
CTONE


CTONE

×


NTUST
CONNECTIVITY LAB

CTONE

O-RAN的資安威脅，以Near-RT RIC為例

CVE-2023-40997 & 40998
CVE-2023-41627

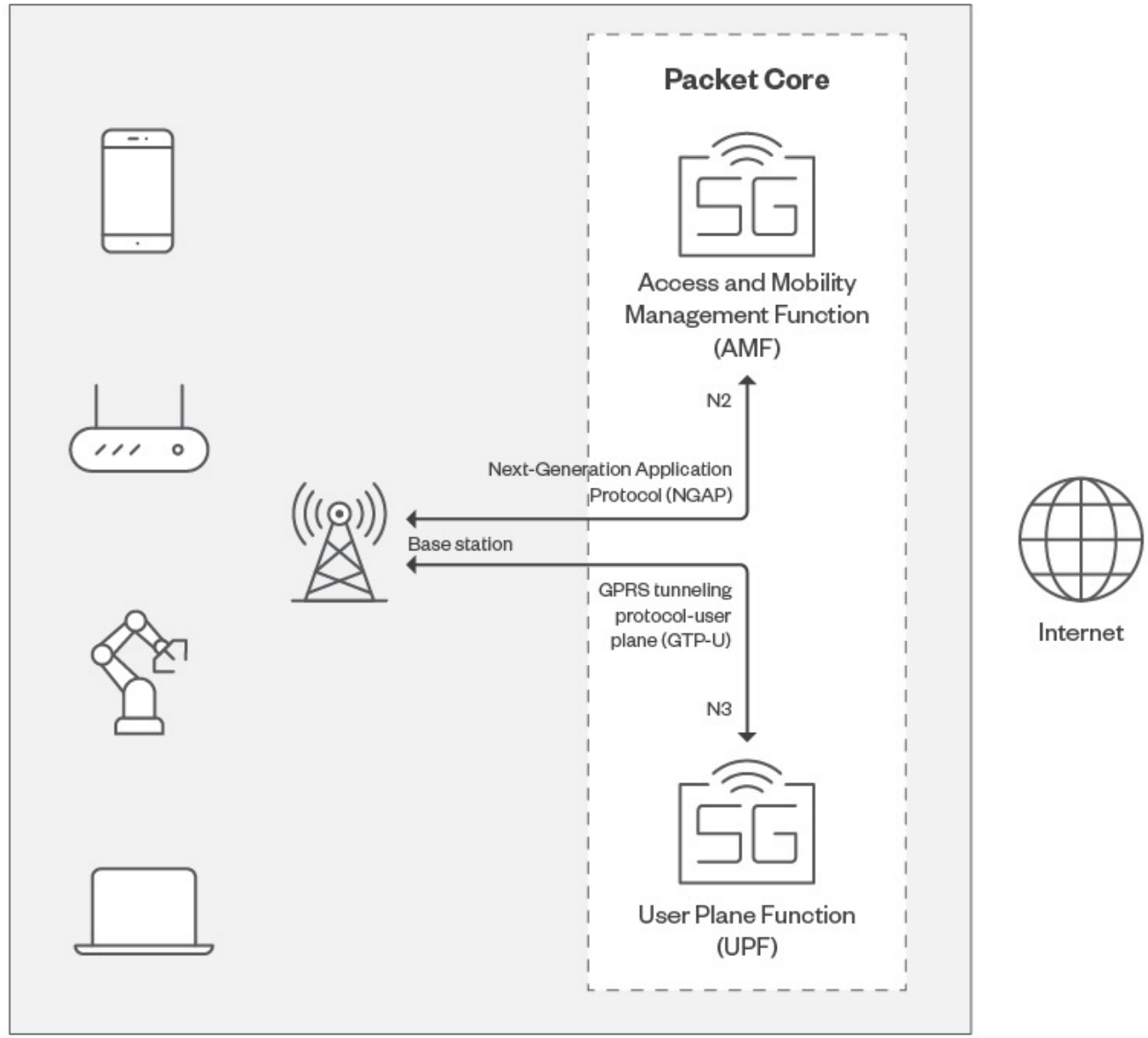
Richard Y Lin
May 16, 2024

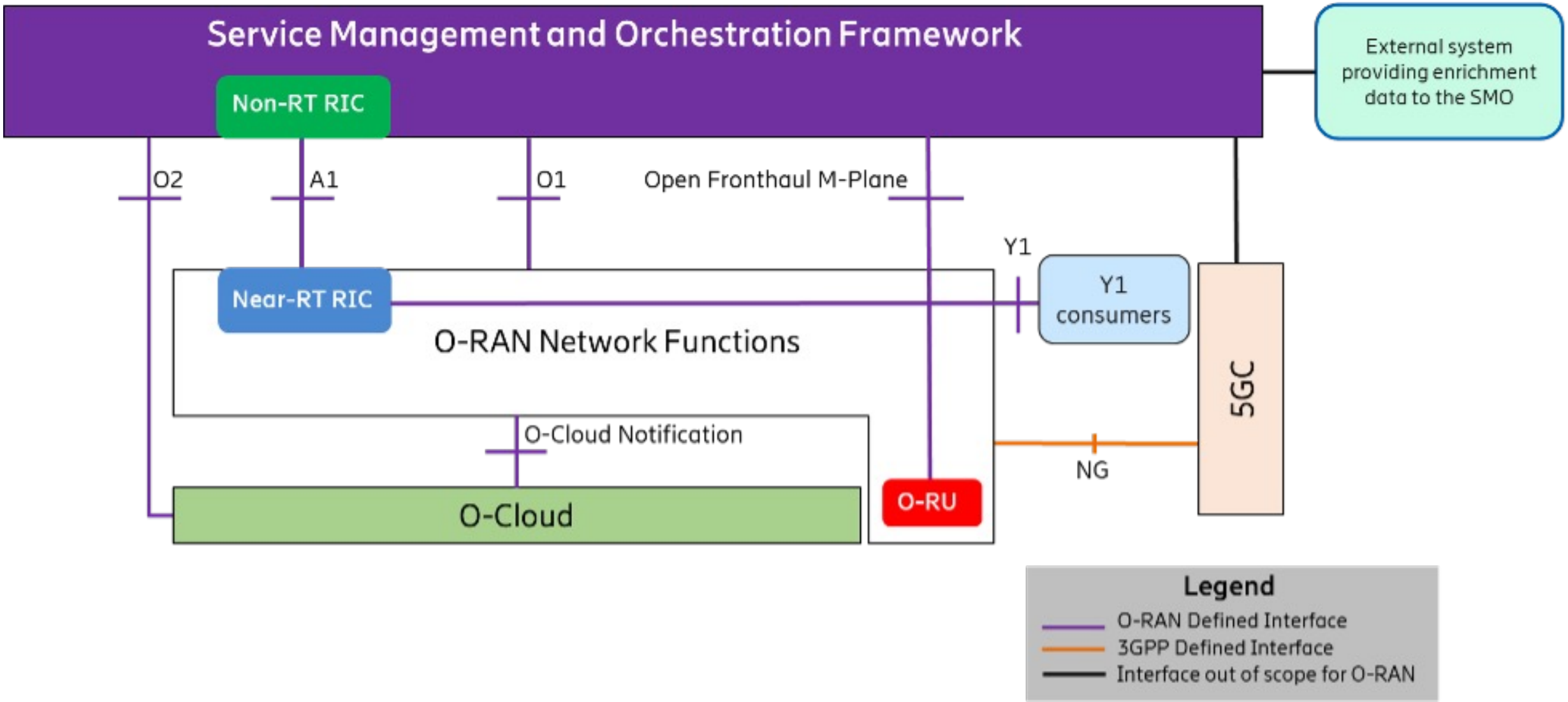
Agenda

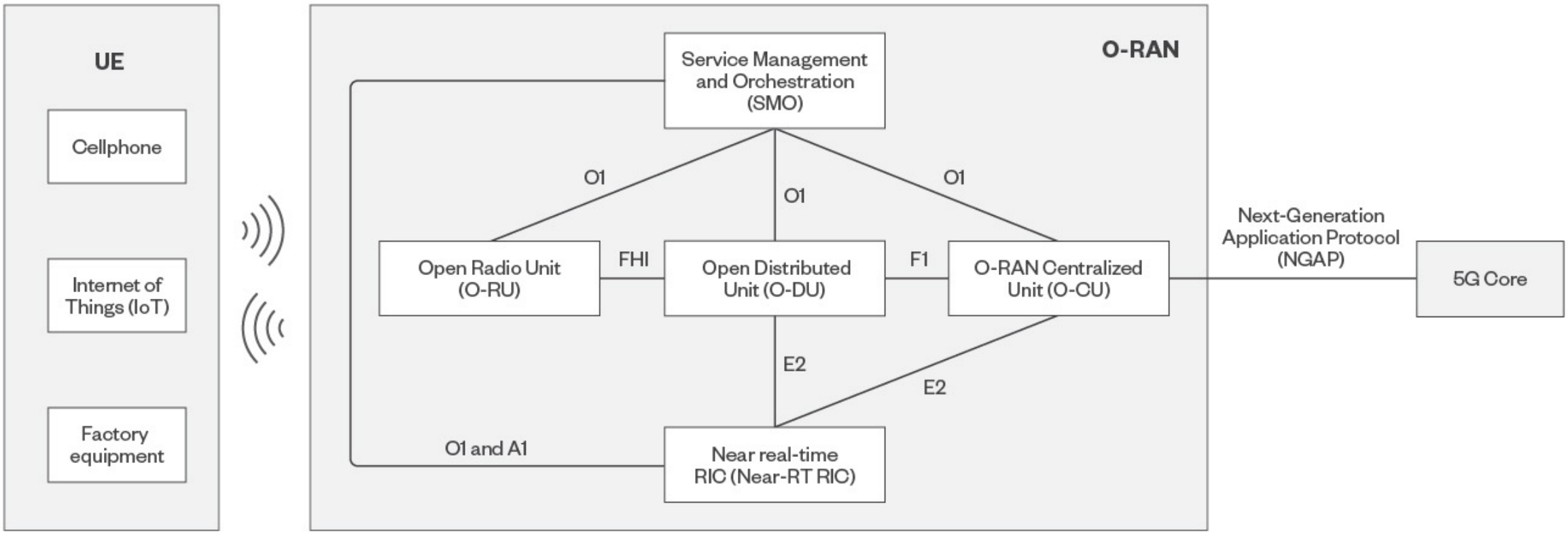
- 5G/O-RAN Architecture
- Near-RT RIC & xApps
- What is RMR
- Three Vulnerabilities in Near-RT RIC
- How was it discovered

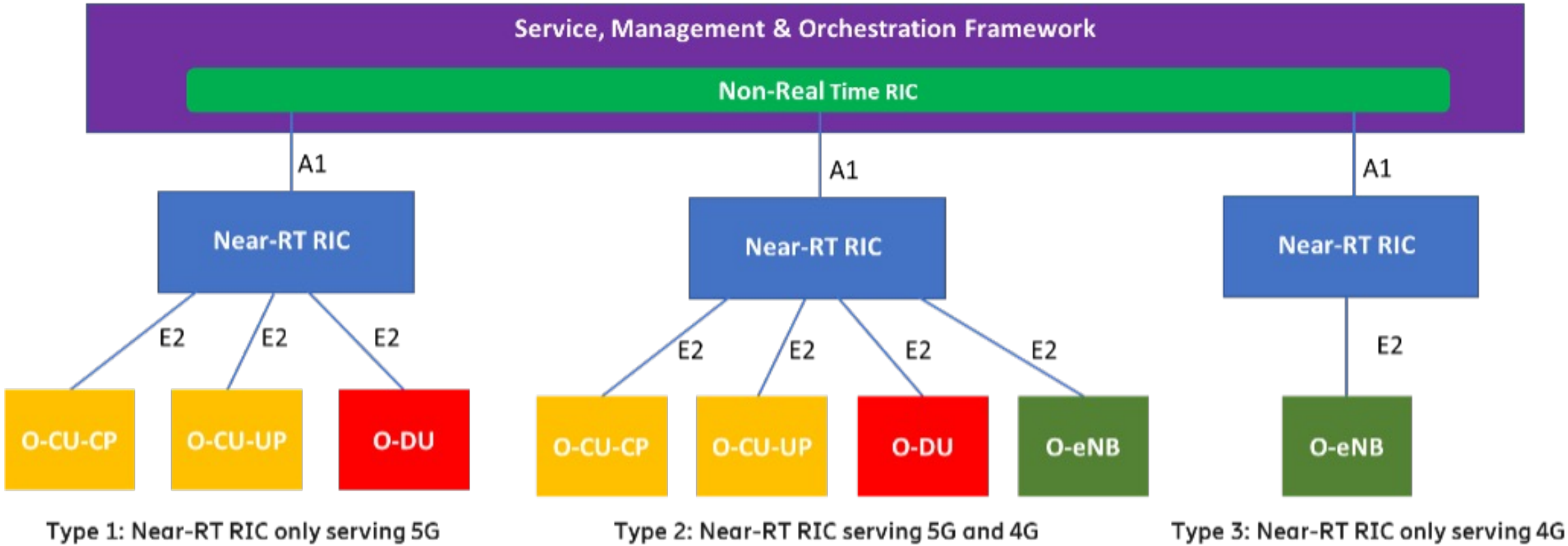
O-RAN Architecture

SMO / Near-RT RIC / RAN




Service Management and Orchestration Framework
Non-RT RIC
External system providing enrichment data to the SMO
O2
A1
O1
Open Fronthaul M-Plane
Y1
Y1 consumers
Near-RT RIC
O-RAN Network Functions
5GC
O-Cloud Notification
NG
O-Cloud
O-RU
Legend
O-RAN Defined Interface
3GPP Defined Interface
Interface out of scope for O-RAN

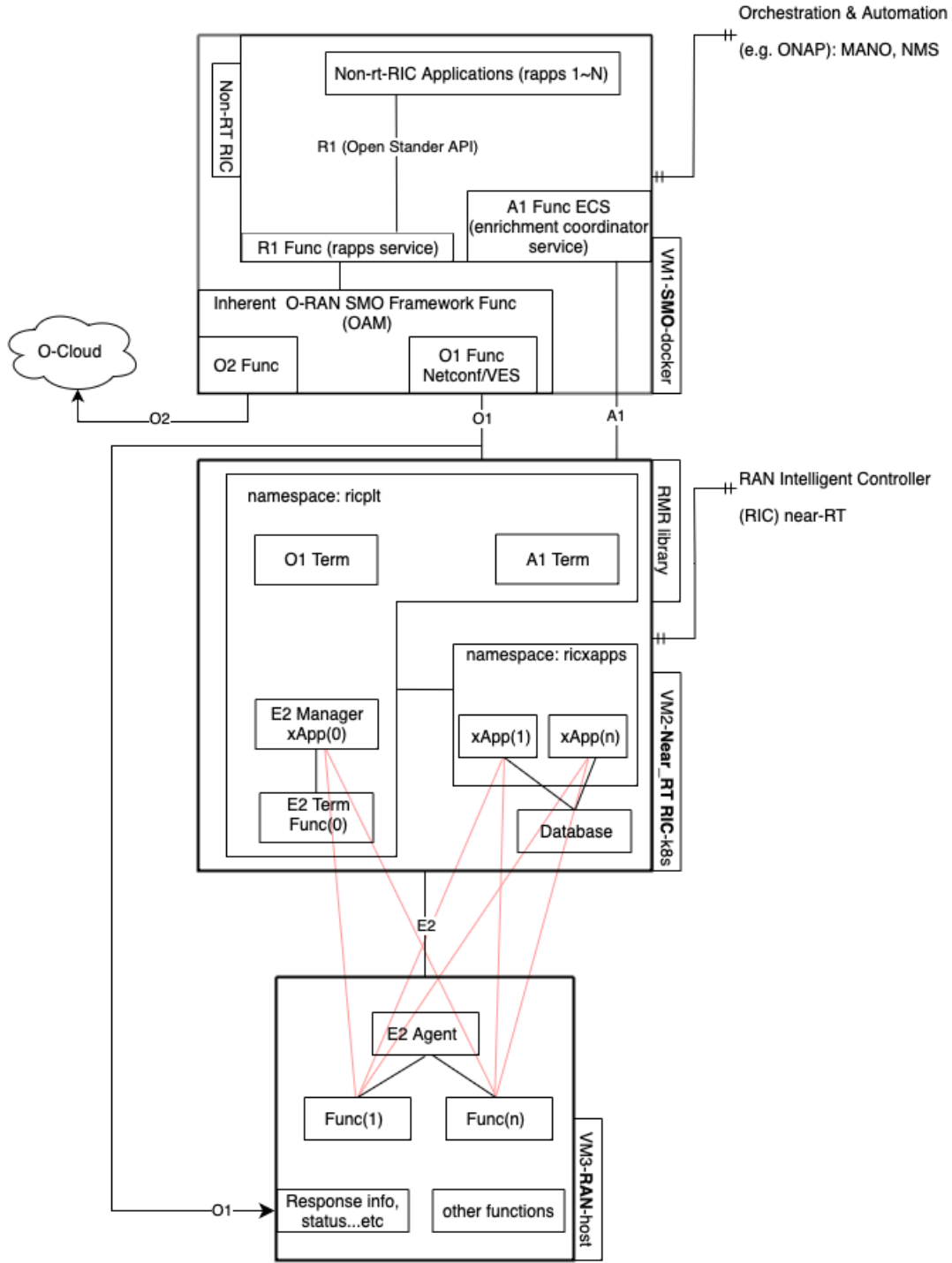




Type 1: Near-RT RIC only serving 5G

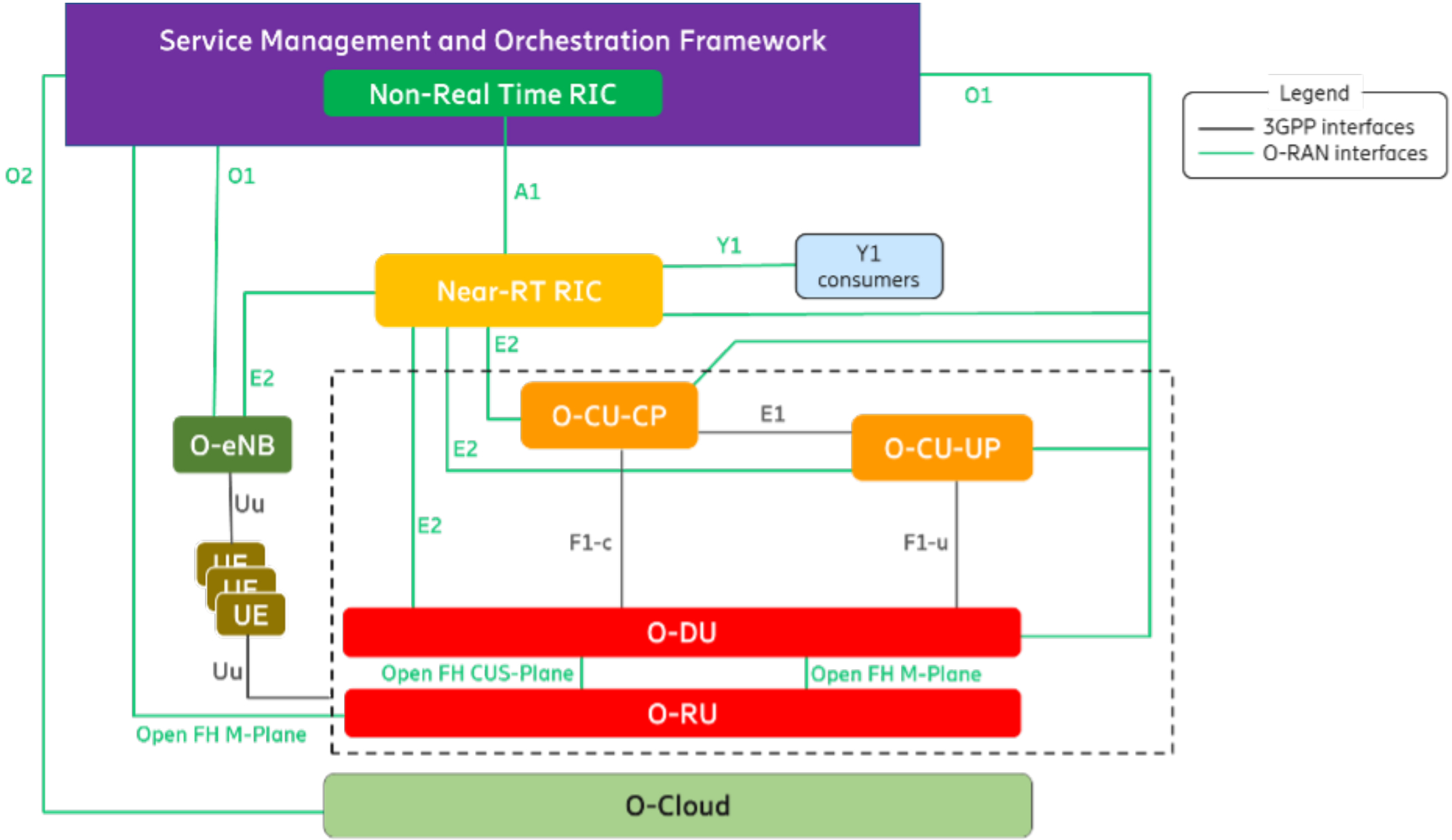
Type 2: Near-RT RIC serving 5G and 4G

Type 3: Near-RT RIC only serving 4G


Orchestration & Automation
(e.g. ONAP): MANO, NMS
Non-RT RIC
Non-rt-RIC Applications (rapps 1~N)
R1 (Open Stander API)
A1 Func ECS
(enrichment coordinator service)
R1 Func (rapps service)
VM1-SMO-docker
Inherent O-RAN SMO Framework Func (OAM)
O-Cloud
O2 Func
O1 Func Netconf/VES
O2
O1
A1
RAN Intelligent Controller
(RIC) near-RT
RMR library
namespace: ricplt
O1 Term
A1 Term
namespace: ricxapps
E2 Manager xApp(0)
xApp(1)
xApp(n)
E2 Term Func(0)
Database
VM2-Near_RT RIC-k8s
E2
E2 Agent
Func(1)
Func(n)
VM3-RAN-host
O1
Response info, status...etc
other functions

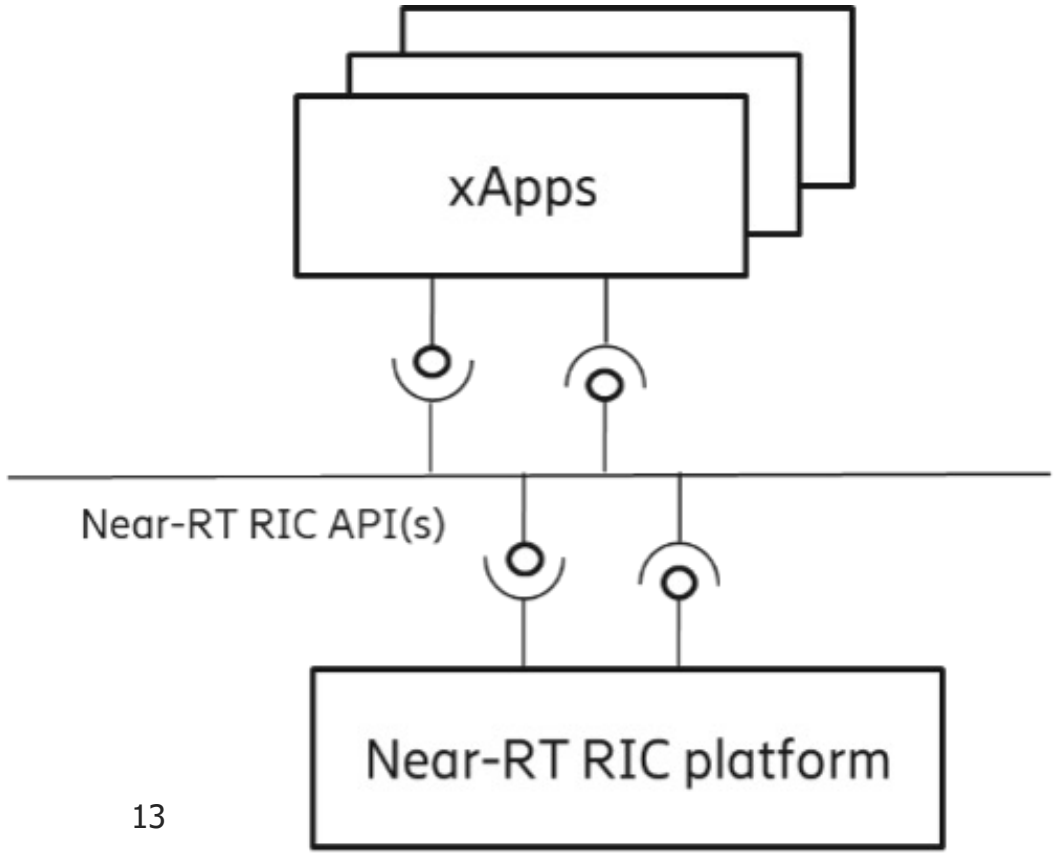
Near-RT RIC & xApps

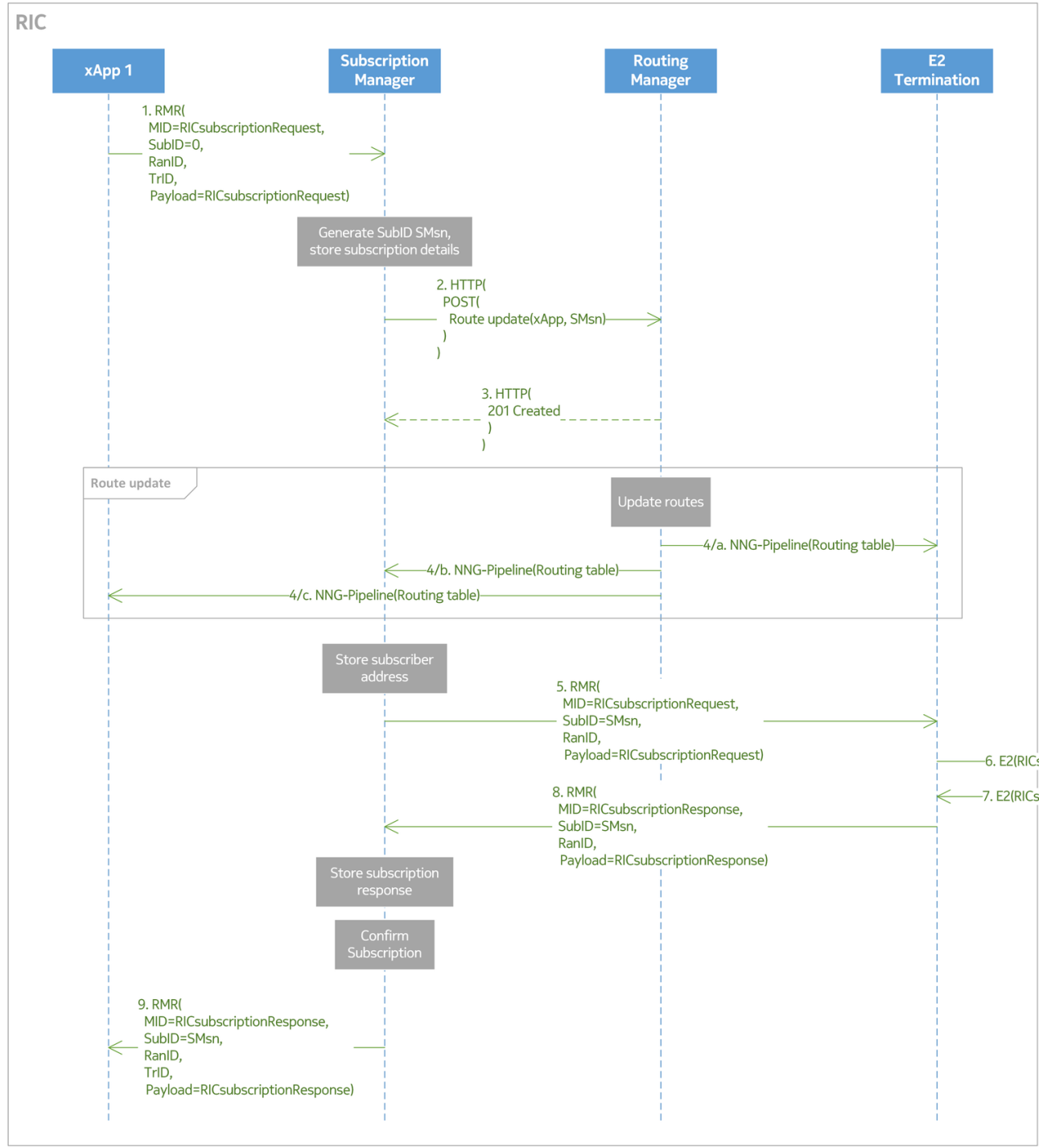
How it works


Service Management and Orchestration Framework
Non-Real Time RIC
O1
O2
O1
A1
Y1
Y1 consumers
Near-RT RIC
E2
E2
O-CU-CP
E1
O-CU-UP
O-eNB
E2
Uu
E2
F1-c
F1-u
UE
UE
UE
O-DU
Uu
Open FH CUS-Plane
Open FH M-Plane
O-RU
Open FH M-Plane
O-Cloud
Legend
3GPP interfaces
O-RAN interfaces

Near-RT RIC Application

xApp: An application designed to run on the Near-RT RIC. Such an application is likely to consist of one or more microservices and at the point of on-boarding will identify which data it consumes and which data it provides. The application is independent of the Near-RT RIC and may be provided by any third party. The E2 enables a direct association between the xApp and the RAN functionality [19].




RIC
xApp 1
Subscription Manager
Routing Manager
E2 Termination
gNodeB/ eNodeB
1. RMR(MID=RICsubscriptionRequest, SubID=0, RanID, TrID, Payload=RICsubscriptionRequest)
Generate SubID SMsn, store subscription details
2. HTTP(POST(Route update(xApp, SMsn)))
3. HTTP(201 Created))
Route update
Update routes
4/a. NNG-Pipeline(Routing table)
4/b. NNG-Pipeline(Routing table)
4/c. NNG-Pipeline(Routing table)
Store subscriber address
5. RMR(MID=RICsubscriptionRequest, SubID=SMsn, RanID, Payload=RICsubscriptionRequest)
6. E2(RICsubscriptionRequest)
7. E2(RICsubscriptionResponse)
8. RMR(MID=RICsubscriptionResponse, SubID=SMsn, RanID, Payload=RICsubscriptionResponse)
Store subscription response
Confirm Subscription
9. RMR(MID=RICsubscriptionResponse, SubID=SMsn, RanID, TrID, Payload=RICsubscriptionResponse)

Near-RT RIC Application

- 6.2.7 Interface Termination
- 6.2.7.1 E2 Termination

This functionality enables termination of E2 interface with the following:

- Terminating SCTP connection from each E2 Node;
- Routing messages from xApps through the SCTP connection to an E2 Node;
- Decoding the payload of an incoming ASN.1 message enough to determine message type;
- Handling incoming E2 messages related to E2 connectivity;
- Receiving and responding to the E2 Setup Request from an E2 Node;
- Notifying xApps of the list of RAN functions supported by an E2 Node based on information derived from the E2 Setup and RIC Service Update procedures [3];
- Notifying the newly connected E2 Node of the list of accepted functions.

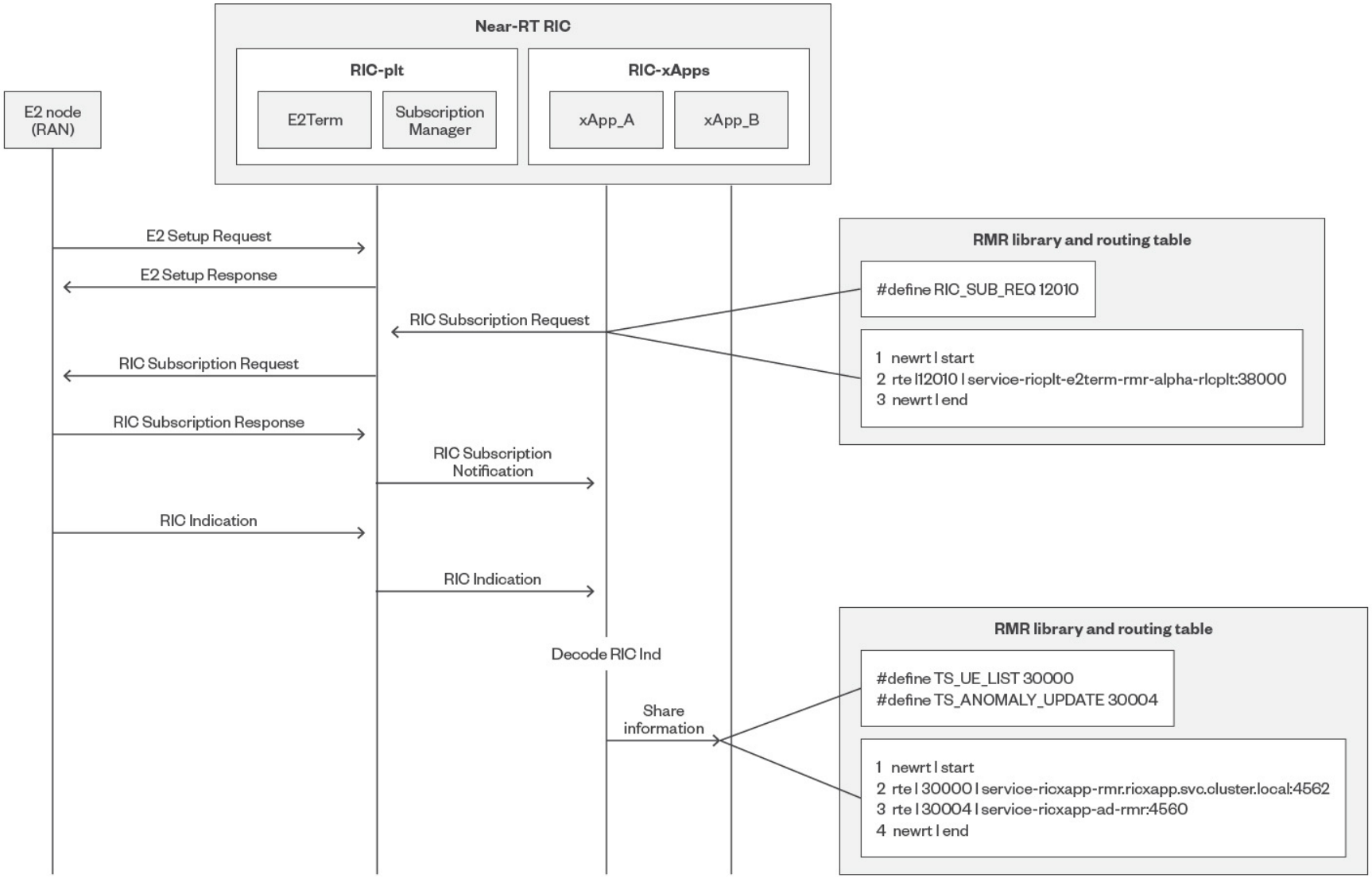
What is RMR

Messaging infrastructure

Messaging Infrastructure

- It supports registration/discovery/deletion of endpoints
- It provides the following APIs:
  - An API for sending messages to the messaging infrastructure.
  - An API for receiving messages from the messaging infrastructure.
- It supports multiple messaging modes, e.g., point-to-point mode (e.g., message exchange among endpoints), publish/subscribe mode (e.g., real-time data dispatching from E2 termination to multiple subscriber xApps).
- It provides message routing, namely according to the message routing information, messages can be dispatched to different endpoints.
- It supports message robustness to avoid data loss during a messaging infrastructure outage/restart or to release resources from the messaging infrastructure once a message is outdated.

RMR Work Process



CTONE

RMR Table Format

[ric-app/ts.git] / routes.txt

```
1 newrt|start
2 rte|20011|service-ricplt-a1mediator-rmr:10000
3 rte|30000|service-ricxapp-qp.ricxapp.svc.cluster.local:4562
4 rte|30004|service-ricxapp-ad-rmr:4560
5 newrt|end
```

"ric_app_ts"

[ric-app/qp.git] / tests / fixtures / local.rt

```
1 # static route table to direct messages sent by mock QP xApp
2 newrt|start
3 rte|30002|service-ricxapp-trafficxapp-rmr.ricxapp.svc.cluster.local:4560
4 newrt|end
```

"ric_app_qp"

O-RAN SC gerrit

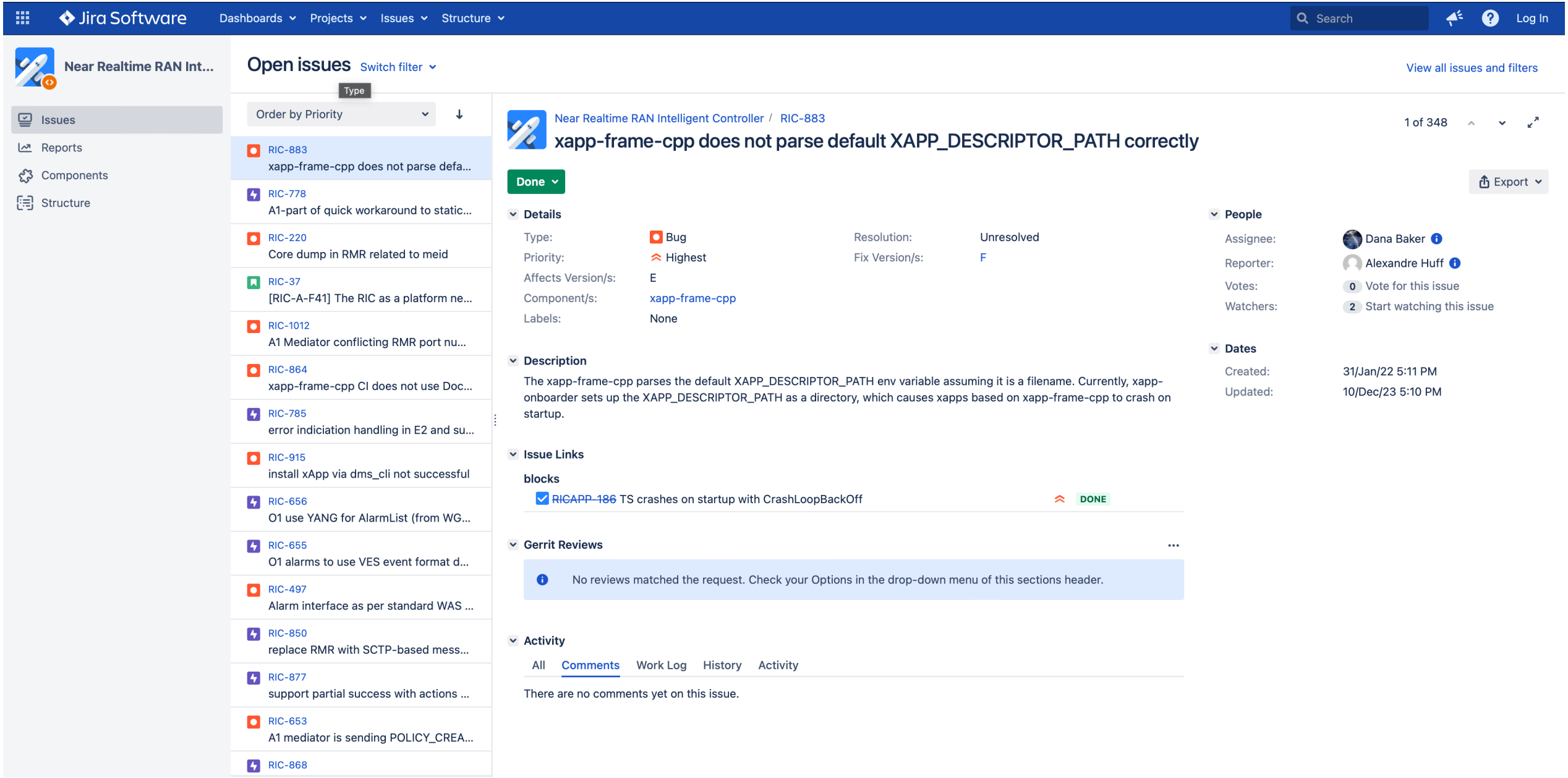
CTONE

Vulnerability in Near-RT RIC

The first publicly disclosed CVE on O-RAN

CVE-2023-40997 & 40998

CTONE


Jira Software
Dashboards
Projects
Issues
Structure
Search
Log In
Near Realtime RAN Int...
Issues
Reports
Components
Structure
Open issues
Switch filter
View all issues and filters
Type
Order by Priority
RIC-883
xapp-frame-cpp does not parse defa...
RIC-778
A1-part of quick workaround to static...
RIC-220
Core dump in RMR related to meid
RIC-37
[RIC-A-F41] The RIC as a platform ne...
RIC-1012
A1 Mediator conflicting RMR port nu...
RIC-864
xapp-frame-cpp CI does not use Doc...
RIC-785
error indiciation handling in E2 and su...
RIC-915
install xApp via dms_cli not successful
RIC-656
O1 use YANG for AlarmList (from WG...
RIC-655
O1 alarms to use VES event format d...
RIC-497
Alarm interface as per standard WAS ...
RIC-850
replace RMR with SCTP-based mess...
RIC-877
support partial success with actions ...
RIC-653
A1 mediator is sending POLICY_CREA...
RIC-868
Near Realtime RAN Intelligent Controller / RIC-883
1 of 348
xapp-frame-cpp does not parse default XAPP_DESCRIPTOR_PATH correctly
Done
Export
Details
Type: Bug
Priority: Highest
Affects Version/s: E
Component/s: xapp-frame-cpp
Labels: None
Resolution: Unresolved
Fix Version/s: F
People
Assignee: Dana Baker
Reporter: Alexandre Huff
Votes: 0 Vote for this issue
Watchers: 2 Start watching this issue
Dates
Created: 31/Jan/22 5:11 PM
Updated: 10/Dec/23 5:10 PM
Description
The xapp-frame-cpp parses the default XAPP_DESCRIPTOR_PATH env variable assuming it is a filename. Currently, xapp-onboarder sets up the XAPP_DESCRIPTOR_PATH as a directory, which causes xapps based on xapp-frame-cpp to crash on startup.
Issue Links
blocks
RICAPP-186 TS crashes on startup with CrashLoopBackOff
DONE
Gerrit Reviews
No reviews matched the request. Check your Options in the drop-down menu of this sections header.
Activity
All
Comments
Work Log
History
Activity
There are no comments yet on this issue.

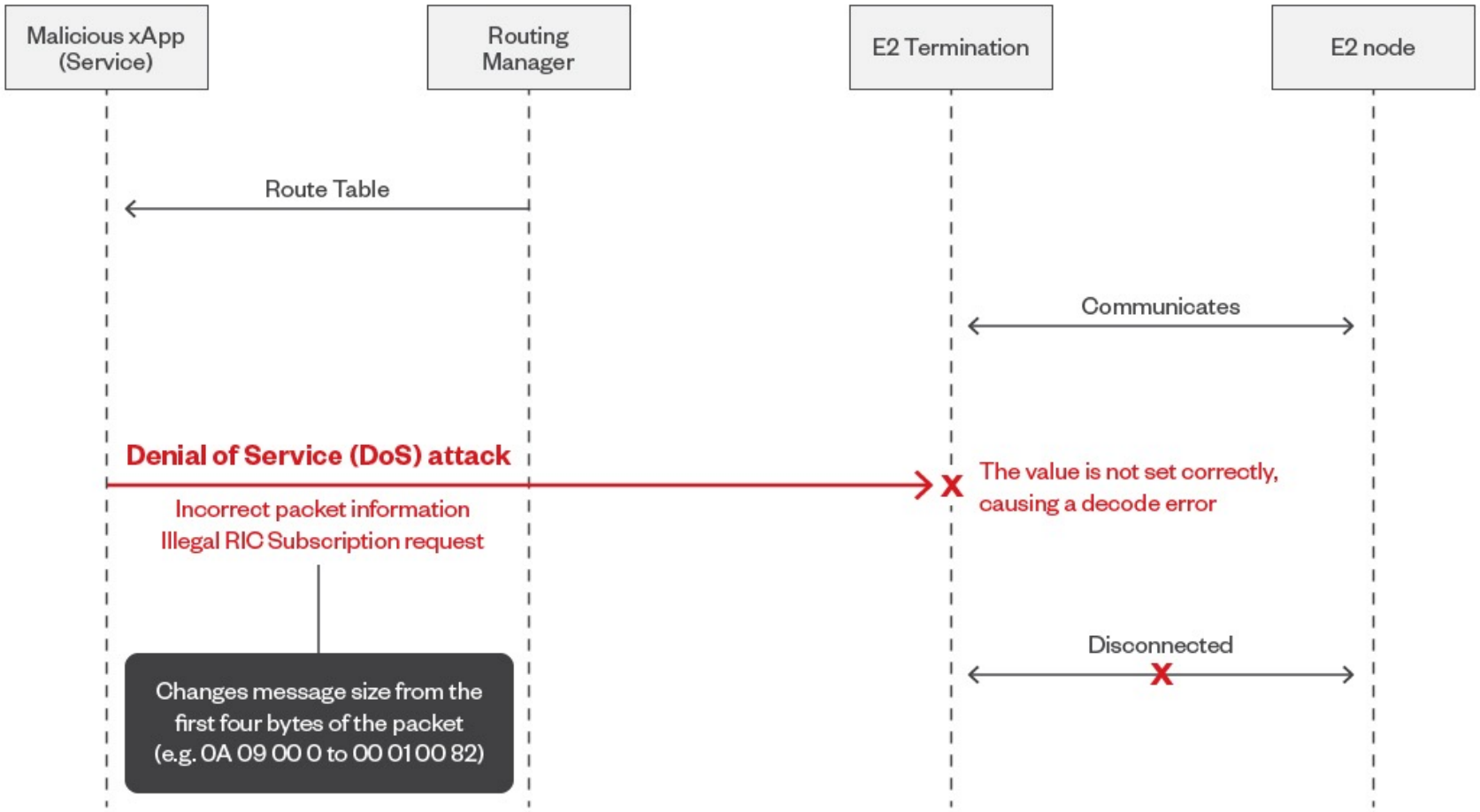
CTONE

O-RAN CVEs

There are **11** CVE Records that match your search.

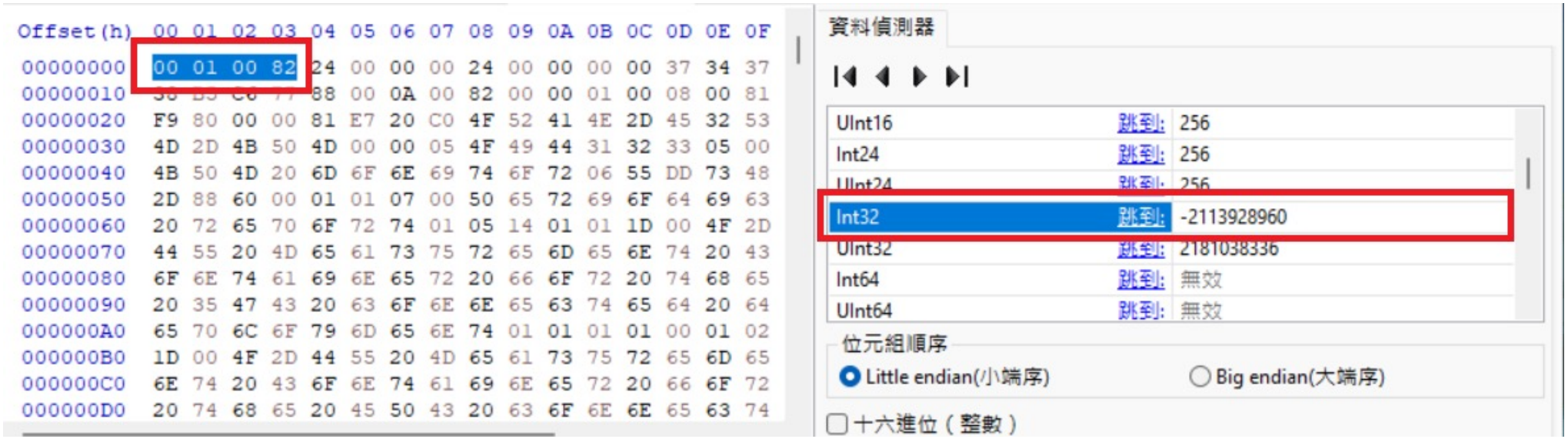
Name	Description
CVE-2024-34048	O-RAN RIC I-Release e2mgr lacks array size checks in E2nodeConfigUpdateNotificationHandler.
CVE-2024-34047	O-RAN RIC I-Release e2mgr lacks array size checks in RicServiceUpdateHandler.
CVE-2024-34046	The O-RAN E2T I-Release Prometheus metric Increment function can crash in sctpThread.cpp for message.peerInfo->sctpParams->e2tCounters[IN_SUCC][
CVE-2024-34045	The O-RAN E2T I-Release Prometheus metric Increment function can crash in sctpThread.cpp for message.peerInfo->counters[IN_INITI][MSG_COUNTER][
CVE-2024-34044	The O-RAN E2T I-Release buildPrometheusList function can have a NULL pointer dereference because peerInfo can be NULL.
CVE-2024-34043	O-RAN RICAPP kpimon-go I-Release has a segmentation violation via a certain E2AP-PDU message.
CVE-2023-42358	An issue was discovered in O-RAN Software Community ric-plt-e2mgr in the G-Release environment, allows remote attackers to cause a denial of service (
CVE-2023-41628	An issue in O-RAN Software Community E2 G-Release allows attackers to cause a Denial of Service (DoS) by incorrectly initiating the messaging procedure
CVE-2023-41627	O-RAN Software Community ric-plt-lib-rmr v4.9.0 does not validate the source of the routing tables it receives, potentially allowing attackers to send forge
CVE-2023-40998	Buffer Overflow vulnerability in O-RAN Software Community ric-plt-lib-rmr v.4.9.0 allows a remote attacker to cause a denial of service via the packet size
CVE-2023-40997	Buffer Overflow vulnerability in O-RAN Software Community ric-plt-lib-rmr v.4.9.0 allows a remote attacker to cause a denial of service via a crafted packet

Negative Packet Size in RMR



CTONE

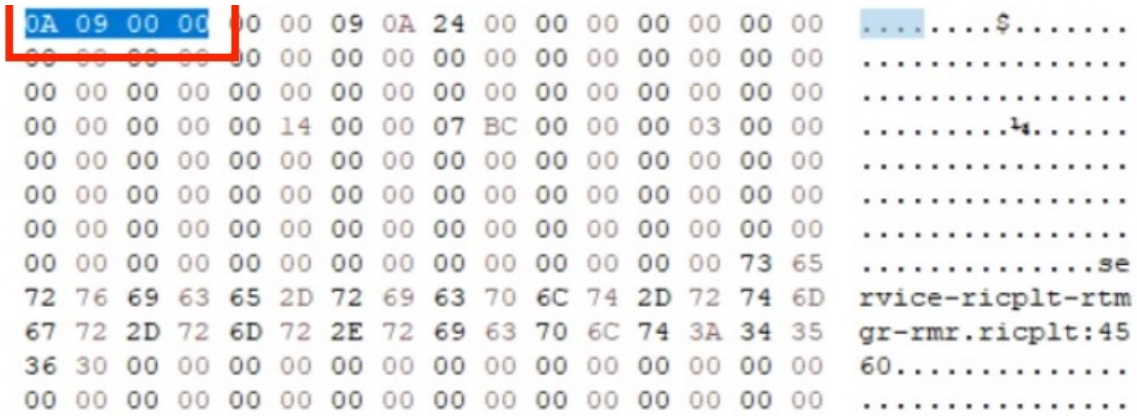
Negative Packet Size in RMR

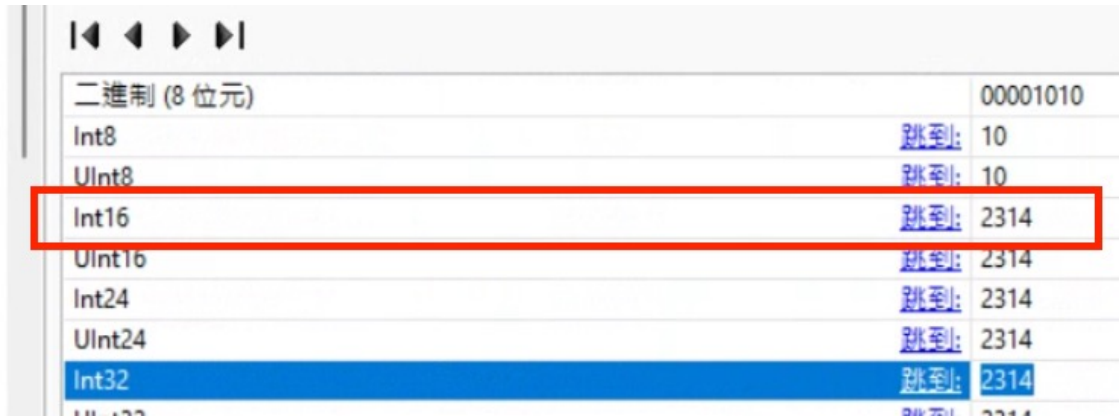


CTONE

Negative Packet Size in RMR

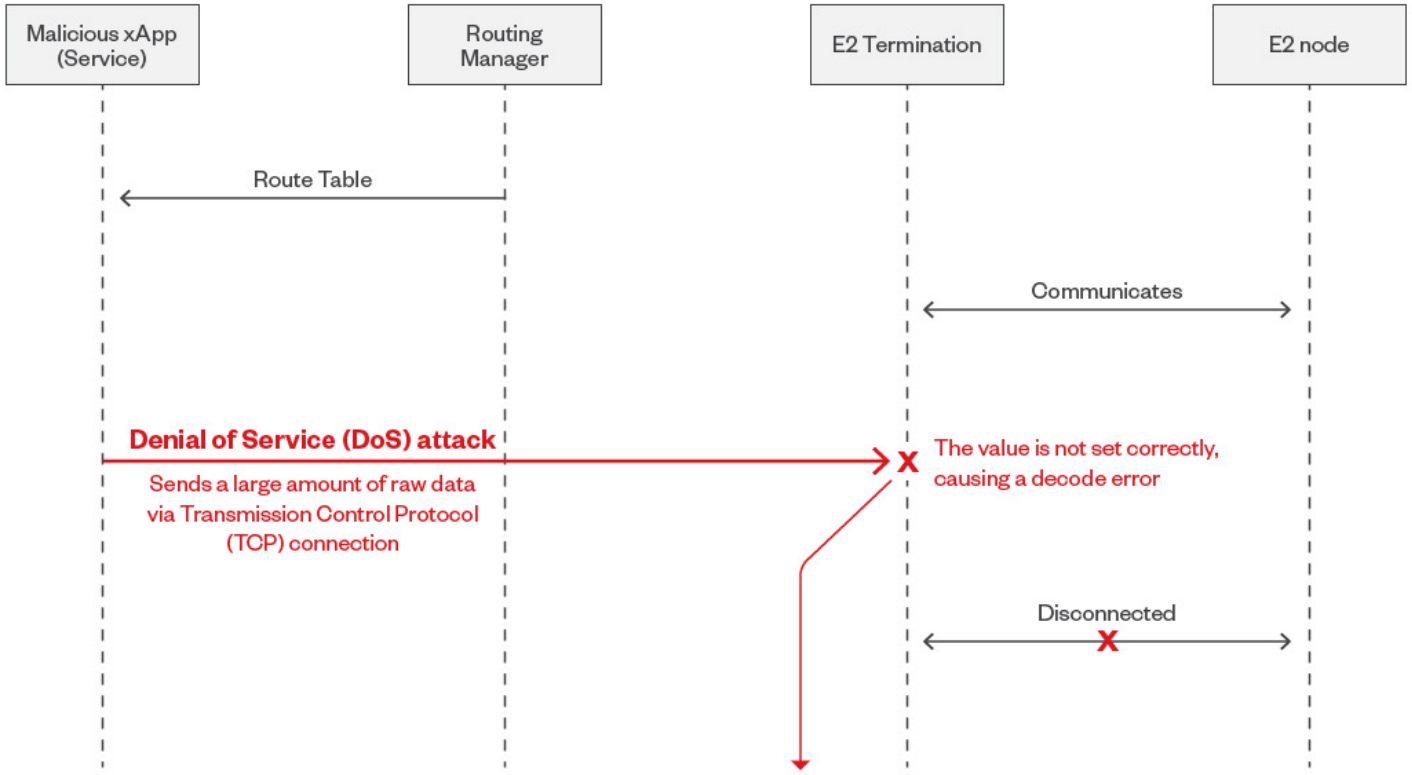
```
1298508/RMR [DBUG] ====== data callback top of loop bidx=0 msize=0 ipt=0 remain=2314
1298508/RMR [DBUG] need 9
1298508/RMR [DBUG] river->ipt 0
1298508/RMR [DBUG] extract msg len converted from net order to: 2314
1298508/RMR [DBUG] data callback setting msg size: 2314
1298508/RMR [DBUG] data callback enough in the buffer size=2314 need=2314 remain=2314 flgs=00
1298508/RMR [DBUG] ##### data callback finished
```





CTONE

Incorrect RMR Format Parsing



Incorrect RMR format at function

```
buf2mbuf ( ctx, raw_mag, msg_size, sender_fd) {
    ref_tpbuf ( mbuf, msg_size); // parse the packet and get header
    ..
    ..
    d1 = DATA1_ADDR ( hdr );
    if ( call_id = ( unsigned int ) d1 [D1_CALLID_IDX] == 0 ){
    ...                    //abnormal access
```


CTONE

Incorrect RMR Format Parsing

```
#0  0x00007fcaba4c86c7 in buf2mbuf (ctx=0x23e4f30, raw_msg=0x7fcaac001420 "\346\271", msg_size=263566, sender_fd=18)
    at /w/workspace/ric-plt-lib-rmr-rt-cmake-packagecloud-stage-master/src/rmr/si/src/mt_call_si_static.c:96

warning: Source file is more recent than executable.
96                                      if( (call_id = (unsigned int) d1[D1_CALLID_IDX]) == 0 ) {
t set, just queue
[Current thread is 1 (Thread 0x7fcab54ef700 (LWP 4038311))]
(gdb) bt
#0  0x00007fcaba4c86c7 in buf2mbuf (ctx=0x23e4f30, raw_msg=0x7fcaac001420 "\346\271", msg_size=263566, sender_fd=18)
    at /w/workspace/ric-plt-lib-rmr-rt-cmake-packagecloud-stage-master/src/rmr/si/src/mt_call_si_static.c:96
#1  0x00007fcaba4c8bad in mt_data_cb (vctx=0x23e4f30, fd=18,
    buf=0x2405070 "tainer for the EPC connected deployment\001\001\001\001", buflen=6634)
    at /w/workspace/ric-plt-lib-rmr-rt-cmake-packagecloud-stage-master/src/rmr/si/src/mt_call_si_static.c:269
#2  0x00007fcaba4ce3bf in SIwait (gptr=0x2404ea0)
    at /w/workspace/ric-plt-lib-rmr-rt-cmake-packagecloud-stage-master/src/rmr/si/src/si95/siwait.c:126
#3  0x00007fcaba4c8e38 in mt_receive (vctx=0x23e4f30)
    at /w/workspace/ric-plt-lib-rmr-rt-cmake-packagecloud-stage-master/src/rmr/si/src/mt_call_si_static.c:370
#4  0x00007fcab9f6f609 in start_thread (arg=<optimized out>) at pthread_create.c:477
#5  0x00007fcabab53133 in clone () at ../sysdeps/unix/sysv/linux/x86_64/clone.S:95
```

Incorrect RMR Format Parsing

```
ref_tpbuf( mbuf, msg_size );                    // point mbuf at bits in the datagram
hdr = mbuf->header;
if( hdr->flags & HFL_CALL_MSG ) {               // call generated message; ignore call-id etc and queue
    queue_normal( ctx, mbuf );
} else {
    if( RMR_D1_LEN( hdr ) <= 0 ) {                                  // no call-id data; just queue
        queue_normal( ctx, mbuf );
    } else {
        d1 = DATA1_ADDR( hdr );
        if( (call_id = (unsigned int) d1[D1_CALLID_IDX]) == 0 ) {   // call_id not set, just queue
            queue_normal( ctx, mbuf );
        } else {
            chute = &ctx->chutes[call_id];
            chute->mbuf = mbuf;
            sem_post( &chute->barrier );            // the call function can vet xaction id in their own thread
        }
```

1. Parse the packet and get the header

2. Calculate memory address

3. illegal access

```
[DBUG] header raw : 0x14000000
[DBUG] n1:280 n2:0
[DBUG] hdr->flags= 4
```

```
[DBUG] header raw : 0x2D4D5332
[DBUG] n1:16981632 n2:1328366421
[DBUG] hdr->flags= 65793
```

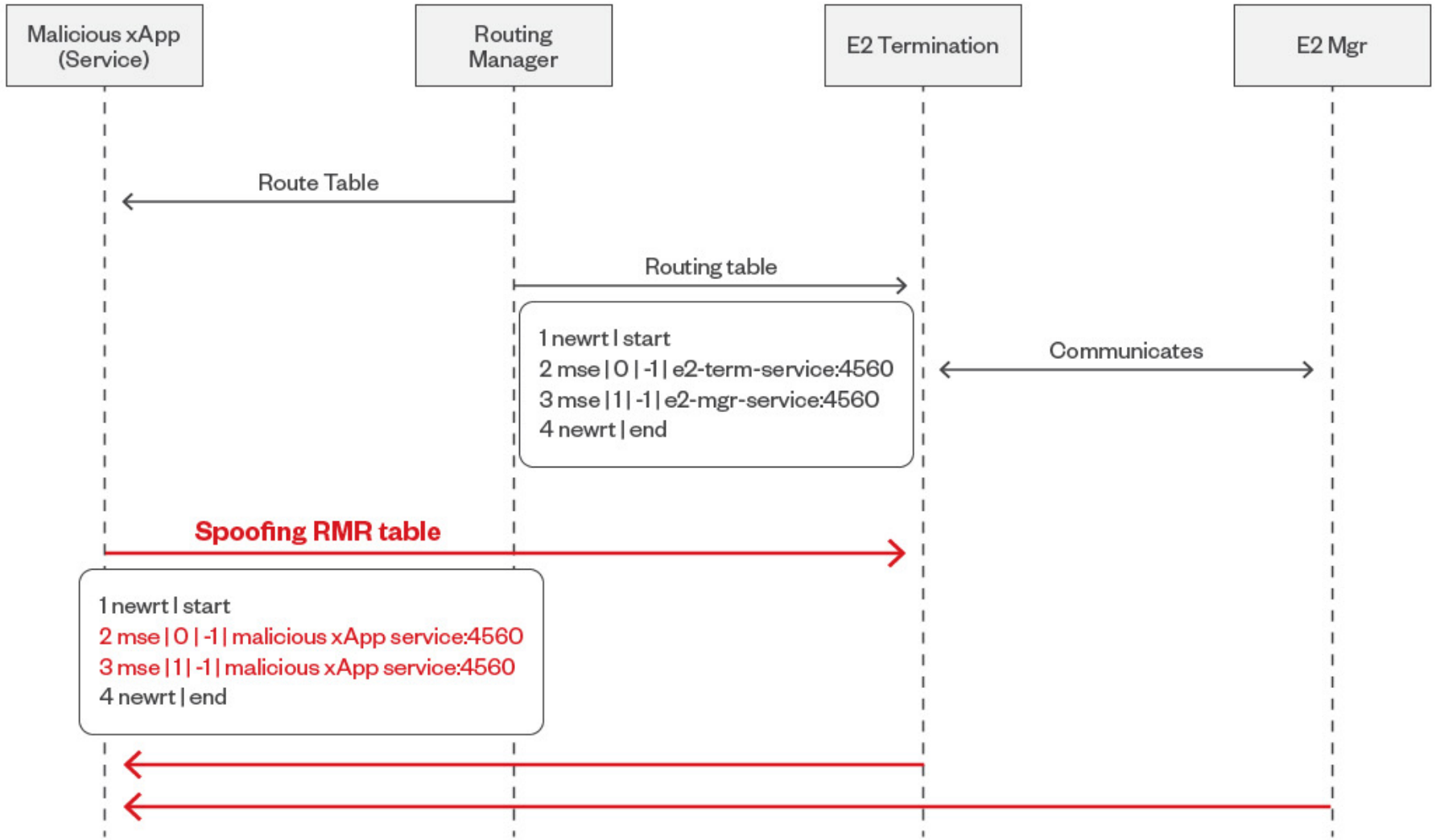

CTONE

Vulnerability in Near-RT RIC

CVE-2023-41627

CTONE

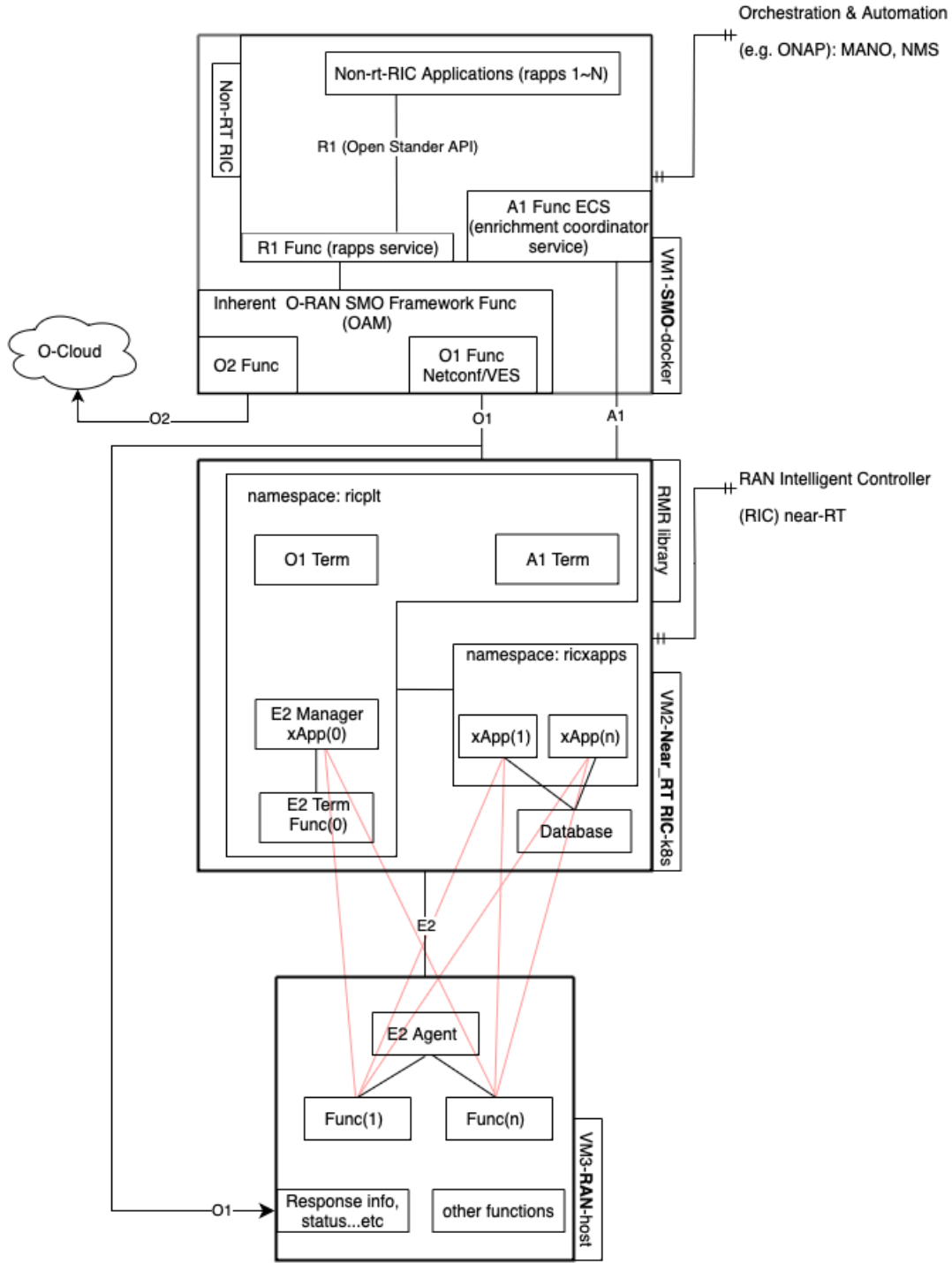
Route Table Spoofing



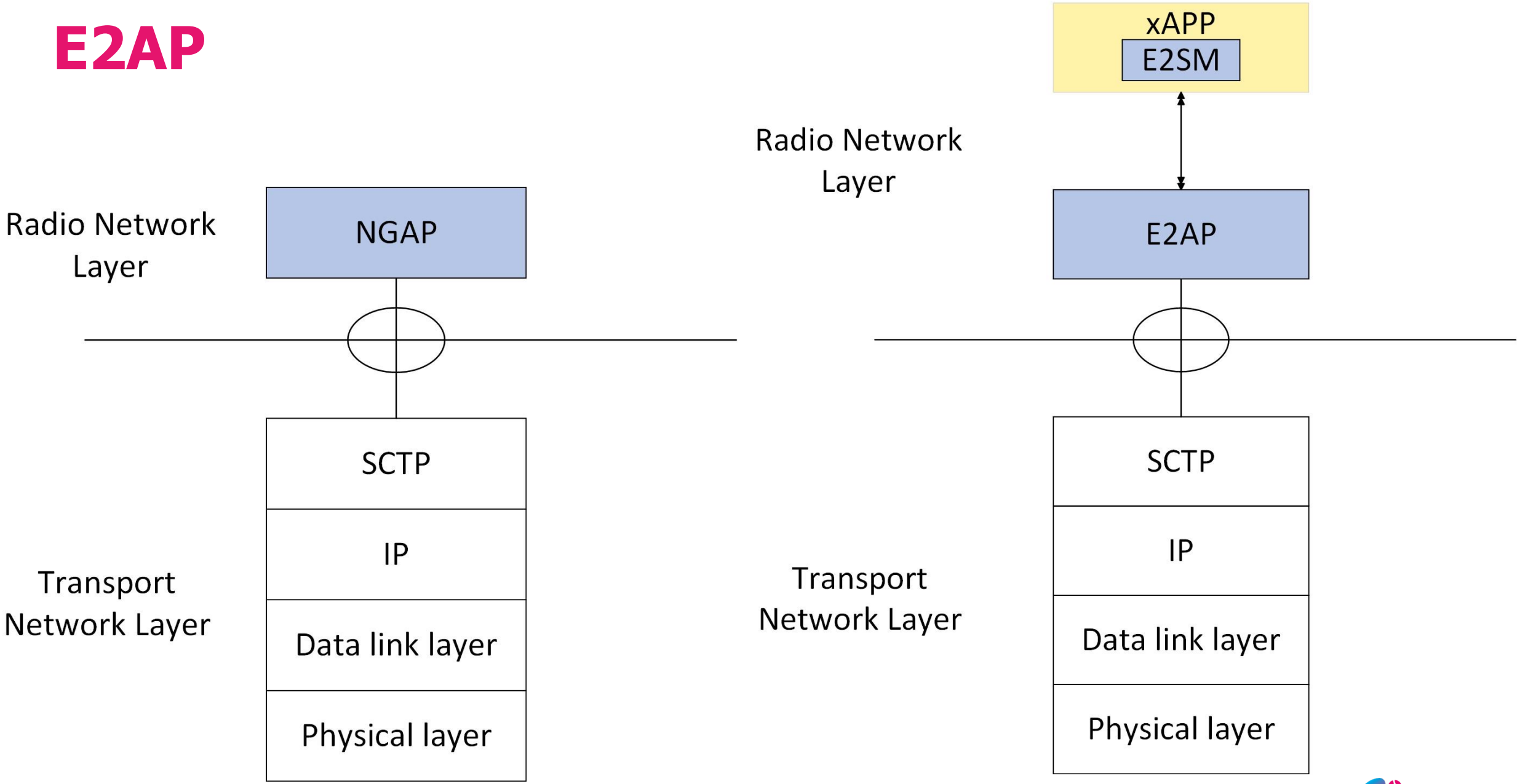
CTONE

How was it discovered?

Parser & Fuzzer


Orchestration & Automation
(e.g. ONAP): MANO, NMS
Non-RT RIC
Non-rt-RIC Applications (rapps 1~N)
R1 (Open Stander API)
A1 Func ECS
(enrichment coordinator service)
R1 Func (rapps service)
VM1-SMO-docker
Inherent O-RAN SMO Framework Func (OAM)
O-Cloud
O2 Func
O1 Func Netconf/VES
O2
O1
A1
RAN Intelligent Controller
(RIC) near-RT
RMR library
namespace: ricplt
O1 Term
A1 Term
namespace: ricxapps
E2 Manager xApp(0)
xApp(1)
xApp(n)
E2 Term Func(0)
Database
VM2-Near_RT RIC-k8s
E2
E2 Agent
Func(1)
Func(n)
VM3-RAN-host
O1
Response info, status...etc
other functions

E2AP

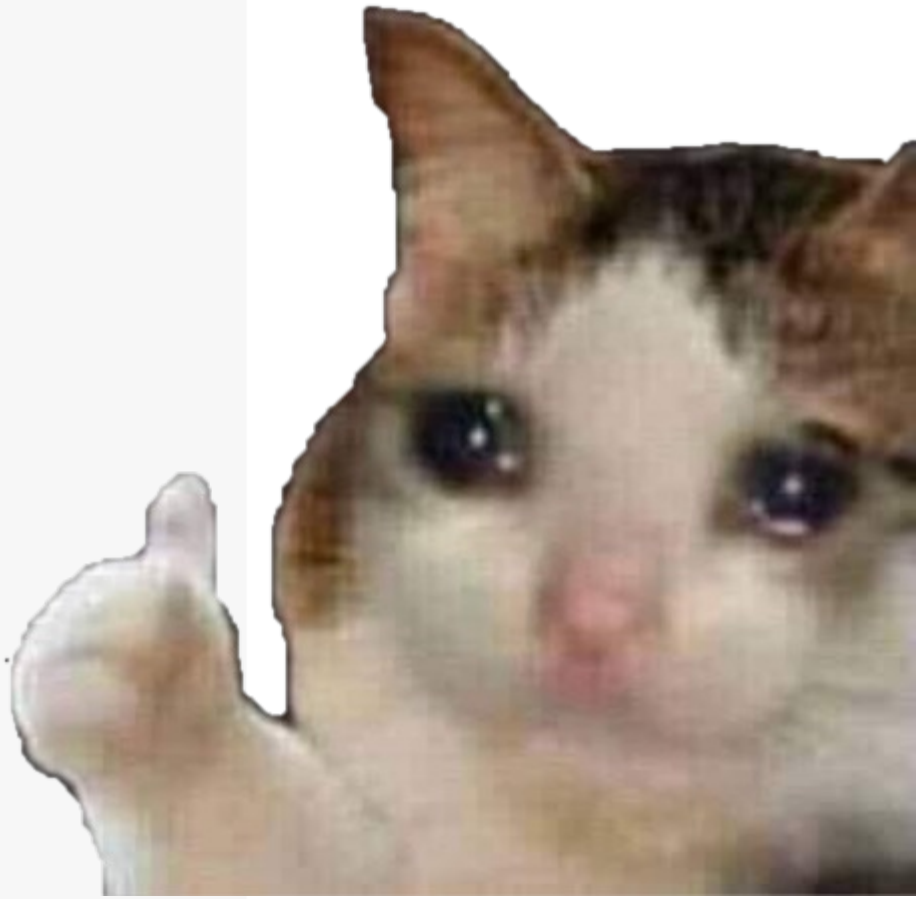

Radio Network Layer
NGAP
Transport Network Layer
SCTP
IP
Data link layer
Physical layer
xAPP
E2SM
Radio Network Layer
E2AP
Transport Network Layer
SCTP
IP
Data link layer
Physical layer

CTONE

xAPP collection information from E2 nodes via KPM E2SM

```
plmnID
sliceID
fiveQI
qFI
qCI
qCImax
qCImin
aRPmax
aRPmin
bitrateRange
layerMU-MIMO
sUM
distBinX
distBinY
distBinZ
preLabelOverride
startEndInd
min
max
avg
ssbIndex
nonGoB-BFmode-Index
mIMO-mode-Index

gBR
aMBR
isStat
isCatM
rSRP
rSRQ
ul-rSRP
cQI
fiveQI
qCI
Cell Global ID
UE ID
```



O-RAN.WG3.E2SM-R003-03.00

CTONE

CTONE

Thank You.