C++ test code and compile:
  - Code to be placed in alljoyn_core/test
  - Add an entry in alljoyn_core/test/SConscript

Debugging

Disclaimer:

- Debugging is not limited to what we see on these slides
- Application can malfunction for an infinite number of reasons

DEBUG vs. RELEASE variant

- DEBUG variant
  - Easy to decode errors with literals
  - Used to debug/develop application
  - Logging is enabled

```
0.899 ****** ERROR NETWORK lepDisp           common/os/posix/Socket.cc:283 | Binding (sockfd
= 55) to 0.0.0.0 9955: 98 - Address already in use: ER_OS_ERROR
```

- RELEASE variant
  - It only prints error codes
  - Small in size
  - Use RELEASE variant for final application deployment
  - Status.h for error code lookup

```
 0.082 ****** ERROR NETWORK lepDisp           common/os/posix/Socket.cc:283 |  0x0004
```

DEBUG vs. RELEASE variant

- scons builds the debug variant by default
- scons … VARIANT=release
- AllJoyn SDKs are released with both the variants

AllJoyn logging

- Modular logging can be enabled
- Works only for DEBUG variant*
- “ALL” captures log from all modules (not preferred).
- Various module information can be obtained from “ALL”*
- To reduce log size, use specific module logging
- Specific module names can be found in the AllJoyn source code (e.g., Module IPNS will have macro: “IPNS”)
- Verbosity varies from 1 to 7
- A new AllJoyn module must have a new macro

```
mBus.useOSLogging(true);
mBus.setDebugLevel("ALL", 7);
mBus.setDebugLevel("ALLJOYN", 7);  //(preferred)
```

AllJoyn logging

- Logging has to be enabled and can be controlled

```
HL_DBG   ALLJOYN_DAEMON external       ...ndled/BundledDaemon.cc:257 | Using BundledDaemon
```

- But, error messages are printed by default and cannot be controlled

```
****** ERROR NETWORK external         common/os/posix/Socket.cc:249 | Connecting (sockfd = 43) to
@alljoyn : 111 - Connection refused: ER_OS_ERROR
```

- Some important logging module names:
  - All modules– *"ALL"*
  - Wi-Fi discovery – *"IPNS"*
  - AllJoyn messaging – *"ALLJOYN"*
  - Session management – *"AllJOYN_OBJ"*

sessions app.

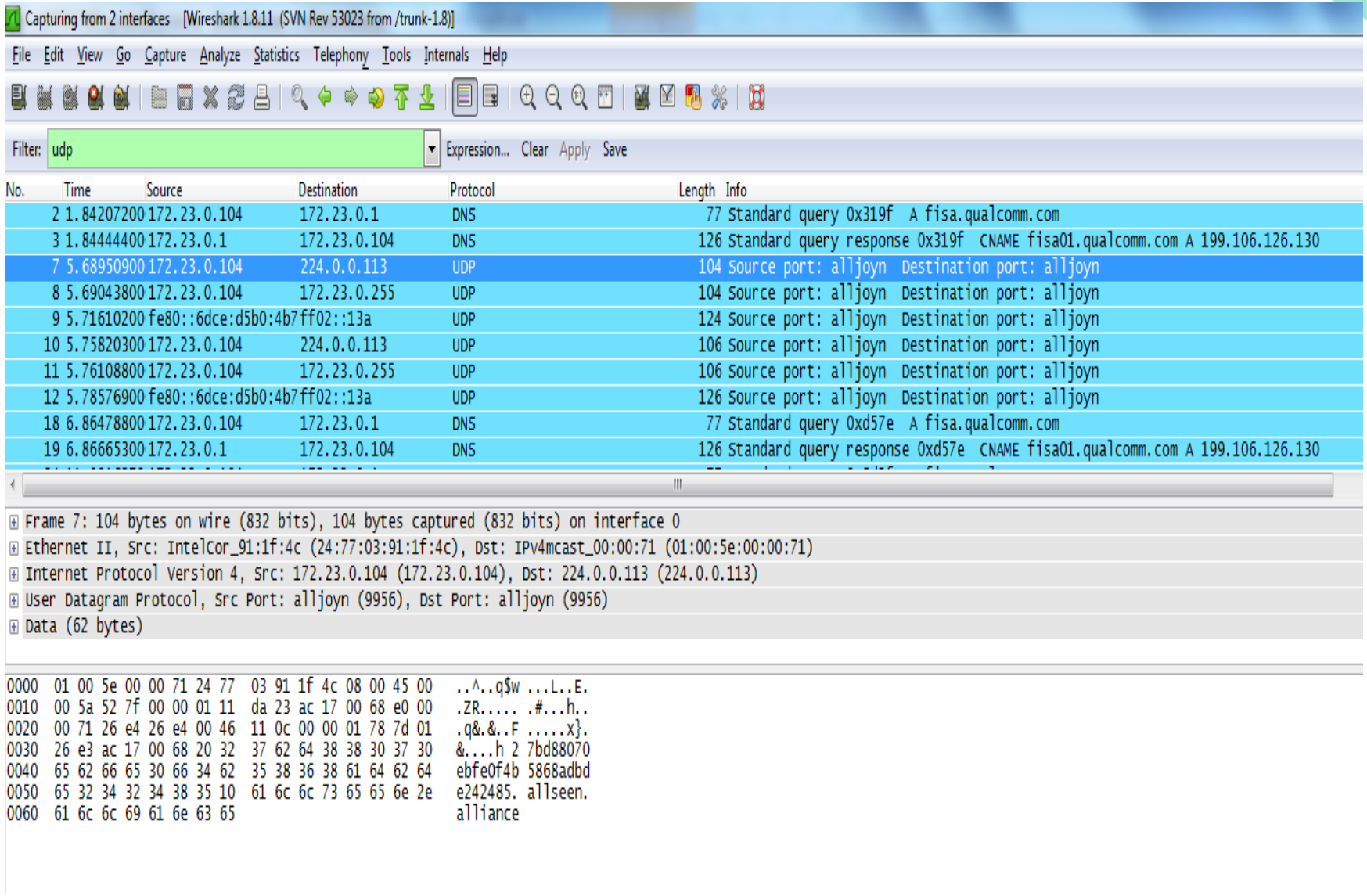
- C++ test application
- Released as part of SDK

  <AllJoyn SDK>/cpp/bin
- Can be built from source
- Useful for creating topologies on demand
- Useful debugging app.
- Tests various features of AllJoyn
- Interactive because of command line
- Easy interoperability

Debugging IP discovery issues

- set ER_DEBUG_IPNS=1
- Ensure application is sending discovery packets
- Run Wireshark and see if you can capture packets on another machine
- For every interface, three types of packets will be sent:
  - IPv4 multicast UDP packet, dst. address = 224.0.0.113
  - IPv6 multicast UDP packet, dst. address = ff02::13a
  - IPv4 subnet broadcast UDP packet, dst. address = subnet broadcast address
- Check for peer isolation on access points
- Check for firewall rules on devices

Wireshark


Capturing from 2 interfaces [Wireshark 1.8.11 (SVN Rev 53023 from /trunk-1.8)]
File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help
Filter: udp
Expression... Clear Apply Save
No. Time Source Destination Protocol Length Info
2 1.84207200 172.23.0.104 172.23.0.1 DNS 77 Standard query 0x319f A fisa.qualcomm.com
3 1.84444400 172.23.0.1 172.23.0.104 DNS 126 Standard query response 0x319f CNAME fisa01.qualcomm.com A 199.106.126.130
7 5.68950900 172.23.0.104 224.0.0.113 UDP 104 Source port: alljoyn Destination port: alljoyn
8 5.69043800 172.23.0.104 172.23.0.255 UDP 104 Source port: alljoyn Destination port: alljoyn
9 5.71610200 fe80::6dce:d5b0:4b7 ff02::13a UDP 124 Source port: alljoyn Destination port: alljoyn
10 5.75820300 172.23.0.104 224.0.0.113 UDP 106 Source port: alljoyn Destination port: alljoyn
11 5.76108800 172.23.0.104 172.23.0.255 UDP 106 Source port: alljoyn Destination port: alljoyn
12 5.78576900 fe80::6dce:d5b0:4b7 ff02::13a UDP 126 Source port: alljoyn Destination port: alljoyn
18 6.86478800 172.23.0.104 172.23.0.1 DNS 77 Standard query 0xd57e A fisa.qualcomm.com
19 6.86665300 172.23.0.1 172.23.0.104 DNS 126 Standard query response 0xd57e CNAME fisa01.qualcomm.com A 199.106.126.130
Frame 7: 104 bytes on wire (832 bits), 104 bytes captured (832 bits) on interface 0
Ethernet II, Src: IntelCor_91:1f:4c (24:77:03:91:1f:4c), Dst: IPv4mcast_00:00:71 (01:00:5e:00:00:71)
Internet Protocol Version 4, Src: 172.23.0.104 (172.23.0.104), Dst: 224.0.0.113 (224.0.0.113)
User Datagram Protocol, Src Port: alljoyn (9956), Dst Port: alljoyn (9956)
Data (62 bytes)
0000 01 00 5e 00 00 71 24 77 03 91 1f 4c 08 00 45 00 ..^..q$w ...L..E.
0010 00 5a 52 7f 00 00 01 11 da 23 ac 17 00 68 e0 00 .ZR..... .#...h..
0020 00 71 26 e4 26 e4 00 46 11 0c 00 00 01 78 7d 01 .q&.&..F .....x}.
0030 26 e3 ac 17 00 68 20 32 37 62 64 38 38 30 37 30 &....h 2 7bd88070
0040 65 62 66 65 30 66 34 62 35 38 36 38 61 64 62 64 ebfe0f4b 5868adbd
0050 65 32 34 32 34 38 35 10 61 6c 6c 73 65 65 6e 2e e242485. allseen.
0060 61 6c 6c 69 61 6e 63 65 alliance

Connectivity related issues

- Firewall blocks a connection

```
*** ERROR NETWORK JoinS-1   common/os/posix/Socket.cc:214 | Connecting (sockfd = 94) to 192.168.20.15 49277: 110 –
                Connection timed out: ER_OS_ERROR
*** ERROR TCP JoinS-1  ...aemon/TCPTransport.cc:2760 | TCPTransport::Connect(): Failed: ER_OS_ERROR
*** ERROR ALLJOYN_OBJ JoinS-1  ...e/daemon/AllJoynObj.cc:808 | trans->Connect(tcp:r4addr=192.168.20.15,r4port=49277)
                failed: ER_OS_ERROR
*** ERROR ALLJOYN lepDisp  ..._core/test/bbclient.cc:114 | JoinSession(com.a) failed:
                ER_ALLJOYN_JOINSESSION_REPLY_UNREACHABLE
```

- Connection limits exceeded

```
*** ERROR TCP     TCPTransport ...aemon\TCPTransport.cc:1658 | TCPTransport::Run(): No slot for new connection:
                ER_AUTH_FAIL
*** ERROR ALLJOYN JoinS-1  ...re/src/EndpointAuth.cc:461 | Failed to read from stream: ER_SOCK_OTHER_END_CLOSED
*** ERROR TCP     JoinS-1   ...aemon/TCPTransport.cc:2851 | TCPTransport::Connect():
                Starting the TCPEndpoint failed: ER_SOCK_OTHER_END_CLOSED
*** ERROR ALLJOYN_OBJ JoinS-1 ...e/daemon/AllJoynObj.cc:808 | trans->Connect(tcp:r4addr=192.168.20.15,r4port=9955)
                failed: ER_SOCK_OTHER_END_CLOSED
*** ERROR ALLJOYN lepDisp   ..._core/test/bbclient.cc:114 | JoinSession(com.a) failed:
                ER_ALLJOYN_JOINSESSION_REPLY_UNREACHABLE
```

- Join session on a correct name, correct port

Other useful tools

- ICMP ping
- ifconfig/ipconfig
- netstat
- Wireshark/tcpdump
- gdb
- valgrind/massif/appverif

Miscellaneous

- Always check for API return status.
- To separate routing node errors from the library errors, run the routing node component separately (e.g. alljoyn-daemon)
- Ensure that a standalone routing node (alljoyn-daemon) is configured to support untrusted leaf nodes.
- Ensure that multiple standalone routing nodes are not run on a single device by accident. This leads to routing node initialization errors.


Questions?
質問は？
Questions?
问题？
Ερωτήσεις?
Questions?
Fragen?
Вопросы?
Spørsmål?
Questions?
Domande?
Vragen?
Questions?
질문이 있습니까?
Cwestiynau?
Spørgsmål?
Frågor?
¿Preguntas?
Kysymyksiä?

Thank You!