

LTE/CBRS

Implementation of CBRS deployment modeled after Seattle Community Network infrastructure

- [High-Level References](#)
- [The Controller \(Evolved Packet Core/EPC\)](#)
- [The Radio \(eNodeB/eNB\)](#)

High-Level References

Resources from SCN

- [lte-background - Google Slides](#)
 - Discusses various software components that make up a typical LTE deployment - most relevant to our deployment
- [Supplemental-cellular-networks-and-5g.pptx - Google Slides](#)
 - Provides background on cellular technology - more of a general overview of the technologies
- [SCN LTE Online Resources - Google Docs](#)
 - Collection of links to provide information on individual components (terminology, other software, etc.)

The Controller (Evolved Packet Core/EPC)

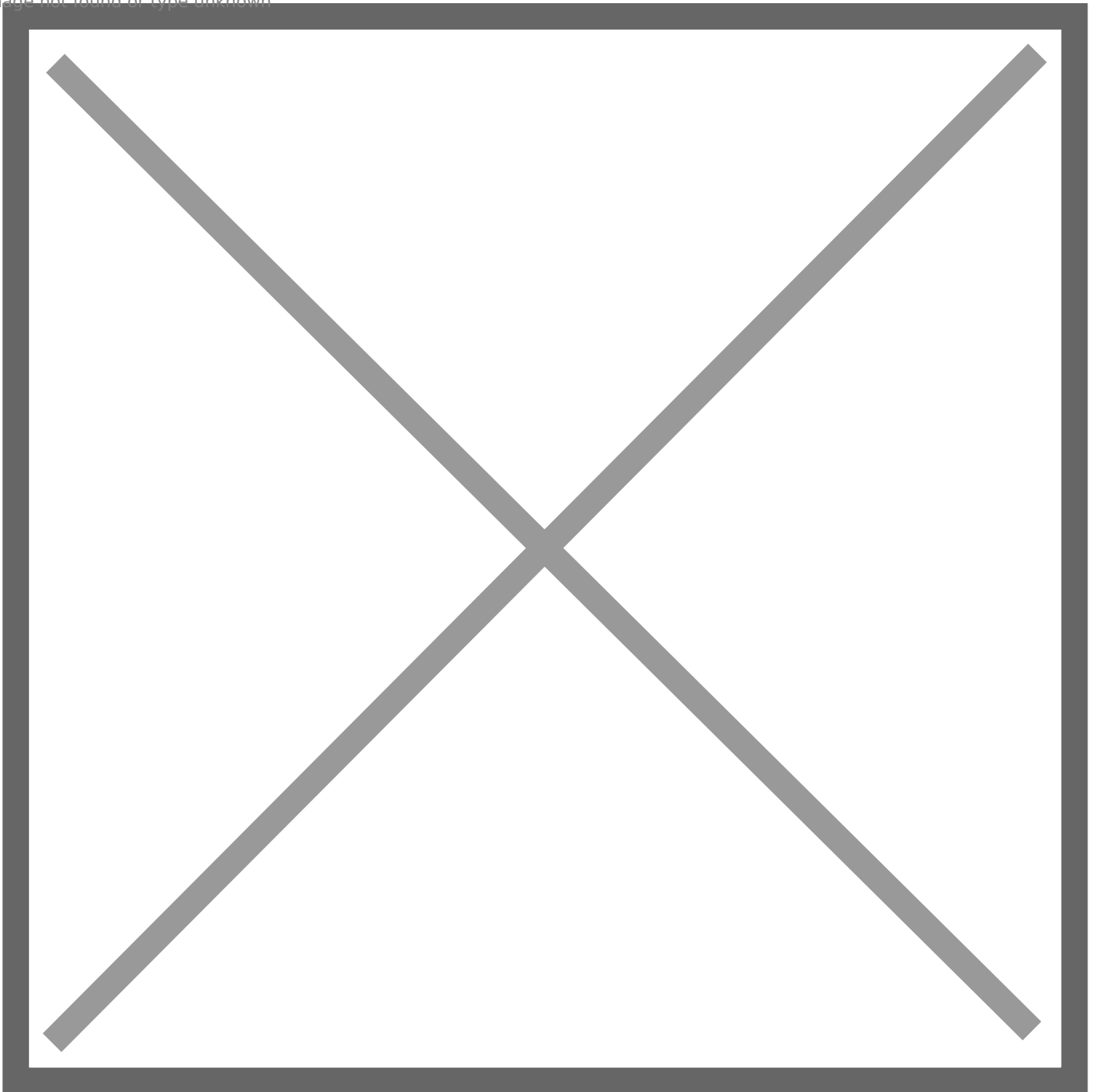
What is Open5GS and why it is needed?

From [GitHub](#):

Open5GS is a C-language Open Source implementation for 5G Core and EPC, i.e. the core network of LTE/NR network

Components

Image not found or type unknown



From Page 18 of Ite-background presentation (see [High-Level Resources](#))

Mesh Implementation

The Open5GS setup used by NYC Mesh is bootstrapped using software from the University of Washington [Information and Communication Technology for Development \(ICTD\) Lab](#), called **CoLTE**, which is used and recommended by the [Seattle Community Network](#) (SCN).

What is CoLTE?

From [GitHub](#):

CoLTE is the Community LTE Project. It is designed to be an all-in-one turnkey solution that sets up a small-scale locally-run cellular network.

Deployment

- Open5GS lives on a virtual machine running on [Jon](#) at SN3
 - Access: lte.mesh (10.70.90.170)
- colte-cn-4g was installed and configured according to the [documentation](#)
 - Note: colte-prepaid was not installed as it is not required for LTE operation
 - Information regarding SIM cards and credentials provided directly from SCN
- Unlike SCN, our deployment runs the EPC "off-prem"
 - Instead of connecting the eNB (the LTE radio, e.g. BaiCells) directly to a computer/router with two ethernet ports, we connect the eNB directly to a hub with an existing router
 - The stipulation of this is we do not care about the 192.168.150.0/24 or 192.168.151.0/24 IP addresses mentioned in the documentation, as the eNB will be configured to get DHCP from the local hub
 - For routing purposes, we will use OSPF through the tunnel for clients that are able to use it (e.g. a MikroTik LTE device). The iptables rule can remain for other devices (e.g. a cellphone)

Integration

- Running the script produces an LTE tunnel that all of the LTE clients will use on a common subnet, the "ogstun" interface:

- ```
3: ogstun: <POINTOPOINT, MULTICAST, NOARP, UP, LOWER_UP> mtu 1400 qdisc fq_codel state UP group default qlen 500
 link/none
 inet 10.45.0.1/16 brd 10.45.255.255 scope global ogstun
 valid_lft forever preferred_lft forever
 inet6 fe80::ae33:20bf:d586:8ed6/64 scope link stable-privacy
 valid_lft forever preferred_lft forever
```

- We will use BIRD to run OSPF on this tunnel:

- ```
protocol ospf {
    ...
    area 0 {
        networks {
            10.70.90.0/24;
            10.45.0.0/16;
        };
        interface "enp6s18" {
            cost 10;
            type broadcast;
        };
        interface "ogstun" {
            cost 100;
            type ptmp;
```

```
};
```

```
};
```

```
};
```

The Radio (eNodeB/eNB)

Equipment Used

- Baicells Nova 233 ([datasheet](#))
 - Paired antenna: Alpha Wireless AW-3014-T4 ([datasheet](#))
 - Meant for long-range/outdoor deployment
- Baicells Nova 227 ([datasheet](#))
 - Meant for short-range/microcell
 - Integrated antenna

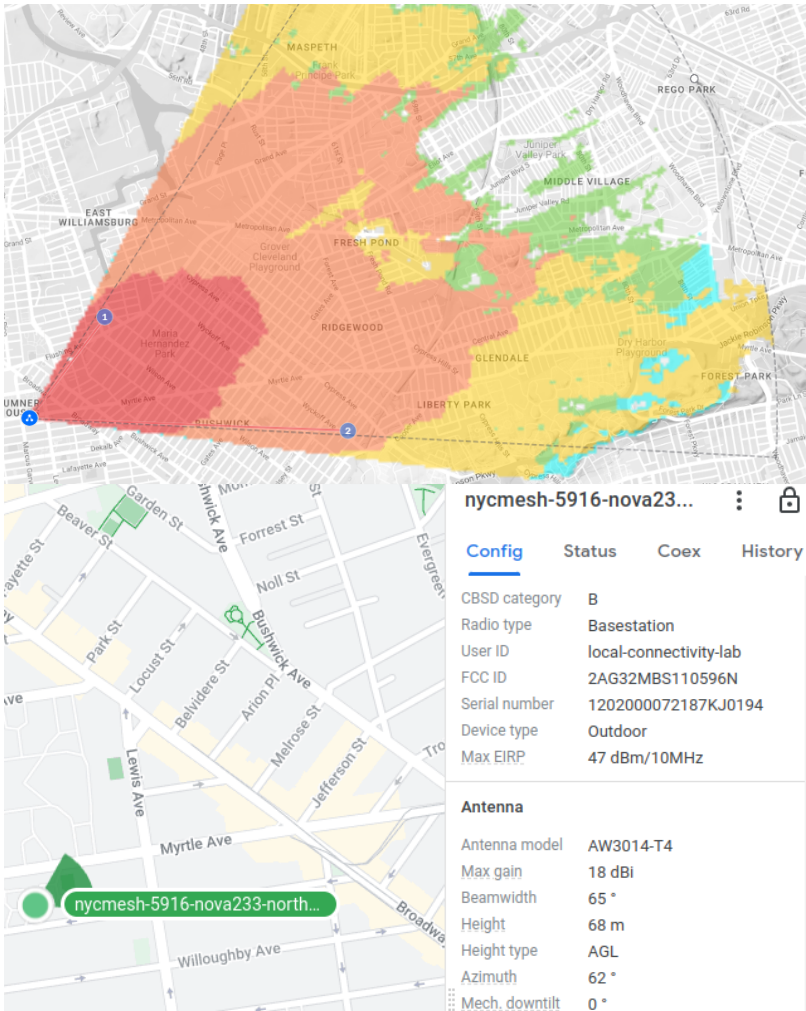
Configuration

- eNB was configured per [documentation](#) from the Seattle Community Network (SCN)
 - Because our controller/EPC will not be directly attached to the radio, we will configure the Baicells' WAN interface to be DHCP (see more about the NYC Mesh implementation [here](#))
 - The MME IP will always be 10.70.90.170 to use the Open5GS server at SN3, routed through the Mesh network
 - Because the Google SAS product is [paid](#) software we are using the account from UW's local connectivity lab
 - To sign configurations, a CPI License is still needed - otherwise we would have to ask another licensed CPI to sign the configuration any time a change is made (this is attached to a person, not an organization)
 - The PLMN used by SCN (as we are using their SIM cards) is 91054

Current Deployment



The Nova 233 and AW-3014-T4 installed on the East Mast of Vernon.



The estimated range for this deployment (sourced from Ubiquiti's UISP Design Center [tool](#) and generic 60° antenna parameters and their LIDAR info) and the configuration parameters for the SAS platform based on the real-world installation characteristics.

Radio Install Details

- The Nova 233 is powered using a DC cable (crimped ToughCable) into a DC junction box inside the water tower, and plugged into the EP-S16
- No additional configuration was done to the router to support the LTE connectivity
- Baicells supports a mode called **LGW**, which allows the radio hardware to perform NAT and terminate LTE traffic locally, but this would then require per-hub configuration to support OSPF (out-of-scope at this time)

Client Install Details



- Current client is a MikroTik LHG LTE device with a R11e-4G modem ([datasheet](#)) at 5107 (10.69.51.207)
 - RouterOS 7 ptmp mode appears to work natively with BIRD ptmp mode, so ROS7 is used
- Connects to a standard rooftop OmniTik and provides a backup with cost 100 to SN3 (due to the LTE tunnel, no intermediate hubs are considered in cost calculation)
- Specific configuration for OSPF deployment:

```
/interface lte apn add apn="" ip-type=ipv4 name=internet
/ip address add address=10.45.0.7/16 interface=lte1 network=10.45.0.0
/interface lte set [ find default-name=lte1 ] apn-profiles=internet band=43 mtu=1400 sms-
read=no
```

```

/routing filter rule
add chain=ospf-in disabled=no rule="set distance 205;\
\nset bgp-communities 65000:110;\
\naccept;"
add chain=ospf-out disabled=no rule="if (dst == 10.69.0.0/16) {\
\n reject\
\n}"
add chain=ospf-out disabled=no rule="if (dst == 10.45.0.0/16) {\
\n reject\
\n}"
add chain=ospf-out disabled=no rule=accept
/routing ospf interface-template
add area=default cost=100 disabled=no interfaces=lte1 networks=10.45.0.0/16 priority=1
type=ptmp
add area=default cost=10 disabled=no interfaces=ether1 networks=10.69.0.0/16 priority=1
type=ptmp-broadcast

```

1. Sets the APN to "internet" (required for use with the Open5GS core)
2. Adds a /16 to match the 10.45.0.0/16 network on the LTE tunnel
 - Note: upon connection, a 10.45.0.0/32 address is assigned by the Open5GS core, which will not allow OSPF to function (as it does not match the CIDR of the "ogstun" interface on the EPC/controller)
3. Sets the LTE interface to use the "internet" APN and use Band 43 (the CBRS frequency supported on the Nova 233)
4. Adds the standard OSPF route filters used, while also filtering announcement of the 10.45.0.0/16 LTE address space
5. Adds the standard OSPF interface templates for the LTE and "mesh" OSPF interfaces