OFConnect
An OpenSource OpenFlow Networking Library

Ramya Bolla
Deepa Karnad Dhurka

18 Nov 2014
Agenda

- About OFConnect
- Design & Architecture
- OFConnect API
- Demo Screenshots
- Get Started!
What did we create?

An OF network driver

Open source library

with true OF implementation

to aid new OF controllers and switches
What did we create?

- Documentation
- Unit Test Automation
- SDN
- OFConnect

- OpenFlow Driver
- Mul
- SDN controller
Requirements

Functional

- Dynamically link with controller or switch
- Manage OF connections
- Support OF 1.3.1, 1.3 and 1.0
- Hide wire protocol

Other

- Platform independence
- Multi-language bindings
- Full SDN Demo
Design Goals

• Performance under scale

• Controller is capable of handling 256K connections
Other Design Goals

• Generic framework to add new network protocols (TCP/UDP/TLS/IPv4/IPv6)

• A common API which works with both controller and switch
OFConnect API

cc_of_lib.h

cc_of_ret cc_of_lib_init();
cc_of_ret cc_of_lib_free();
cc_of_ret cc_of_dev_register();
cc_of_ret cc_of_dev_free();
cc_of_ret cc_of_create_channel();

cc_of_lib.h

cc_of_ret cc_of_destroy_channel();
cc_of_ret cc_of_send_pkt();
cc_of_log_toggle();
typedef int (*cc_of_recv_pkt)();
typedef int (*cc_of_accept_channel)();
typedef int (*cc_of_delete_channel)();
OF Channel Setup

Controller

cc_of_lib_init()
cc_of_dev_register()
Socket Create
Socket Bind
Socket Listen
Socket Connect Non-blocking
Socket Accept
cc_of_accept_channel()

cc_of_set_real_dpid_auxid()
cc_of_send_pkt()
cc_of_recv_pkt()

Switch

cc_of_LIB_init()
cc_of_dev_register()
Socket Create
Socket Bind
cc_of_create_channel()

OFConnect

OpenFlow Packet
OpenFlow Packet

OpenFlow Setup

cc_of_recv_pkt()
cc_of_send_pkt()
Design Choices

• Static vs Dynamic thread model
• OFHeader Processing
• Poll vs Event based socket processing
Net Services

typedef struct net_svcs_ {
    int (*open_clientfd)(cc_ofdev_key_t key, cc_ofchannel_key_t ofchann_key);
    int (*open_serverfd)(cc_ofdev_key_t key);
    int (*accept_conn)(int listenfd, cc_ofdev_key_t key);
    int (*close_conn)(int sockfd);
    ssize_t (*read_data)(int sockfd, void *buf, size_t len, int flags,
        struct sockaddr *src_addr, socklen_t *addrlen);
    ssize_t (*write_data)(int sockfd, const void *buf, size_t len, int flags,
        const struct sockaddr *dest_addr, socklen_t addrlen);
} net_svcs_t;

/* Callback Registration for TCP */
net_svcs_t tcp_sockfns = {
    tcp_open_clientfd,
    tcp_open_listenfd,
    tcp_accept,
    tcp_close,
    tcp_read,
    tcp_write,
};
Thread Management

- Poll-thread
- Util
- Thread mgr

Thread Admin Block
Elastic Pool

Block 1

Poll Thread
Util

Thread Mgr

Block 2

Poll Thread
Util

Thread Mgr

Block n
Inside A Thread

- Net Services
- Session Setup/Teardown
- Add/Del FD's Control Pipe
- Poll Loop
- Net Services Data Tx/Rx
- Pkt Tx Data Pipe
- Pkt Rx
- API Tx
- API Rx
- POLL IN/OUT
- POLL IN/OUT
- POLL IN/OUT
Test Setup

```bash
sudo mn --controller=remote, ip=127.0.0.1 --switch ovsk
```
DEMO SCREENSHOTS

<table>
<thead>
<tr>
<th>No.</th>
<th>Time</th>
<th>Source</th>
<th>Destination</th>
<th>Protocol</th>
<th>Length</th>
<th>Info</th>
</tr>
</thead>
<tbody>
<tr>
<td>70</td>
<td>214.824121000</td>
<td>127.0.0.1</td>
<td>127.0.0.1</td>
<td>OpenFlow</td>
<td></td>
<td>OFPT_HELLO</td>
</tr>
<tr>
<td>72</td>
<td>214.828774000</td>
<td>127.0.0.1</td>
<td>127.0.0.1</td>
<td>OpenFlow</td>
<td></td>
<td>OFPT_HELLO</td>
</tr>
<tr>
<td>74</td>
<td>214.828815000</td>
<td>127.0.0.1</td>
<td>127.0.0.1</td>
<td>OpenFlow</td>
<td></td>
<td>OFPT_FEATURES_REQUEST</td>
</tr>
<tr>
<td>76</td>
<td>214.831413000</td>
<td>127.0.0.1</td>
<td>127.0.0.1</td>
<td>OpenFlow</td>
<td></td>
<td>OFPT_FEATURES_REPLY</td>
</tr>
<tr>
<td>77</td>
<td>214.849630000</td>
<td>ff02::1:ff00:2</td>
<td>127.0.0.1</td>
<td>OpenFlow</td>
<td>162</td>
<td>OFPT_PACKET_IN</td>
</tr>
<tr>
<td>79</td>
<td>215.829464000</td>
<td>ff02::1:ff00:2</td>
<td>127.0.0.1</td>
<td>OpenFlow</td>
<td>174</td>
<td>OFPT_PACKET_IN</td>
</tr>
<tr>
<td>80</td>
<td>215.865572000</td>
<td>ff02::1:ff00:2</td>
<td>127.0.0.1</td>
<td>OpenFlow</td>
<td>174</td>
<td>OFPT_PACKET_IN</td>
</tr>
<tr>
<td>82</td>
<td>215.497545000</td>
<td>ff02::1:ffec:e7f8</td>
<td>127.0.0.1</td>
<td>OpenFlow</td>
<td>174</td>
<td>OFPT_PACKET_IN</td>
</tr>
<tr>
<td>84</td>
<td>215.585386000</td>
<td>ff02::1:ffec:e7f8</td>
<td>127.0.0.1</td>
<td>OpenFlow</td>
<td>174</td>
<td>OFPT_PACKET_IN</td>
</tr>
<tr>
<td>86</td>
<td>215.589517000</td>
<td>ff02::1:ffec:e7f8</td>
<td>127.0.0.1</td>
<td>OpenFlow</td>
<td>174</td>
<td>OFPT_PACKET_IN</td>
</tr>
<tr>
<td>88</td>
<td>215.850296000</td>
<td>fe80::200:ff:fe00:2</td>
<td>127.0.0.1</td>
<td>OpenFlow</td>
<td>174</td>
<td>OFPT_PACKET_IN</td>
</tr>
<tr>
<td>90</td>
<td>215.850333000</td>
<td>fe80::200:ff:fe00:2</td>
<td>127.0.0.1</td>
<td>OpenFlow</td>
<td>174</td>
<td>OFPT_PACKET_IN</td>
</tr>
<tr>
<td>92</td>
<td>215.865463000</td>
<td>fe80::200:ff:fe00:2</td>
<td>127.0.0.1</td>
<td>OpenFlow</td>
<td>174</td>
<td>OFPT_PACKET_IN</td>
</tr>
<tr>
<td>94</td>
<td>215.865492000</td>
<td>fe80::200:ff:fe00:2</td>
<td>127.0.0.1</td>
<td>OpenFlow</td>
<td>174</td>
<td>OFPT_PACKET_IN</td>
</tr>
<tr>
<td>96</td>
<td>215.149372000</td>
<td>fe80::200:ff:fe00:2</td>
<td>127.0.0.1</td>
<td>OpenFlow</td>
<td>174</td>
<td>OFPT_PACKET_IN</td>
</tr>
<tr>
<td>98</td>
<td>215.499151000</td>
<td>fe80::3429:dcff:fece:e7f8</td>
<td>127.0.0.1</td>
<td>OpenFlow</td>
<td>174</td>
<td>OFPT_PACKET_IN</td>
</tr>
<tr>
<td>100</td>
<td>215.499190000</td>
<td>fe80::3429:dcff:fece:e7f8</td>
<td>127.0.0.1</td>
<td>OpenFlow</td>
<td>174</td>
<td>OFPT_PACKET_IN</td>
</tr>
<tr>
<td>102</td>
<td>215.517476000</td>
<td>fe80::200:ff:fe00:1</td>
<td>127.0.0.1</td>
<td>OpenFlow</td>
<td>174</td>
<td>OFPT_PACKET_IN</td>
</tr>
<tr>
<td>104</td>
<td>215.377180000</td>
<td>fe80::3429:dcff:fece:e7f8</td>
<td>127.0.0.1</td>
<td>OpenFlow</td>
<td>174</td>
<td>OFPT_PACKET_IN</td>
</tr>
<tr>
<td>106</td>
<td>215.639390000</td>
<td>fe80::200:ff:fe00:1</td>
<td>127.0.0.1</td>
<td>OpenFlow</td>
<td>174</td>
<td>OFPT_PACKET_IN</td>
</tr>
<tr>
<td>110</td>
<td>215.624580000</td>
<td>fe80::200:ff:fe00:1</td>
<td>127.0.0.1</td>
<td>OpenFlow</td>
<td>174</td>
<td>OFPT_PACKET_IN</td>
</tr>
<tr>
<td>112</td>
<td>215.861742000</td>
<td>fe80::200:ff:fe00:1</td>
<td>127.0.0.1</td>
<td>OpenFlow</td>
<td>174</td>
<td>OFPT_PACKET_IN</td>
</tr>
<tr>
<td>114</td>
<td>215.802190000</td>
<td>fe80::200:ff:fe00:1</td>
<td>127.0.0.1</td>
<td>OpenFlow</td>
<td>174</td>
<td>OFPT_PACKET_IN</td>
</tr>
<tr>
<td>116</td>
<td>215.501372000</td>
<td>fe80::3429:dcff:fece:e7f8</td>
<td>127.0.0.1</td>
<td>OpenFlow</td>
<td>174</td>
<td>OFPT_PACKET_IN</td>
</tr>
<tr>
<td>118</td>
<td>215.611524000</td>
<td>00:00:00:00:00:01</td>
<td>Broadcast</td>
<td>OpenFlow</td>
<td>126</td>
<td>OFPT_PACKET_IN</td>
</tr>
<tr>
<td>120</td>
<td>215.610560000</td>
<td>00:00:00:00:00:01</td>
<td>Broadcast</td>
<td>OpenFlow</td>
<td>126</td>
<td>OFPT_PACKET_IN</td>
</tr>
<tr>
<td>122</td>
<td>215.610810000</td>
<td>00:00:00:00:00:01</td>
<td>Broadcast</td>
<td>OpenFlow</td>
<td>126</td>
<td>OFPT_PACKET_IN</td>
</tr>
<tr>
<td>124</td>
<td>215.622339000</td>
<td>00:00:00:00:00:01</td>
<td>Broadcast</td>
<td>OpenFlow</td>
<td>126</td>
<td>OFPT_PACKET_IN</td>
</tr>
<tr>
<td>126</td>
<td>215.622267000</td>
<td>00:00:00:00:00:01</td>
<td>Broadcast</td>
<td>OpenFlow</td>
<td>126</td>
<td>OFPT_PACKET_IN</td>
</tr>
<tr>
<td>128</td>
<td>215.622532000</td>
<td>00:00:00:00:00:01</td>
<td>Broadcast</td>
<td>OpenFlow</td>
<td>126</td>
<td>OFPT_PACKET_IN</td>
</tr>
<tr>
<td>130</td>
<td>215.622533000</td>
<td>00:00:00:00:00:01</td>
<td>Broadcast</td>
<td>OpenFlow</td>
<td>126</td>
<td>OFPT_PACKET_IN</td>
</tr>
<tr>
<td>132</td>
<td>215.622534000</td>
<td>00:00:00:00:00:01</td>
<td>Broadcast</td>
<td>OpenFlow</td>
<td>126</td>
<td>OFPT_PACKET_IN</td>
</tr>
<tr>
<td>134</td>
<td>215.622535000</td>
<td>00:00:00:00:00:01</td>
<td>Broadcast</td>
<td>OpenFlow</td>
<td>126</td>
<td>OFPT_PACKET_IN</td>
</tr>
<tr>
<td>136</td>
<td>215.622536000</td>
<td>00:00:00:00:00:01</td>
<td>Broadcast</td>
<td>OpenFlow</td>
<td>126</td>
<td>OFPT_PACKET_IN</td>
</tr>
</tbody>
</table>
2014/11/17 18:14:43 MUL-CONTROLLER: (c_thread_tx) Sending packet to library
2014/11/17 18:14:43 MUL-CONTROLLER: (of_switch_recv_msg) OF MSG RX TYPE (0) ——> HELLO
2014/11/17 18:14:43 MUL-CONTROLLER: (of_switch_recv_msg) OF MSG RX TYPE (6) ——> FEATURES_REQUEST
2014/11/17 18:14:43 MUL-CONTROLLER: dpid:1 being looked up is_dummy:1
2014/11/17 18:14:43 MUL-CONTROLLER: (of_recv_features_reply) real dpid sent to library
2014/11/17 18:14:43 MUL-CONTROLLER: (of_recv_features_reply) switch is being registered
2014/11/17 18:14:48 MUL-CONTROLLER: (of_switch_recv_msg) OF MSG RX TYPE (2) ——> ECHO_REQUEST
2014/11/17 18:14:44 MUL-CONTROLLER: (of_switch_recv_msg) OF MSG RX TYPE (10) ——> PACKET_IN
2014/11/17 18:14:44 MUL-CONTROLLER: (of_dfl_fwd) sw->dpid:1
Learnings

- Platform Independence
  - GLib

- Language Bindings
  - SWIG

- Controller to Integrate
  - MUL

- GIT Workflow
  - Upstream/Master/Dev
Next Steps

- Enhance TLS
  Greater OF Awareness

- Scale
  1K switches per controller
  256K connections per controller

- Perform Profile and Benchmark
Summary

• OFConnect: OF Network Library

• Open Source GPL v2

• Join and Contribute
Repos

  - GPL v2

- [https://github.com/codechix/mul-OFconnect](https://github.com/codechix/mul-OFconnect)
  - Private