



JANOG52 ハツカソン報告

トポロジ図可視化

～eXchangeしようぜ！～

Member

大野(ソフトバンク)

普段の業務: サーバ、NW機器の運用保守



蟹江 大成(BBIX)

普段の業務:
社内システム開発・運用
Ansible / xFlow / Golang



絹川 隆大(BBIX)

普段の業務: IXサービス開通業務、RS関連
作業の自動化推進中！！



野上(BBIX)

普段の業務:
IXのルートサーバ構築
RPKI周りを少々



概要

【 悩み事・・・ 】

- ・ネットワークが大きくなった際に構成情報を更新するのが手間。。
- ・構築作業など進めていくと実機と構成情報の差分が出てくることある。。

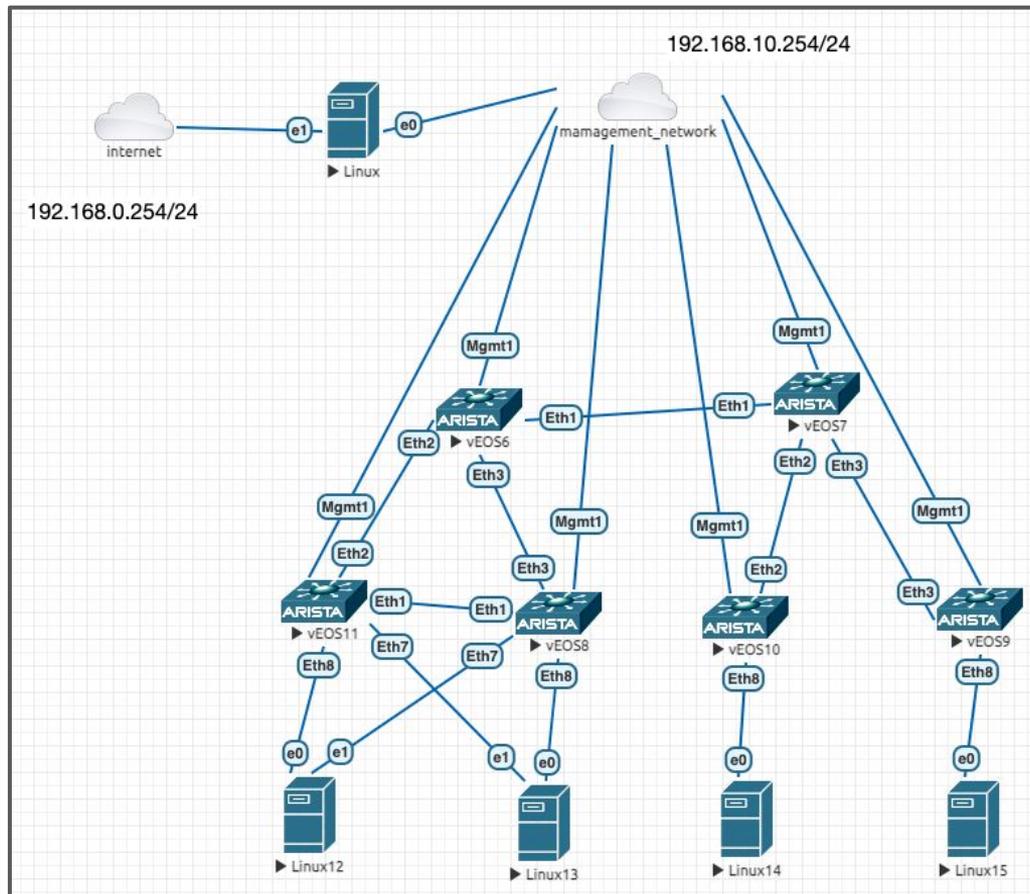


【 解決手段！！ 】

- ・LLDPを使用してAnsibleとinet-hengeを用いたトポロジ図の自動生成
- ・Ansible×Netboxをつかった構成管理



EVE概要図

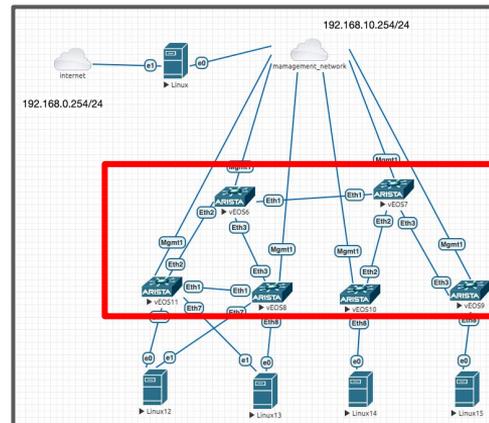


ネットワーク構成

[Network (Arista vEOS-lab)]

・設定の追加

- 1.zerotouchの無効化
- 2.IPアドレス、ホスト名、SSHログインの設定
- 3.全I/FにLLDP設定



サーバ構成

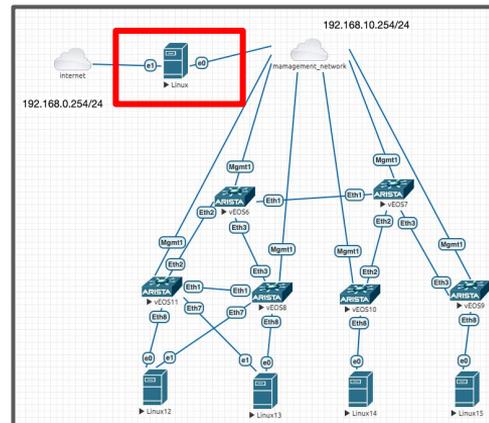
[Linuxサーバ(Ubuntu)]

▪ Ansible, Netbox, inet-hengeのセットアップ

⇒ Ansible: オープンソースの構成管理ツール

⇒ Netbox: ネットワーク関連の情報管理ツール

⇒ inet-henge: 構成管理情報からトポロジ図の作成



可視化できること

LLDPで取れる情報から必要なものを選択的に！

- OS
- Interface名
- TTL
- ホスト名
- native VLAN
- IPアドレス ...

パースできればLLDPで取得できる情報は全て可視化できる

ツール動作概要

LLDPの表示結果

この情報をAnsibleでいい感じに取得してJSON加工！

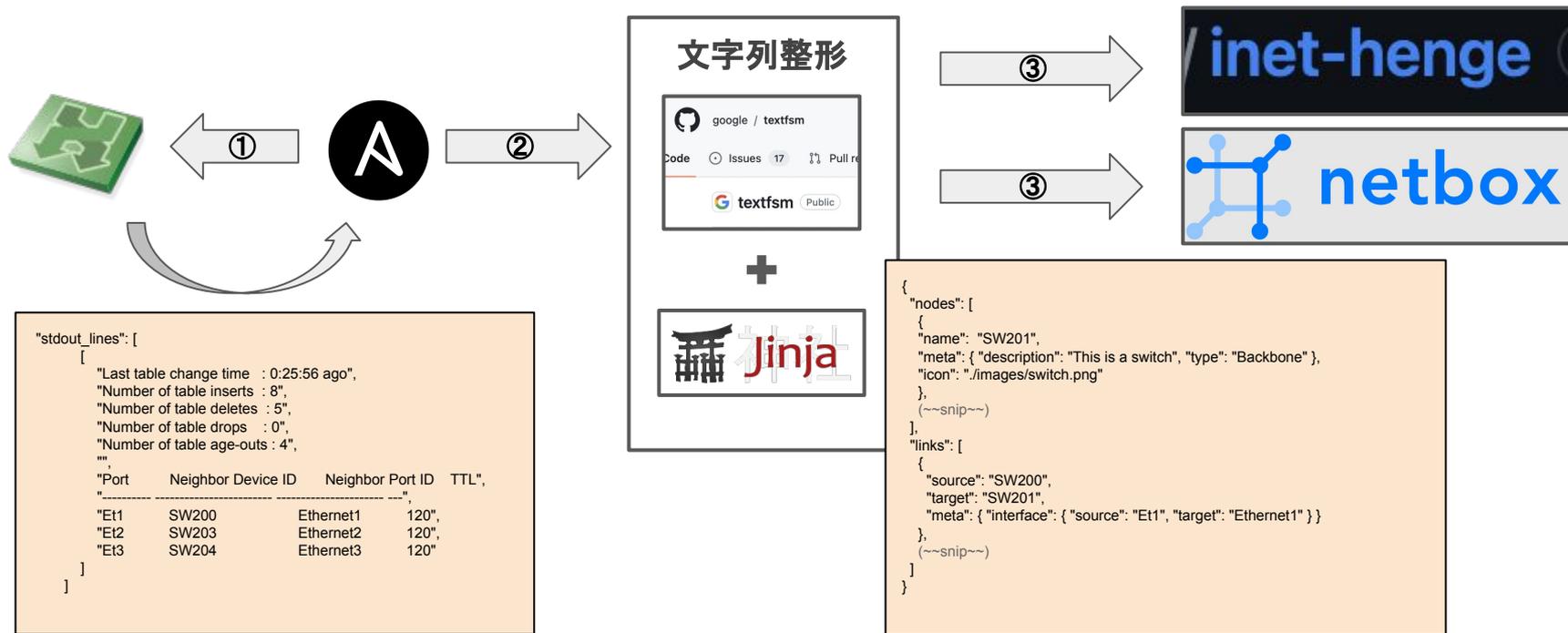
```
SW200#  
SW200#show lldp neighbors  
Last table change time   : 0:02:38 ago  
Number of table inserts  : 13  
Number of table deletes  : 10  
Number of table drops    : 0  
Number of table age-outs : 9
```

Port	Neighbor Device ID	Neighbor Port ID	TTL
Et1	SW201	Ethernet1	120
Et2	SW202	Ethernet2	120
Et3	SW205	Ethernet3	120

```
SW200#
```

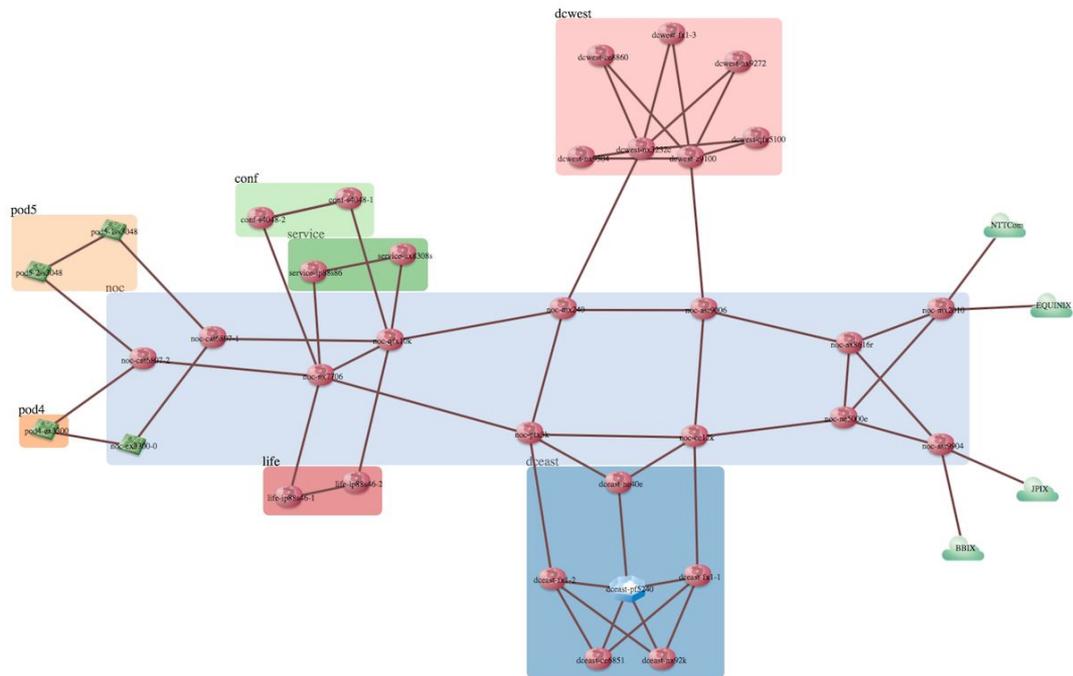
ツール動作概要

- ①AnsibleからSWのLLDPを取得
- ②取得結果を文字列で取得
- ③netbox、inet-hengeに情報提供



inet-henge出力例

取得したSWの情報相结合して、
必要な構成情報を描画



引用: <https://github.com/codeout/inet-henge>

Netbox表示例

取得したSW毎に各インターフェイスの情報を作成

The screenshot shows the Netbox interface for a device named SW200. The breadcrumb navigation is 'Devices > JANOG52@nagasaki'. The device name 'SW200' is prominently displayed, along with its creation and update timestamps. A navigation bar includes tabs for 'Device', 'Interfaces' (with a count of 8), 'Power Ports' (with a count of 1), 'Config Context', 'Render Config', 'Contacts', 'Journal', and 'Changelog'. Action buttons for '+ Add Components', 'Clone', 'Edit', and 'Delete' are visible. A search bar and a 'Configure Table' button are also present. The main content is a table of interfaces.

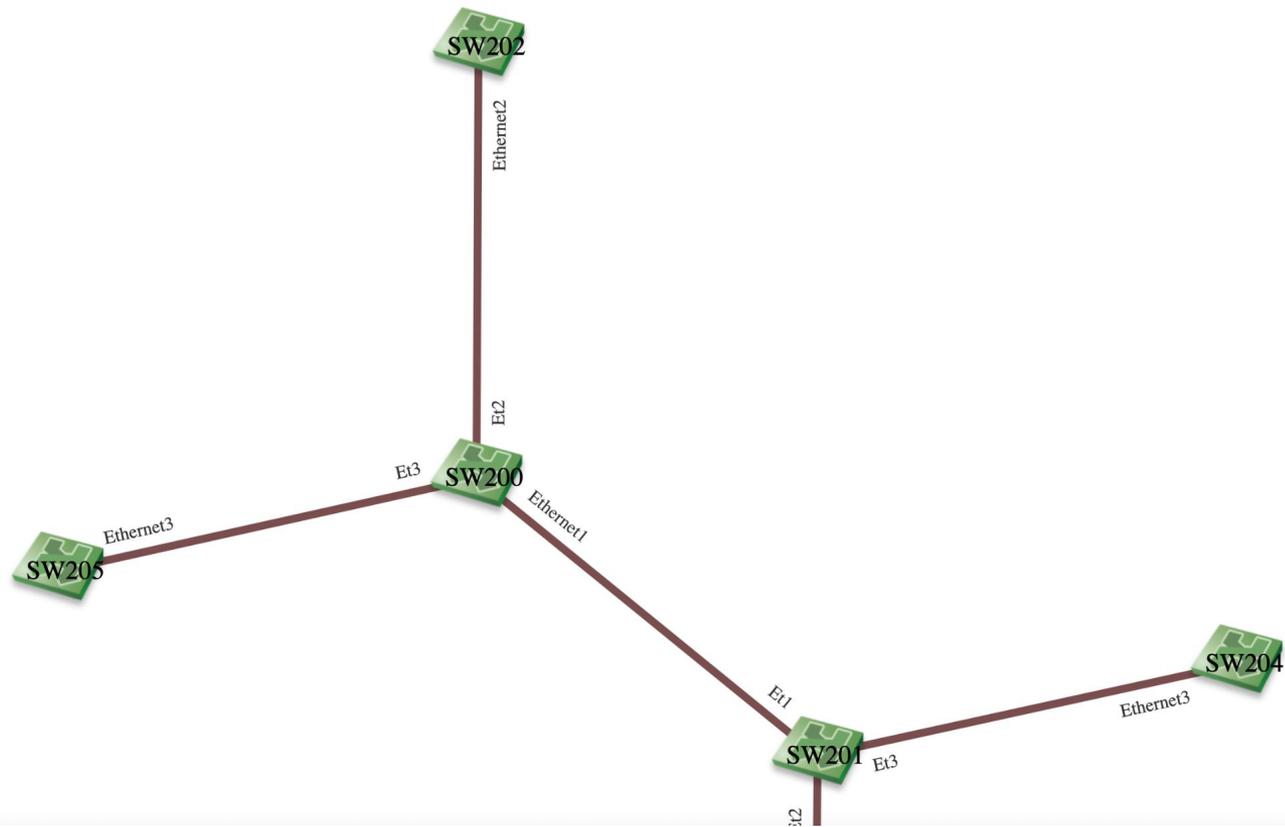
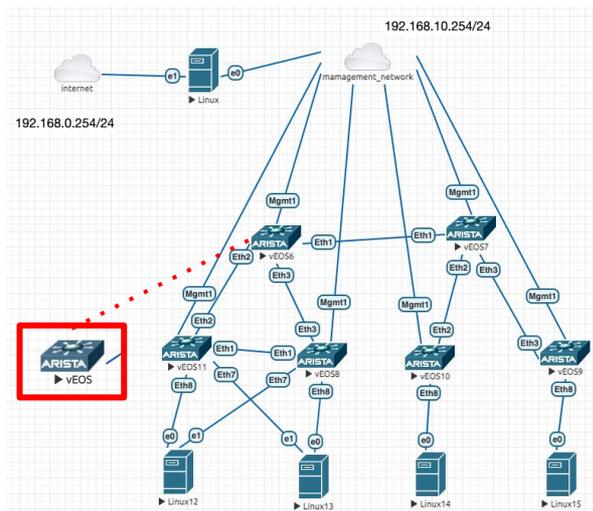
<input type="checkbox"/>	Name	Label	Enabled	Type	Parent interface	LAG	MTU	Mode	Description	IP Addresses	Cable	Connection	
<input type="checkbox"/>	Ethernet 1	—	✓	Other	—	—	—	—	—		#1	SW201 > Ethernet1	+ - [icon] [icon] [icon]
<input type="checkbox"/>	Ethernet 2	—	✓	Other	—	—	—	—	—		#2	SW202 > Ethernet2	+ - [icon] [icon] [icon]
<input type="checkbox"/>	Ethernet 3	—	✓	Other	—	—	—	—	—		#3	SW205 > Ethernet3	+ - [icon] [icon] [icon]

構成現状

SW200に

SW201,202,205

が3つに接続している



ansible結果

ansibleで show lldp neighbors

```
TASK [lldp : display results] *****
ok: [SW200] => {
  "results": {
    "ansible_facts": {
      "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "failed": false,
    "stdout": [
      "Last table change time   : 17:51:58 ago\nNumber of table inserts   : 19\nNumber of table deletes   : 0\nNumber of table drops     : 0\nNumber of table age-outs  : 0\n",
      "Port      Neighbor Device ID  Neighbor Port ID  TTL",
      "-----  -",
      "Et1       SW201                Ethernet1         120",
      "Et2       SW202                Ethernet2         120",
      "Et3       SW205                Ethernet3         120",
      "Et4       SW206                Ethernet1         120"
    ],
    "stdout_lines": [
      "Last table change time   : 17:51:58 ago",
      "Number of table inserts   : 19",
      "Number of table deletes   : 0",
      "Number of table drops     : 0",
      "Number of table age-outs  : 0",
      "",
      "Port      Neighbor Device ID  Neighbor Port ID  TTL",
      "-----  -",
      "Et1       SW201                Ethernet1         120",
      "Et2       SW202                Ethernet2         120",
      "Et3       SW205                Ethernet3         120",
      "Et4       SW206                Ethernet1         120"
    ]
  }
}
```

textfsmで変数化

とりあえず変数にする

```
TASK [lldp : display results] *****
ok: [SW200] => {
  "parse_results": [
    {
      "LOCAL_INTERFACE": "-----",
      "NEIGHBOR": "-----",
      "NEIGHBOR_INTERFACE": "-----"
    },
    {
      "LOCAL_INTERFACE": "Et1",
      "NEIGHBOR": "SW201",
      "NEIGHBOR_INTERFACE": "Ethernet1"
    },
    {
      "LOCAL_INTERFACE": "Et2",
      "NEIGHBOR": "SW202",
      "NEIGHBOR_INTERFACE": "Ethernet2"
    },
    {
      "LOCAL_INTERFACE": "Et3",
      "NEIGHBOR": "SW205",
      "NEIGHBOR_INTERFACE": "Ethernet3"
    },
    {
      "LOCAL_INTERFACE": "Et4",
      "NEIGHBOR": "SW206",
      "NEIGHBOR_INTERFACE": "Ethernet1"
    }
  ]
}
ok: [SW201] => {
```

jinja2はこんな感じ

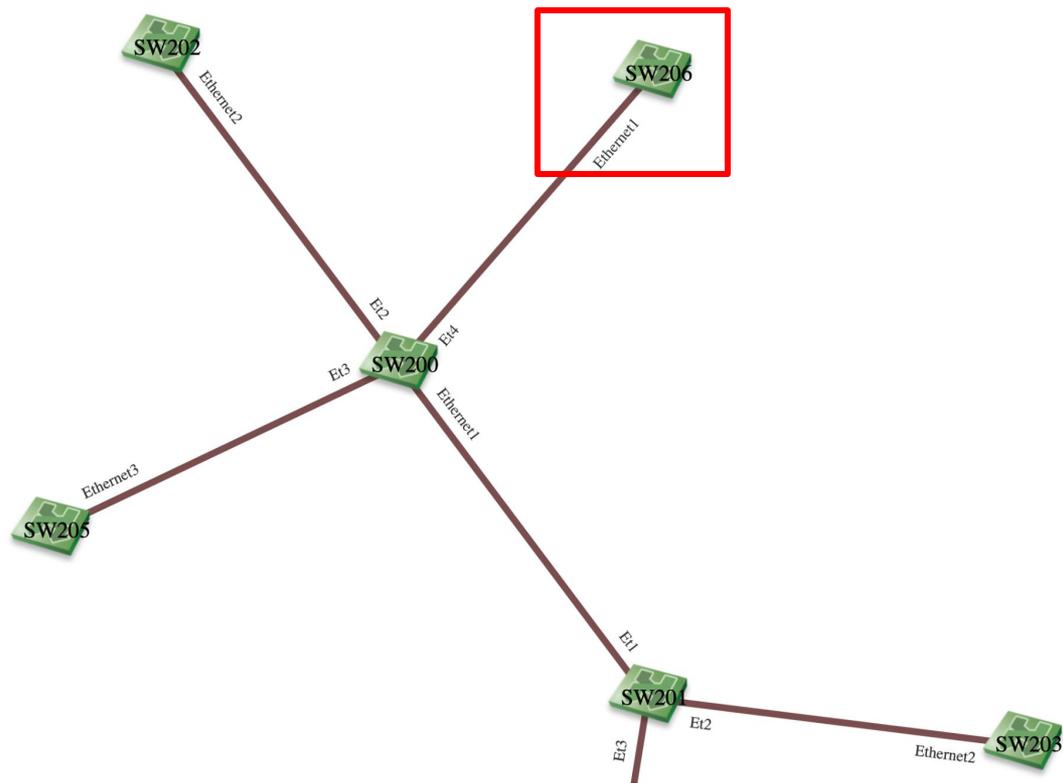
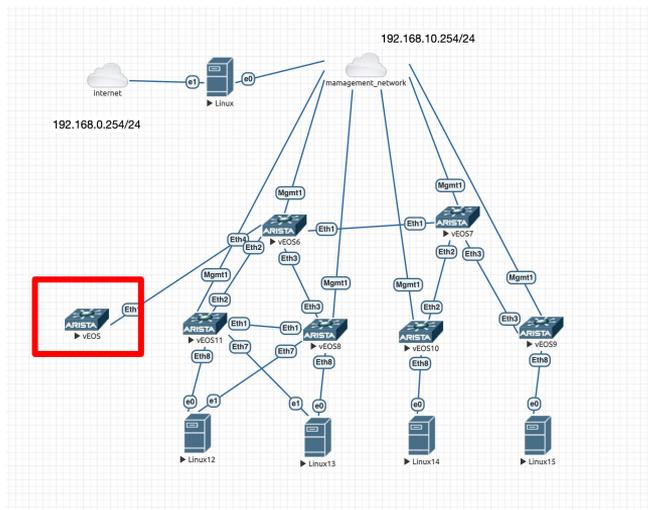
時間なくてかなり雑

inet-henge.j2 ×

roles > lldp > templates > inet-henge.j2

```
1 {
2   "nodes": [
3     {% for p in parse_results %}
4     {%- if loop.index > 1 and loop.index < parse_results|length %}
5
6     {
7       "name": "{{ p['NEIGHBOR'] }}",
8       "meta": { "description": "This is a switch", "type": "Backbone" },
9       "icon": "../images/switch.png"
10    },
11
12    {%- elif loop.index > 1 and loop.index == parse_results|length %}
13
14    {
15      "name": "{{ p['NEIGHBOR'] }}",
16      "meta": { "description": "This is a switch", "type": "Backbone" },
17      "icon": "../images/switch.png"
18    }
19
20    {%- endif %}
21  {% endfor %}
22
23  ],
24  "links": [
25
26    {% for p in parse_results %}
27    {%- if loop.index > 1 and loop.index < parse_results|length %}
28
29    {
30      "source": "{{ inventory_hostname }}"
```

ノード追加！！



まとめ

【感想】

- 素晴らしいものができました！
- 社内のネットワーク構成全部可視化してきます！
- 他部署/他社と時間取って開発する機会もらえてよかったです！
- EVEの環境に慣れるのに時間がかかった。

【他にもやりたかったこと。。。】

- WEBからポチっとansible実行したかった
- LLDPの情報をもう少し可視化したかった(show lldp detail)
- Netboxへの自動登録