

Визуализация топологии сети с помощью скрипта Python и NeXt UI

В данный момент в сети существует немало утилит и программ с открытым кодом для визуализации сетевых топологий. Исходными данными, как правило, для них выступают данные таких протоколов, как Cisco Discovery Protocol (CDP), Link Layer Discovery Protocol (LLDP), Open Shortest Path First (OSPF), Intermediate System to Intermediate System (ISIS).

Реализации CDP и LLDP могут использоваться в качестве решения для небольших доменов, с ограниченным количеством сетевых устройств. В отношении CDP, следует заметить, что данный протокол является исключительно проприетарным и может применяться в сетях только с оборудованием Cisco.

OSPF и ISIS относятся к категории Interior Gateway Protocol (IGP), являются по сути Link State протоколами, и их базы Link State Database (LSDB) содержат информацию о всех хостах домена, состоянии и типе линков, метриках для расчета дерева кратчайшего пути (shortest-path tree, используется алгоритм Dijkstra), т. е. всю необходимую информацию о топологии сети. Любой из этих протоколов в большинстве случаев является неотъемлемым атрибутом дизайна любой сети, будь-то Internet Service Provider, Data Center или Enterprise.

В этой статье речь пойдет о небольшом скрипте Python и наборе шаблонов TextFSM, с помощью которых, в связке с open source решением NeXt UI (применяется в [SDN контроллере ODL](#)), используя LSDB протоколов ISIS, OSPF, легко и просто визуализировать топологию сети, содержащую от нескольких десятков до несколько сотен устройств. Следует отметить, что решение может применяться в мультивендорных сетях (в текущей версии поддерживаются протоколы следующих вендоров: Nokia - OSPF и ISIS; Juniper - OSPF и ISIS; Cisco - OSPF и ISIS; Huawei - ISIS).

Для запуска скрипта необходимо:

1. Загрузить все содержимое из [git clone](#):

```
home@home:~/TMP$ git clone https://github.com/a-kuzik/igp_topo_multi
Клонирование в «igp_topo_multi»...
remote: Enumerating objects: 235, done.
remote: Counting objects: 100% (235/235), done.
remote: Compressing objects: 100% (130/130), done.
remote: Total 235 (delta 126), reused 189 (delta 100), pack-reused 0
Получение объектов: 100% (235/235), 1.24 MiB | 3.07 MiB/s, готово.
Определение изменений: 100% (126/126), готово.
```

2. Зайти в каталог igp_topo_multi и проверить его содержимое:

```
cd igp_topo_multi
ls -l
```

```
home@home:~/TMP$ cd igp_topo_multi/
home@home:~/TMP/igp_topo_multi$ ls -l
итого 32
-rw-rw-r-- 1 home home 657 янв 31 19:09 config.yaml
-rw-rw-r-- 1 home home 9682 янв 31 19:09 igp_topo_light.py
drwxrwxr-x 5 home home 4096 янв 31 19:09 nextUI
-rw-rw-r-- 1 home home 1437 янв 31 19:09 README.md
drwxrwxr-x 2 home home 4096 янв 31 19:09 templates
drwxrwxr-x 2 home home 4096 янв 31 19:09 test_files
```

3. Отредактировать файл config.yaml, используя текстовый редактор:

```
nano config.yaml
```

```
GNU nano 4.8 config.yaml
1 #####
2 ###Please specify here host's vendor and type of IGP
3 #####
4 host_type: juniper
5 igp: ospf
6
7 #####
8 ###Please specify here path to your file with LSDB information
9 #####
10 lsdb_file: test_files/sh_ospf_router_det_jun.txt
11 #lsdb_file: test_files/sh_isis_db_ext_jun.txt
12 #lsdb_file: test_files/sh_isis_db_det_cisco.txt
13 #lsdb_file: test_files/sh_ip_ospf_db_router_cisco.txt
14 #lsdb_file: test_files/sh_router_isis_db_det_nokia.txt
15
16
17
[ Read 16 lines ]
^G Помощь ^O Записать ^W Поиск ^K Вырезать ^J Выворнять ^C ТекПозиц
^X Выход ^R ЧитФайл ^M Замена ^U Paste Text ^T Словарь ^_ К строке
```

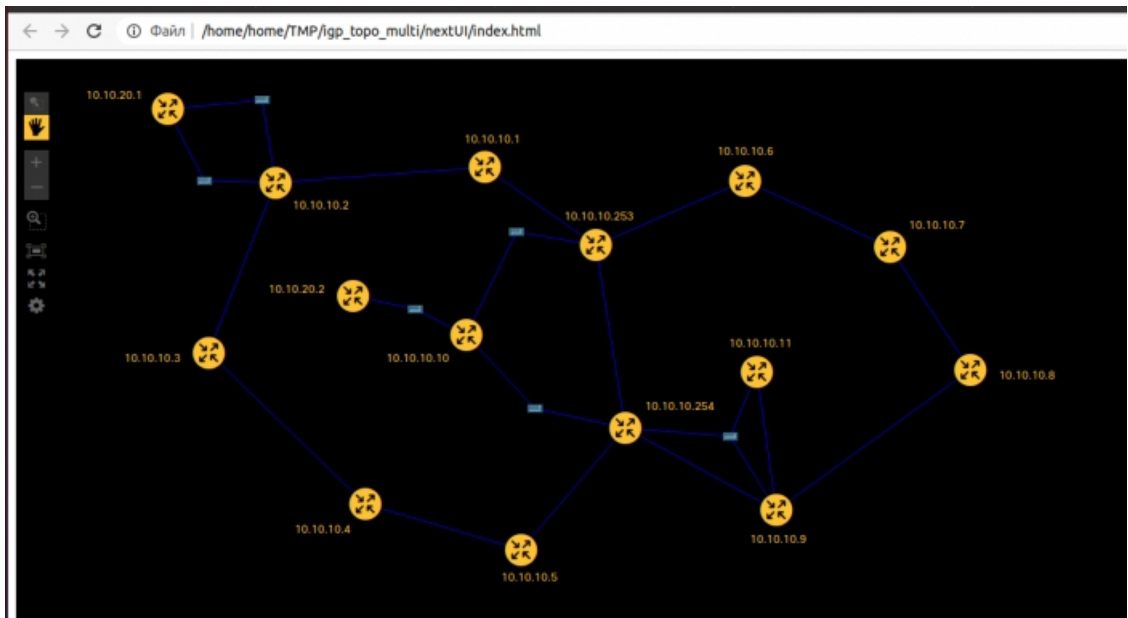
В нем необходимо указать тип хоста (nokia, juniper, cisco, huawei), протокол (ospf или isis), путь и название текстового файла с выводом LSDB (используется каталог test_files, в котором уже содержится несколько файлов с тестовыми топологиями). Сохранить данные и закрыть редактор:

Ctrl+O
Ctrl+x

4. Запустить скрипт:

```
home@home:~/TMP/igp_topo_multi$ python3 igp_topo_light.py
File for NeXtUI was created. If the drawing didn't opened in your browser
automatically, you can find file index.html in the directory /nextUI and open it
manually
home@home:~/TMP/igp_topo_multi$
```

5. В открывшемся окне браузера по умолчанию, отобразится топология сети:



Некоторые пояснения:

1. Загрузку файлов из github.com желательно выполнять в пользовательский каталог без sudo. В противном случае скрипт не сможет запустить браузер. Если браузер не запустится по каким-либо причинам, страницу index.html, которая находится в каталоге nextUI, можно открыть самостоятельно.
2. Пиктограммы маленьких коммутаторов в топологии обозначают broadcast линки (не p2p). Повод задуматься, оправдан ли этот тип для каждого конкретного случая.
3. Цветовые темы NeXt UI можно применять разные из доступных.
4. Положение хостов в браузере можно изменять их перетаскиванием.

