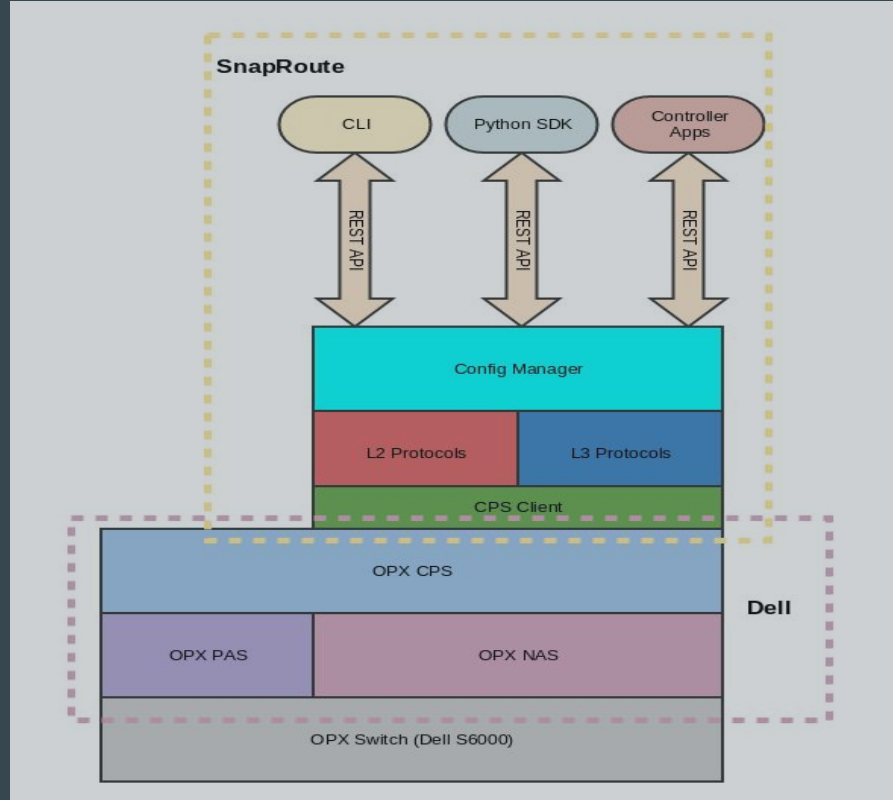


OPX-FlexSwitch Demo

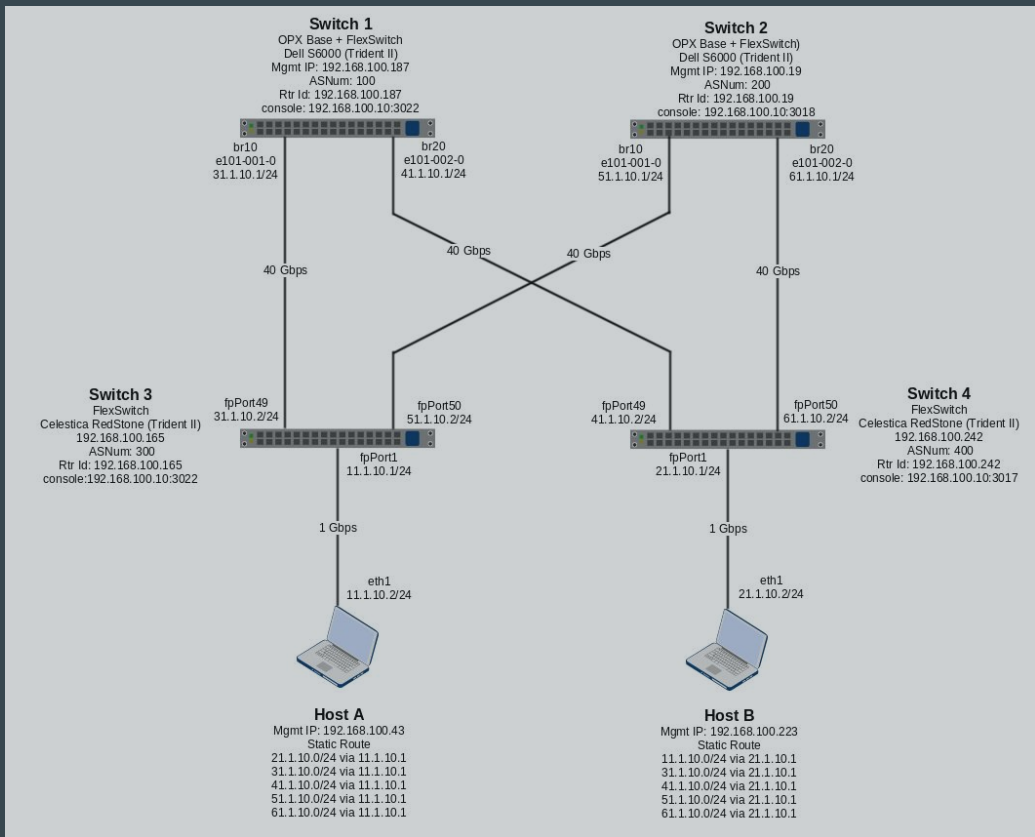


EBGPv4 Demo

OPX + FlexSwitch Architecture

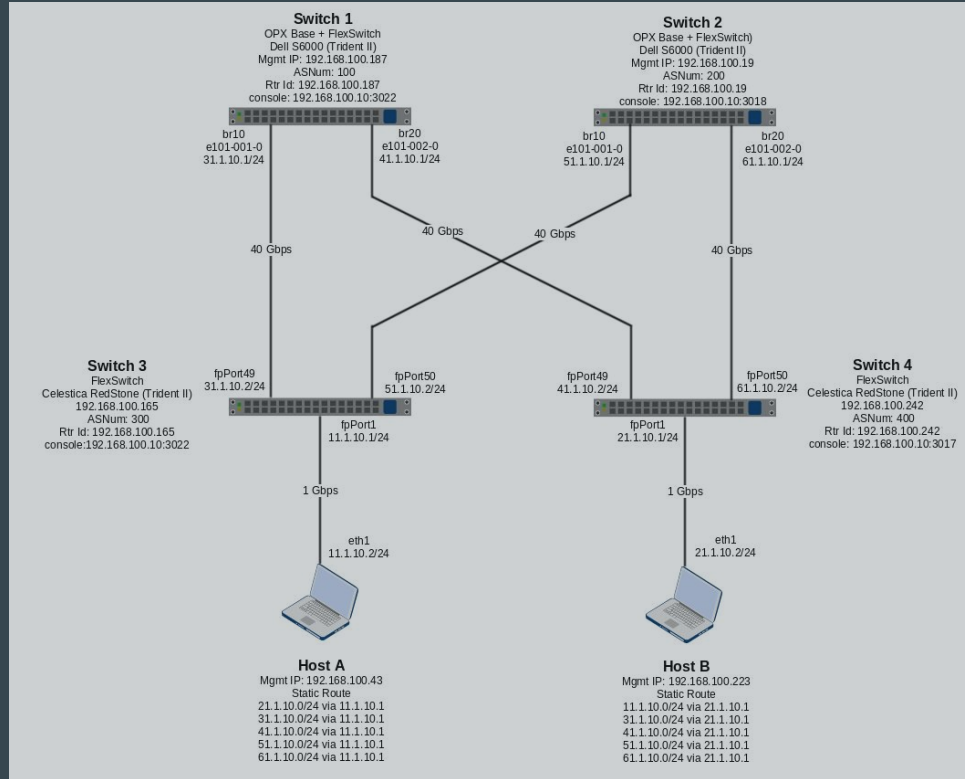


Demo Topology



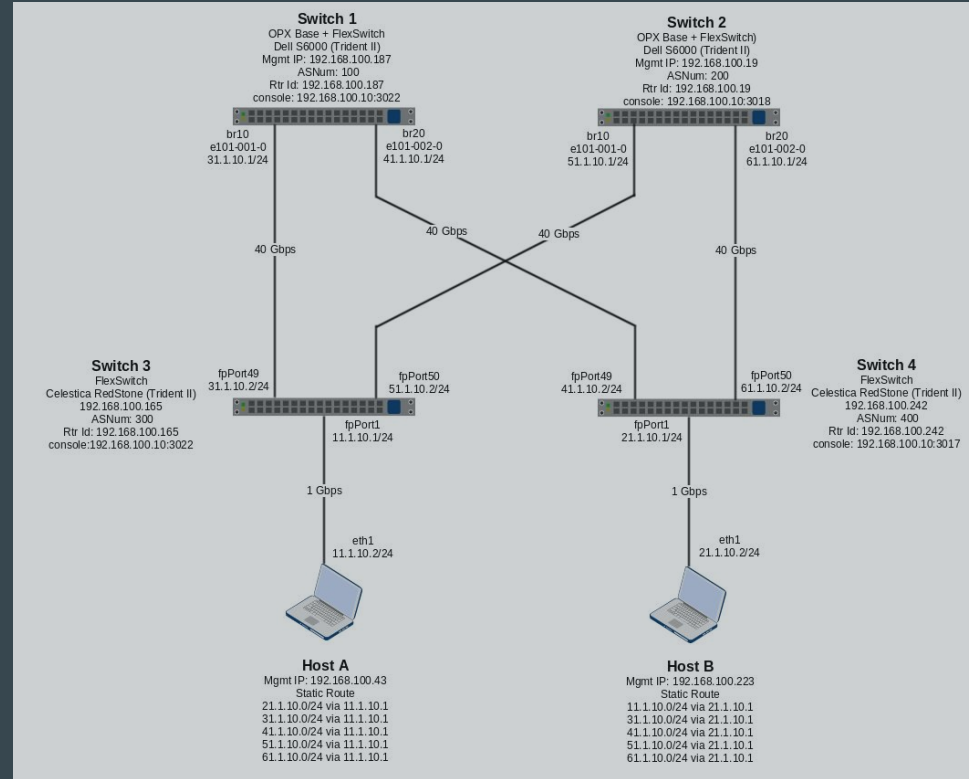
Switch 1 Configuration (Mgmt Ip: 192.168.100.19)

- **Setting Port:** e101-001-0, AdminState: UP
- **Setting Port:** e101-002-0, AdminState: UP
- **Creating Vlan:** 10, Untagged Interface List: [e101-001-0]
- **Creating Vlan:** 20, Untagged Interface List: [e101-002-0]
- **Assigning IPv4 Address** 31.1.10.1/24 on br10
- **Assigning IPv4 Address** 41.1.10.1/24 on br20
- **Creating BGP Global Instance** LocalAS: 100, RouterId: 192.168.100.187
- **Configuring BGPv4 Neighbor** Address: 31.1.10.2, PeerAS: 300, LocalAS: 100
- **Configuring BGPv4 Neighbor** Address: 41.1.10.2, PeerAS: 400, LocalAS: 100



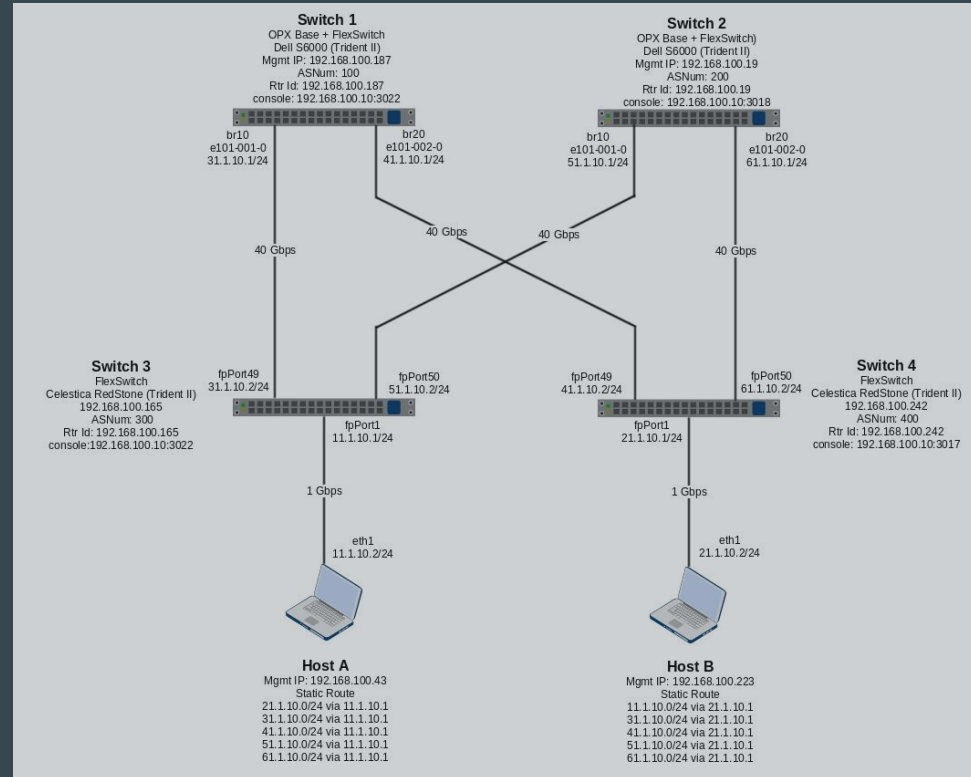
Switch 2 Configuration (Mgmt Ip: 192.168.100.19)

- **Setting Port:** e101-001-0, AdminState: UP
- **Setting Port:** e101-002-0, AdminState: UP
- **Creating Vlan:** 10, Untagged Interface List: [e101-001-0]
- **Creating Vlan:** 20, Untagged Interface List: [e101-002-0]
- **Assigning IPv4 Address** 51.1.10.1/24 on br10
- **Assigning IPv4 Address** 61.1.10.1/24 on br20
- **Creating BGP Global Instance** LocalAS: 200, RouterId: 192.168.100.19
- **Configuring BGPv4 Neighbor** Address: 51.1.10.2, PeerAS: 300, LocalAS: 200
- **Configuring BGPv4 Neighbor** Address: 61.1.10.2, PeerAS: 400, LocalAS: 200



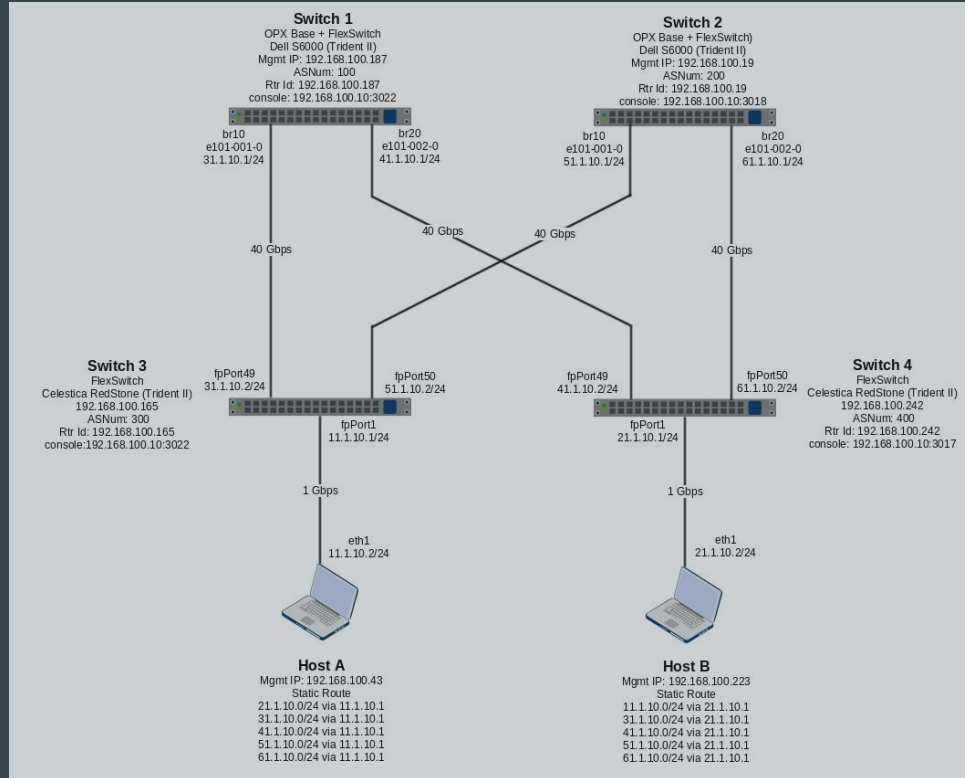
Switch 3 Configuration (Mgmt Ip: 192.168.100.165)

- **Setting Port:** fpPort49 AdminState: UP, Speed: 40000
- **Setting Port:** fpPort50 AdminState: UP, Speed: 40000
- **Setting Port:** fpPort1 AdminState: UP, Speed: 1000
- **Assigning IPv4 Address** 31.1.10.2/24 on fpPort49
- **Assigning IPv4 Address** 51.1.10.2/24 on fpPort50
- **Assigning IPv4 Address** 11.1.10.1/24 on fpPort1
- **Creating Policy :** Redistribute Connected
- **Creating BGP Global Instance** LocalAS: 300, RouterId: 192.168.100.165
- **Configuring BGPv4 Neighbor** Address: 31.1.10.1, PeerAS: 100, LocalAS: 300
- **Configuring BGPv4 Neighbor** Address: 51.1.10.1, PeerAS: 200, LocalAS: 300



Switch 4 Configuration (Mgmt Ip: 192.168.100.242)

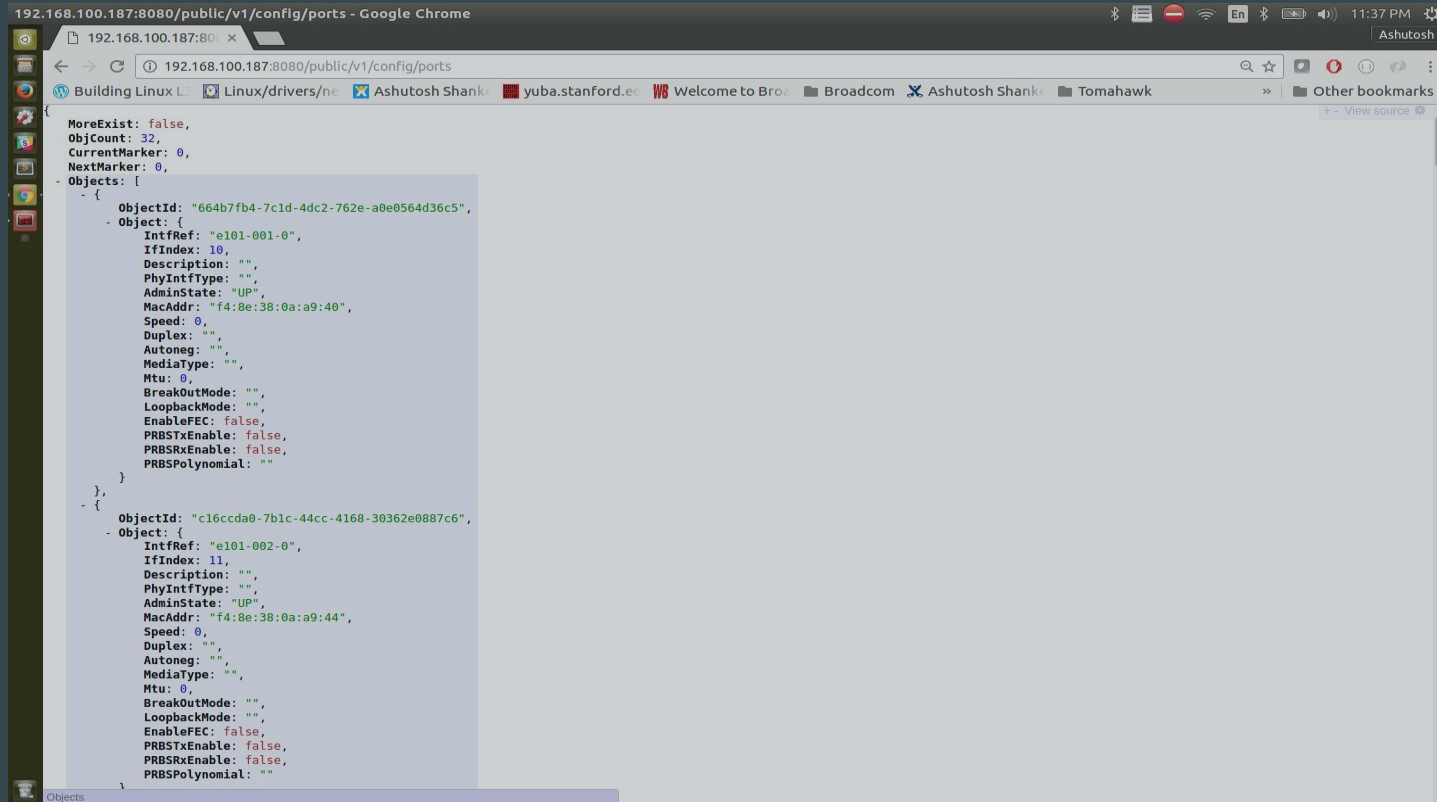
- **Setting Port:** fpPort49, AdminState: UP, Speed: 40000
- **Setting Port:** fpPort50, AdminState: UP, Speed: 40000
- **Setting Port:** fpPort1, AdminState: UP, Speed: 1000
- **Assigning IPv4 Address** 41.1.10.2/24 on fpPort49
- **Assigning IPv4 Address** 61.1.10.2/24 on fpPort50
- **Assigning IPv4 Address** 21.1.10.1/24 on fpPort1
- **Creating Policy:** Redistribute Connected
- **Creating BGP Global Instance** LocalAS: 400, RouterId: 192.168.100.242
- **Configuring BGPv4 Neighbor** Address: 41.1.10.1, PeerAS: 100, LocalAS: 400
- **Configuring BGPv4 Neighbor** Address: 61.1.10.1, PeerAS: 200, LocalAS: 400



Example Configuration

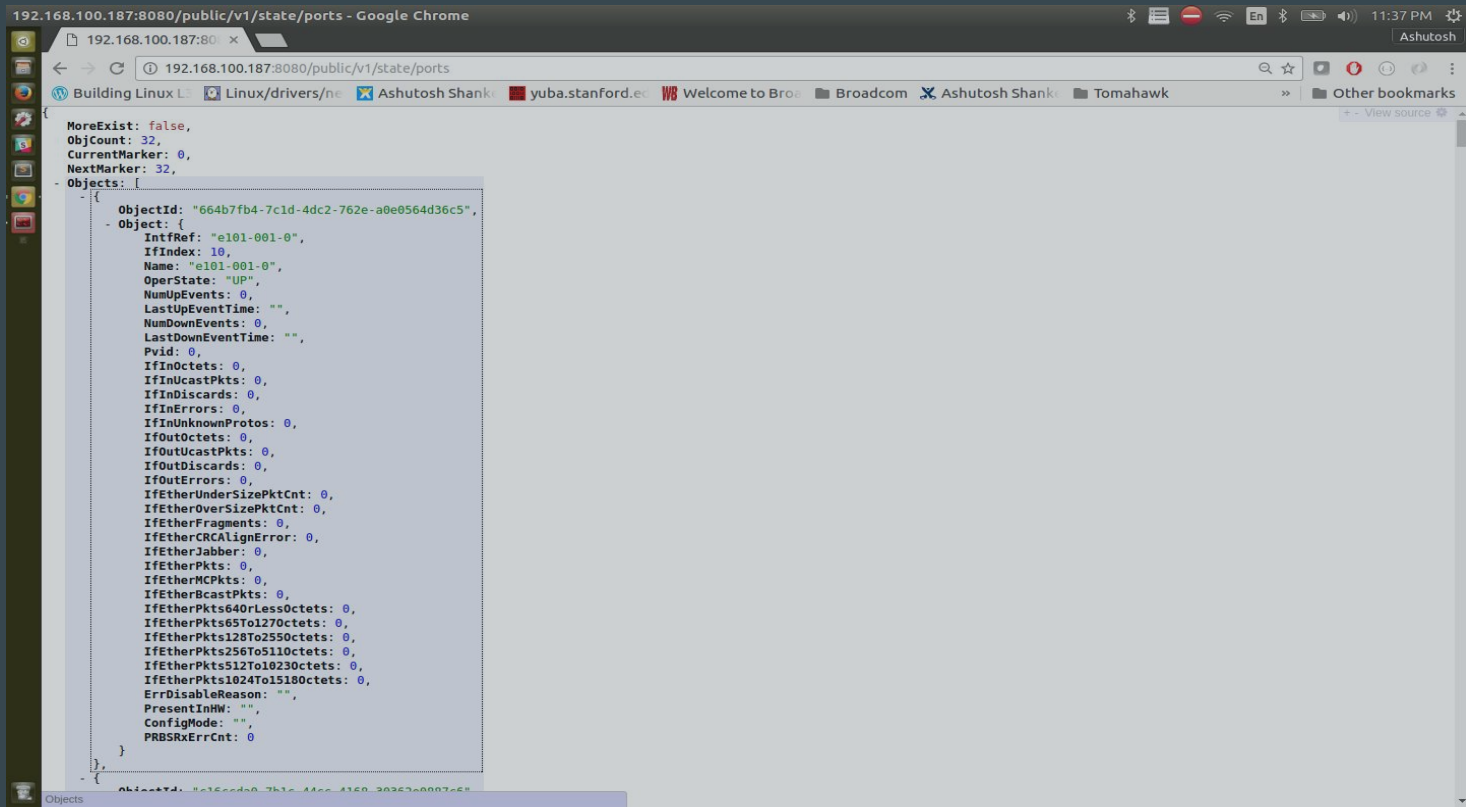
- `curl -X PATCH -H "Content-Type: application/json" -d '{"IntfRef": "e101-001-0", "AdminState": "UP"}'`
`http://192.168.100.187:8080/public/v1/config/Port`
- `curl -H "Content-Type: application/json" -d '{"VlanId": 10, "UntagIntfList": ["e101-001-0"]}'`
`http://192.168.100.187:8080/public/v1/config/Vlan`
- `curl -H "Content-Type: application/json" -d '{"IpAddr": "31.1.10.1/24", "IntfRef": "br10"}'`
`http://192.168.100.187:8080/public/v1/config/IPv4Intf`
- `curl -H "Content-Type: application/json" -d '{"Vrf": "default", "ASNum": "100", "RouterId": "192.168.100.187"}'`
`-X PATCH http://192.168.100.187:8080/public/v1/config/bgpglobal`
- `curl -H "Content-Type: application/json" -d '{"NeighborAddress": "31.1.10.2", "IntfRef": "", "PeerAS": "300", "LocalAS": "100", "ConnectRetryTime": 30, "HoldTime": 3, "KeepaliveTime": 1}' -X POST`
`http://192.168.100.187:8080/public/v1/config/bgpv4neighbor`

Port Configuration



```
192.168.100.187:8080/public/v1/config/ports - Google Chrome
192.168.100.187:8080/public/v1/config/ports
Building Linux L... Linux/drivers/ne... Ashutosh Shank... yuba.stanford.edu Welcome to Bro... Broadcom Ashutosh Shank... Tomahawk
MoreExist: false,
ObjCount: 32,
CurrentMarker: 0,
NextMarker: 0,
Objects: [
  {
    ObjectId: "664b7fb4-7c1d-4dc2-762e-a0e0564d36c5",
    Object: {
      IntfRef: "e101-001-0",
      IfIndex: 10,
      Description: "",
      PhyIntfType: "",
      AdminState: "UP",
      MacAddr: "f4:8e:38:0a:a9:40",
      Speed: 0,
      Duplex: "",
      Autoneg: "",
      MediaType: "",
      Mtu: 0,
      BreakOutMode: "",
      LoopbackMode: "",
      EnableFEC: false,
      PRBSTxEnable: false,
      PRBSRxEnable: false,
      PRBSPolynomial: ""
    }
  },
  {
    ObjectId: "c16ccda0-7b1c-44cc-4168-30362e0887c6",
    Object: {
      IntfRef: "e101-002-0",
      IfIndex: 11,
      Description: "",
      PhyIntfType: "",
      AdminState: "UP",
      MacAddr: "f4:8e:38:0a:a9:44",
      Speed: 0,
      Duplex: "",
      Autoneg: "",
      MediaType: "",
      Mtu: 0,
      BreakOutMode: "",
      LoopbackMode: "",
      EnableFEC: false,
      PRBSTxEnable: false,
      PRBSRxEnable: false,
      PRBSPolynomial: ""
    }
  }
]
```

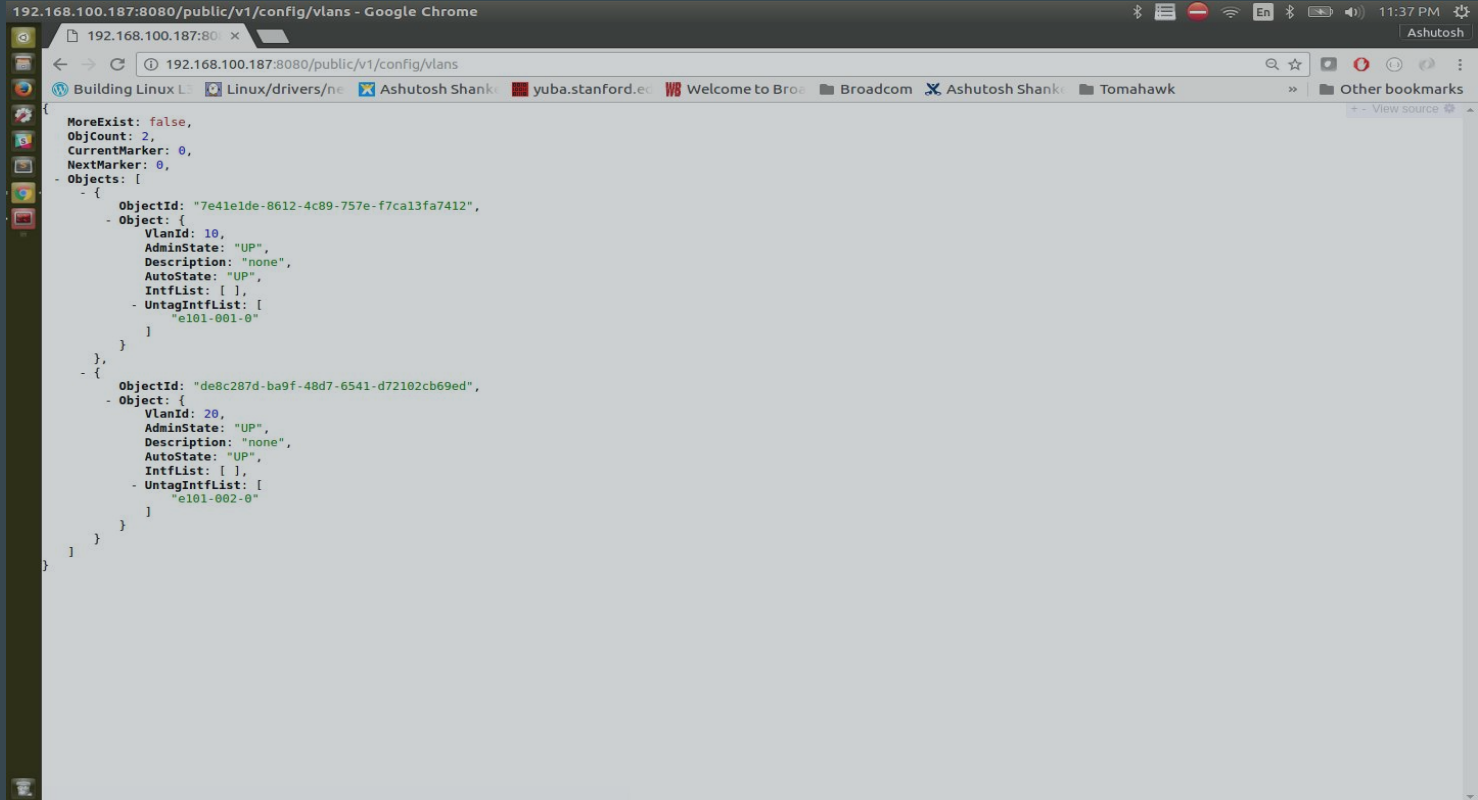
Port State



The screenshot shows a web browser window with the address bar displaying `192.168.100.187:8080/public/v1/state/ports`. The page content shows the state of a network interface, including the number of objects and their details.

```
MoreExist: false,
ObjCount: 32,
CurrentMarker: 0,
NextMarker: 32,
- Objects: [
  - {
    ObjectId: "664b7fb4-7c1d-4dc2-762e-a0e0564d36c5",
    - Object: {
      IntfRef: "e101-001-0",
      IfIndex: 10,
      Name: "e101-001-0",
      OperState: "UP",
      NumUpEvents: 0,
      LastUpEventTime: "",
      NumDownEvents: 0,
      LastDownEventTime: "",
      Pvid: 0,
      IfInOctets: 0,
      IfInUcastPkts: 0,
      IfInDiscards: 0,
      IfInErrors: 0,
      IfInUnknownProtos: 0,
      IfOutOctets: 0,
      IfOutUcastPkts: 0,
      IfOutDiscards: 0,
      IfOutErrors: 0,
      IfEtherUnderSizePktCnt: 0,
      IfEtherOverSizePktCnt: 0,
      IfEtherFragments: 0,
      IfEtherCRCAlignError: 0,
      IfEtherJabber: 0,
      IfEtherPkts: 0,
      IfEtherMCPkts: 0,
      IfEtherBcastPkts: 0,
      IfEtherPkts64orLessOctets: 0,
      IfEtherPkts65To127Octets: 0,
      IfEtherPkts128To255Octets: 0,
      IfEtherPkts256To511Octets: 0,
      IfEtherPkts512To1023Octets: 0,
      IfEtherPkts1024To1518Octets: 0,
      ErrDisableReason: "",
      PresentInHW: "",
      ConfigMode: "",
      PRBSRxErrCnt: 0
    }
  },
  - {
    ObjectId: "a1c6cda0-7b1c-416a-a160-3a267a0007e6"
```

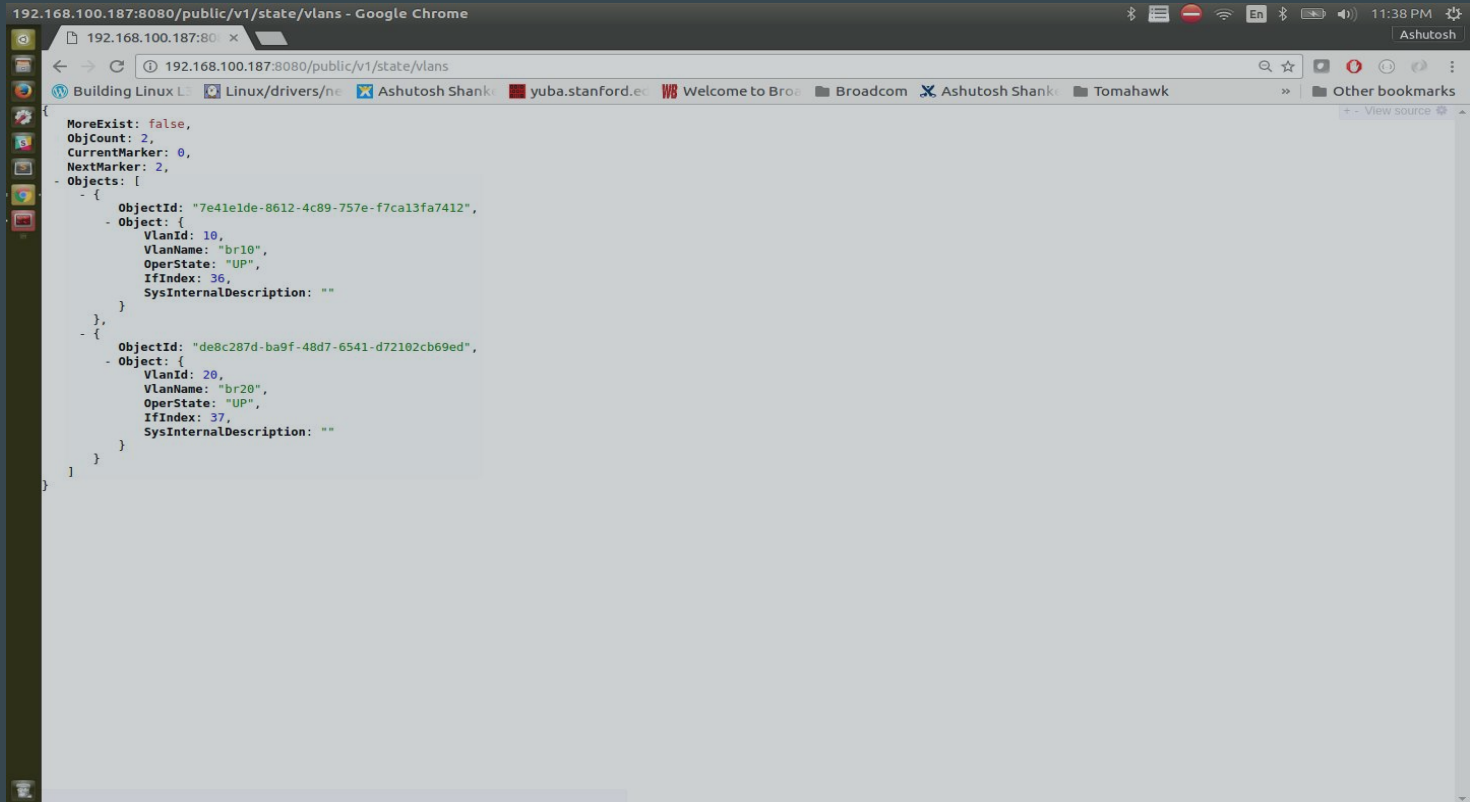
Vlan Configuration



The screenshot shows a Google Chrome browser window with the address bar displaying `192.168.100.187:8080/public/v1/config/vlans`. The page content is a JSON response from a REST API, showing two VLAN objects. The first object has `VlanId: 10` and the second has `VlanId: 20`. Both are in `AdminState: "UP"` and `AutoState: "UP"` with `Description: "none"`. The `IntfList` is an empty array, and the `UntagIntfList` contains the value `"e101-001-0"` for the first VLAN and `"e101-002-0"` for the second.

```
{
  MoreExist: false,
  ObjCount: 2,
  CurrentMarker: 0,
  NextMarker: 0,
  Objects: [
    - {
      ObjectId: "7e41e1de-8612-4c89-757e-f7ca13fa7412",
      Object: {
        VlanId: 10,
        AdminState: "UP",
        Description: "none",
        AutoState: "UP",
        IntfList: [ ],
        UntagIntfList: [
          "e101-001-0"
        ]
      }
    },
    - {
      ObjectId: "de0c287d-ba9f-48d7-6541-d72102cb69ed",
      Object: {
        VlanId: 20,
        AdminState: "UP",
        Description: "none",
        AutoState: "UP",
        IntfList: [ ],
        UntagIntfList: [
          "e101-002-0"
        ]
      }
    }
  ]
}
```

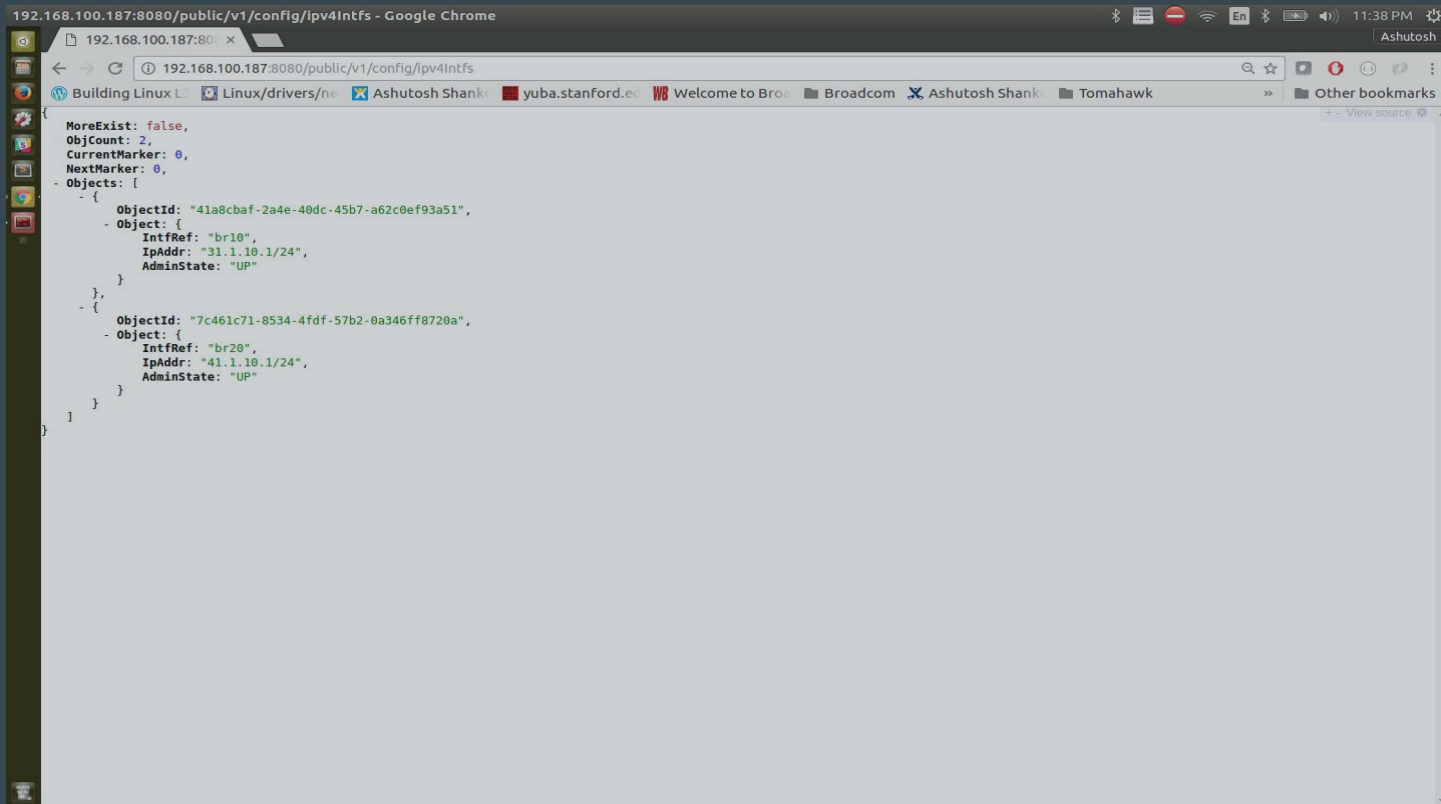
Vlan State



The screenshot shows a Google Chrome browser window with the address bar displaying `192.168.100.187:8080/public/v1/state/vlans`. The page content is a JSON response from a REST API, showing the state of two VLANs. The JSON is formatted with syntax highlighting. The browser's address bar also shows the URL `192.168.100.187:8080/public/v1/state/vlans`. The browser's tab bar shows several tabs, including "Building Linux L...", "Linux/drivers/ne...", "Ashutosh Shank...", "yuba.stanford.ec...", "Welcome to Bro...", "Broadcom", "Ashutosh Shank...", and "Tomahawk". The browser's status bar shows the time as 11:38 PM and the user's name as Ashutosh.

```
{
  "MoreExist": false,
  "ObjCount": 2,
  "CurrentMarker": 0,
  "NextMarker": 2,
  "Objects": [
    {
      "ObjectId": "7e41e1de-8612-4c89-757e-f7ca13fa7412",
      "Object": {
        "VlanId": 10,
        "VlanName": "br10",
        "OperState": "UP",
        "IfIndex": 36,
        "SysInternalDescription": ""
      }
    },
    {
      "ObjectId": "de8c287d-ba9f-48d7-6541-d72102cb69ed",
      "Object": {
        "VlanId": 20,
        "VlanName": "br20",
        "OperState": "UP",
        "IfIndex": 37,
        "SysInternalDescription": ""
      }
    }
  ]
}
```

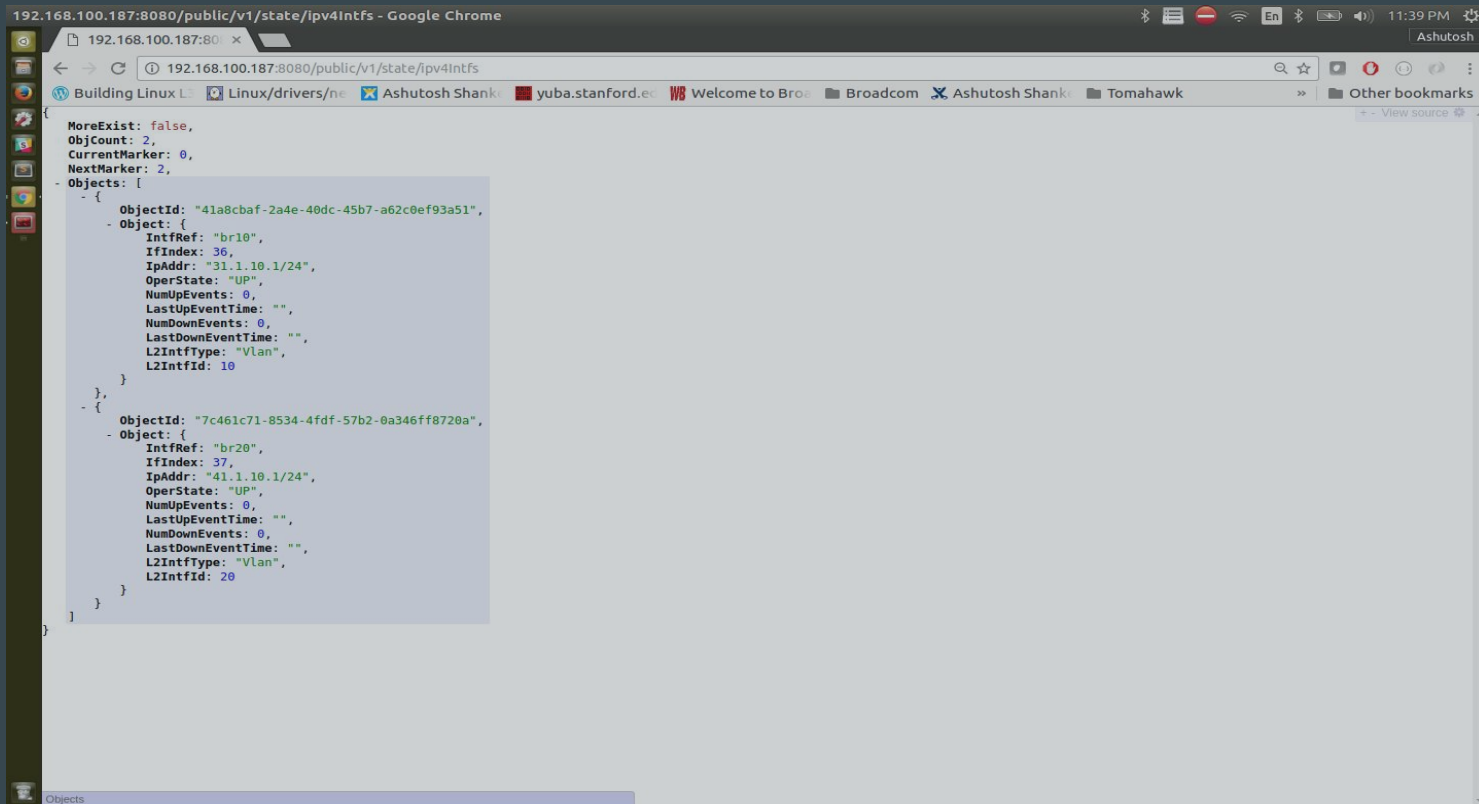
IPv4 Interface Configuration



The screenshot shows a Google Chrome browser window with the address bar displaying `192.168.100.187:8080/public/v1/config/ipv4Intfs`. The page content displays a JSON configuration for IPv4 interfaces. The configuration includes a `MoreExist` flag set to `false`, an `ObjCount` of `2`, a `CurrentMarker` of `0`, and a `NextMarker` of `0`. The `Objects` array contains two interface objects. The first object has an `ObjectId` of `"41a8cbaf-2a4e-40dc-45b7-a62c0ef93a51"`, an `IntfRef` of `"br10"`, an `IpAddr` of `"31.1.10.1/24"`, and an `AdminState` of `"UP"`. The second object has an `ObjectId` of `"7c461c71-8534-4fdf-57b2-0a346ff8720a"`, an `IntfRef` of `"br20"`, an `IpAddr` of `"41.1.10.1/24"`, and an `AdminState` of `"UP"`.

```
{
  MoreExist: false,
  ObjCount: 2,
  CurrentMarker: 0,
  NextMarker: 0,
  Objects: [
    {
      ObjectId: "41a8cbaf-2a4e-40dc-45b7-a62c0ef93a51",
      Object: {
        IntfRef: "br10",
        IpAddr: "31.1.10.1/24",
        AdminState: "UP"
      }
    },
    {
      ObjectId: "7c461c71-8534-4fdf-57b2-0a346ff8720a",
      Object: {
        IntfRef: "br20",
        IpAddr: "41.1.10.1/24",
        AdminState: "UP"
      }
    }
  ]
}
```

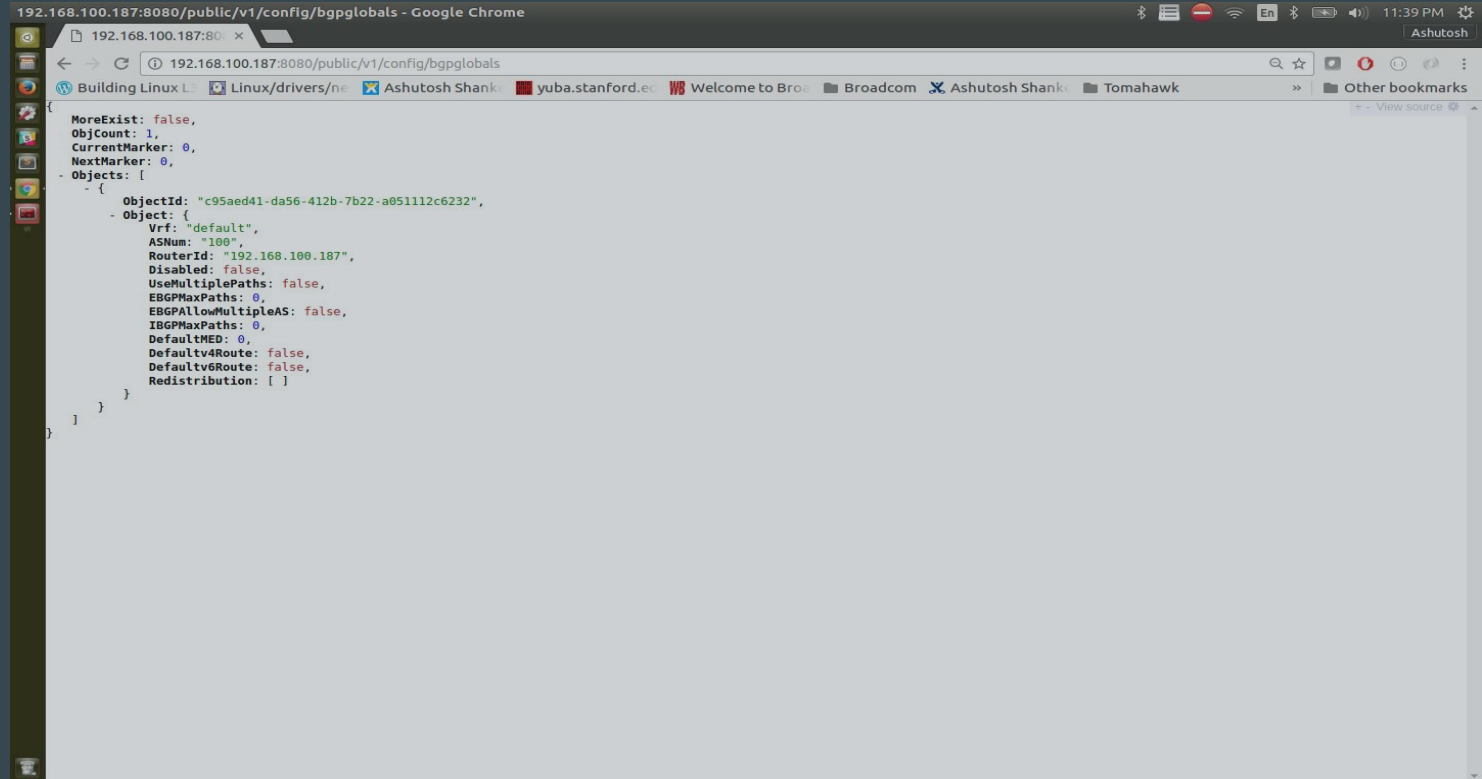
IPv4 Interface State



The screenshot shows a web browser window with the address bar displaying `192.168.100.187:8080/public/v1/state/ipv4Intfs`. The browser's address bar and tabs are visible at the top. The main content area displays a JSON object representing the state of IPv4 interfaces. The JSON is highlighted with a light blue background. The JSON structure includes a `MoreExist` flag, `ObjCount`, `CurrentMarker`, `NextMarker`, and an `Objects` array. The `Objects` array contains two interface objects, each with `ObjectId`, `Object` details (including `IntfRef`, `IfIndex`, `IpAddr`, `OperState`, `NumUpEvents`, `LastUpEventTime`, `NumDownEvents`, `LastDownEventTime`, `L2IntfType`, and `L2IntfId`).

```
{
  "MoreExist": false,
  "ObjCount": 2,
  "CurrentMarker": 0,
  "NextMarker": 2,
  "Objects": [
    {
      "ObjectId": "41a8cbaf-2a4e-40dc-45b7-a62c0ef93a51",
      "Object": {
        "IntfRef": "br10",
        "IfIndex": 36,
        "IpAddr": "31.1.10.1/24",
        "OperState": "UP",
        "NumUpEvents": 0,
        "LastUpEventTime": "",
        "NumDownEvents": 0,
        "LastDownEventTime": "",
        "L2IntfType": "Vlan",
        "L2IntfId": 10
      }
    },
    {
      "ObjectId": "7c461c71-8534-4fdf-57b2-0a346ff8720a",
      "Object": {
        "IntfRef": "br20",
        "IfIndex": 37,
        "IpAddr": "41.1.10.1/24",
        "OperState": "UP",
        "NumUpEvents": 0,
        "LastUpEventTime": "",
        "NumDownEvents": 0,
        "LastDownEventTime": "",
        "L2IntfType": "Vlan",
        "L2IntfId": 20
      }
    }
  ]
}
```

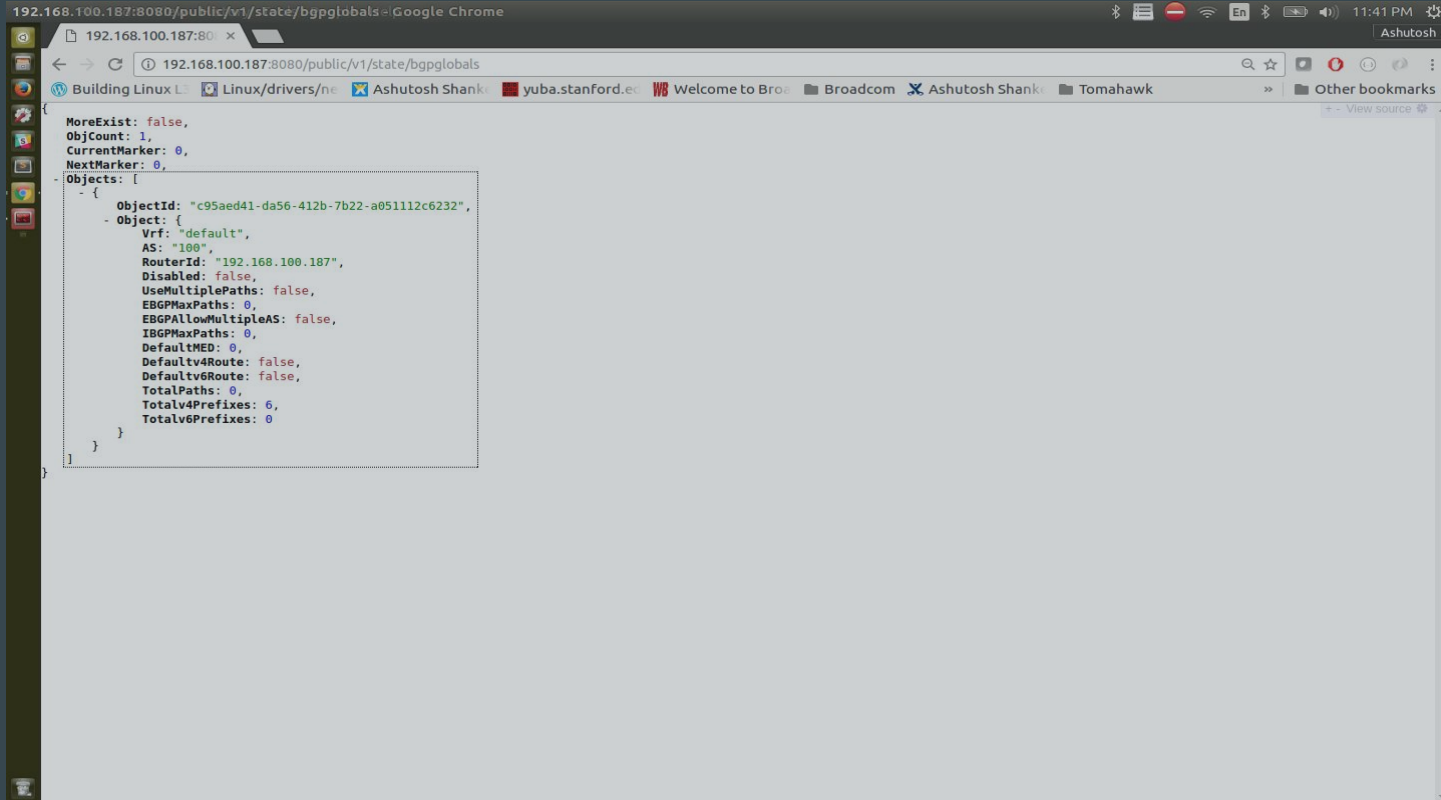
BGP Global Configuration



The screenshot shows a Google Chrome browser window with the address bar displaying `192.168.100.187:8080/public/v1/config/bgpglobals`. The page content displays a JSON configuration for BGP global settings. The configuration includes fields for `MoreExist`, `ObjCount`, `CurrentMarker`, `NextMarker`, and an array of `Objects`. The first object in the array has an `ObjectId` of `"c95aed41-da56-412b-7b22-a051112c6232"` and an `Object` containing various BGP parameters such as `Vrf`, `ASNum`, `RouterId`, `Disabled`, `UseMultiplePaths`, `EBGPMaxPaths`, `EBGPAllowMultipleAS`, `IBGPMaxPaths`, `DefaultMED`, `Defaultv4Route`, `Defaultv6Route`, and `Redistribution`.

```
{
  MoreExist: false,
  ObjCount: 1,
  CurrentMarker: 0,
  NextMarker: 0,
  Objects: [
    {
      ObjectId: "c95aed41-da56-412b-7b22-a051112c6232",
      Object: {
        Vrf: "default",
        ASNum: "100",
        RouterId: "192.168.100.187",
        Disabled: false,
        UseMultiplePaths: false,
        EBGPMaxPaths: 0,
        EBGPAllowMultipleAS: false,
        IBGPMaxPaths: 0,
        DefaultMED: 0,
        Defaultv4Route: false,
        Defaultv6Route: false,
        Redistribution: [ ]
      }
    }
  ]
}
```

BGP Global State

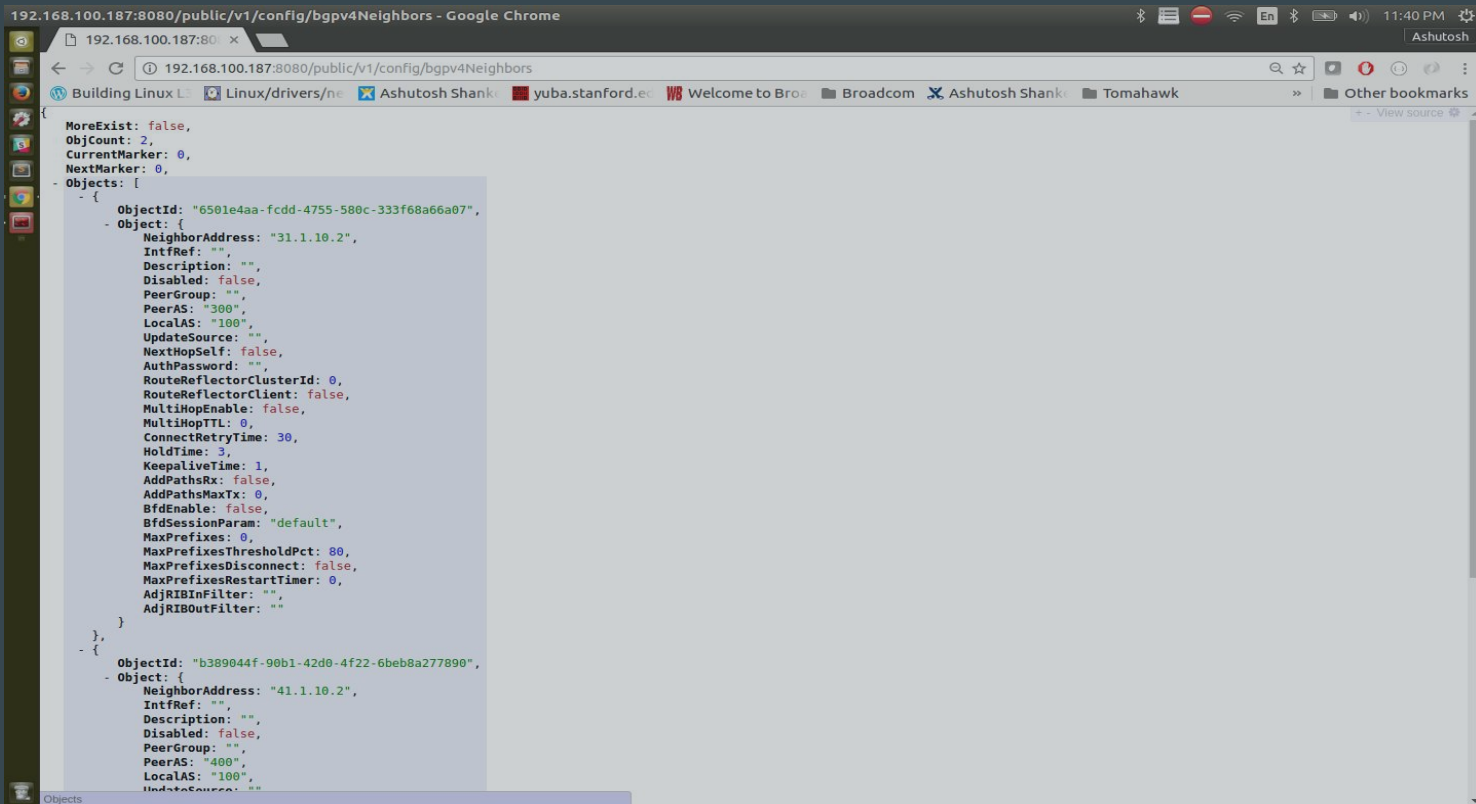


The screenshot shows a web browser window with the address bar displaying `192.168.100.187:8080/public/v1/state/bgpglobals`. The page content displays a JSON object representing the BGP Global State. The JSON is as follows:

```
{
  "MoreExist": false,
  "ObjCount": 1,
  "CurrentMarker": 0,
  "NextMarker": 0,
  "Objects": [
    {
      "ObjectId": "c95aed41-da56-412b-7b22-a051112c6232",
      "Object": {
        "Vrf": "default",
        "AS": "100",
        "RouterId": "192.168.100.187",
        "Disabled": false,
        "UseMultiplePaths": false,
        "EBGPMaxPaths": 0,
        "EBGPAllowMultipleAS": false,
        "IBGPMaxPaths": 0,
        "DefaultMED": 0,
        "Defaultv4Route": false,
        "Defaultv6Route": false,
        "TotalPaths": 0,
        "Totalv4Prefixes": 6,
        "Totalv6Prefixes": 0
      }
    }
  ]
}
```

The browser's address bar also shows the URL `192.168.100.187:8080/public/v1/state/bgpglobals`. The browser's tab bar shows several tabs, including "Building Linux L...", "Linux/drivers/ne...", "Ashutosh Shank...", "yuba.stanford.edu", "Welcome to Bro...", "Broadcom", "Ashutosh Shank...", and "Tomahawk". The browser's status bar shows the time "11:41 PM" and the user's name "Ashutosh".

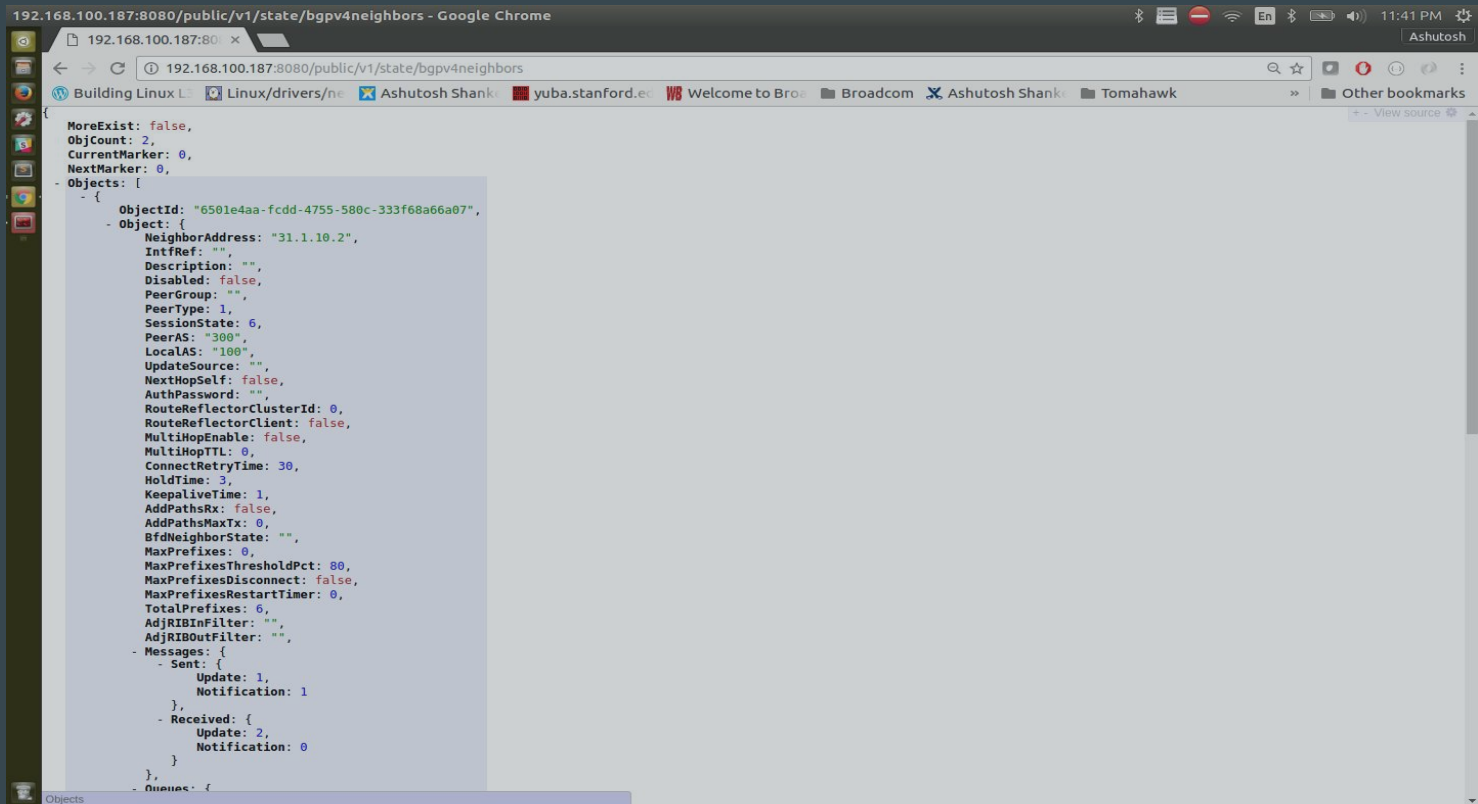
BGP Neighbor Configuration



The screenshot shows a Google Chrome browser window with the address bar displaying `192.168.100.187:8080/public/v1/config/bgpv4Neighbors`. The page content displays a JSON configuration for BGP neighbors. The configuration includes a list of objects, each representing a neighbor with various attributes such as `NeighborAddress`, `PeerAS`, `LocalAS`, and `UpdateSource`.

```
{
  MoreExist: false,
  ObjCount: 2,
  CurrentMarker: 0,
  NextMarker: 0,
  Objects: [
    {
      ObjectId: "6501e4aa-fcdd-4755-580c-333f68a66a07",
      Object: {
        NeighborAddress: "31.1.10.2",
        IntfRef: "",
        Description: "",
        Disabled: false,
        PeerGroup: "",
        PeerAS: "300",
        LocalAS: "100",
        UpdateSource: "",
        NextHopSelf: false,
        AuthPassword: "",
        RouteReflectorClusterId: 0,
        RouteReflectorClient: false,
        MultiHopEnable: false,
        MultiHopTTL: 0,
        ConnectRetryTime: 30,
        HoldTime: 3,
        KeepaliveTime: 1,
        AddPathsRx: false,
        AddPathsMaxTx: 0,
        BfdEnable: false,
        BfdSessionParam: "default",
        MaxPrefixes: 0,
        MaxPrefixesThresholdPct: 80,
        MaxPrefixesDisconnect: false,
        MaxPrefixesRestartTimer: 0,
        AdjRIBInFilter: "",
        AdjRIBOutFilter: ""
      }
    },
    {
      ObjectId: "b389044f-90b1-42d0-4f22-6beb8a277890",
      Object: {
        NeighborAddress: "41.1.10.2",
        IntfRef: "",
        Description: "",
        Disabled: false,
        PeerGroup: "",
        PeerAS: "400",
        LocalAS: "100",
        UpdateSource: ""
      }
    }
  ]
}
```

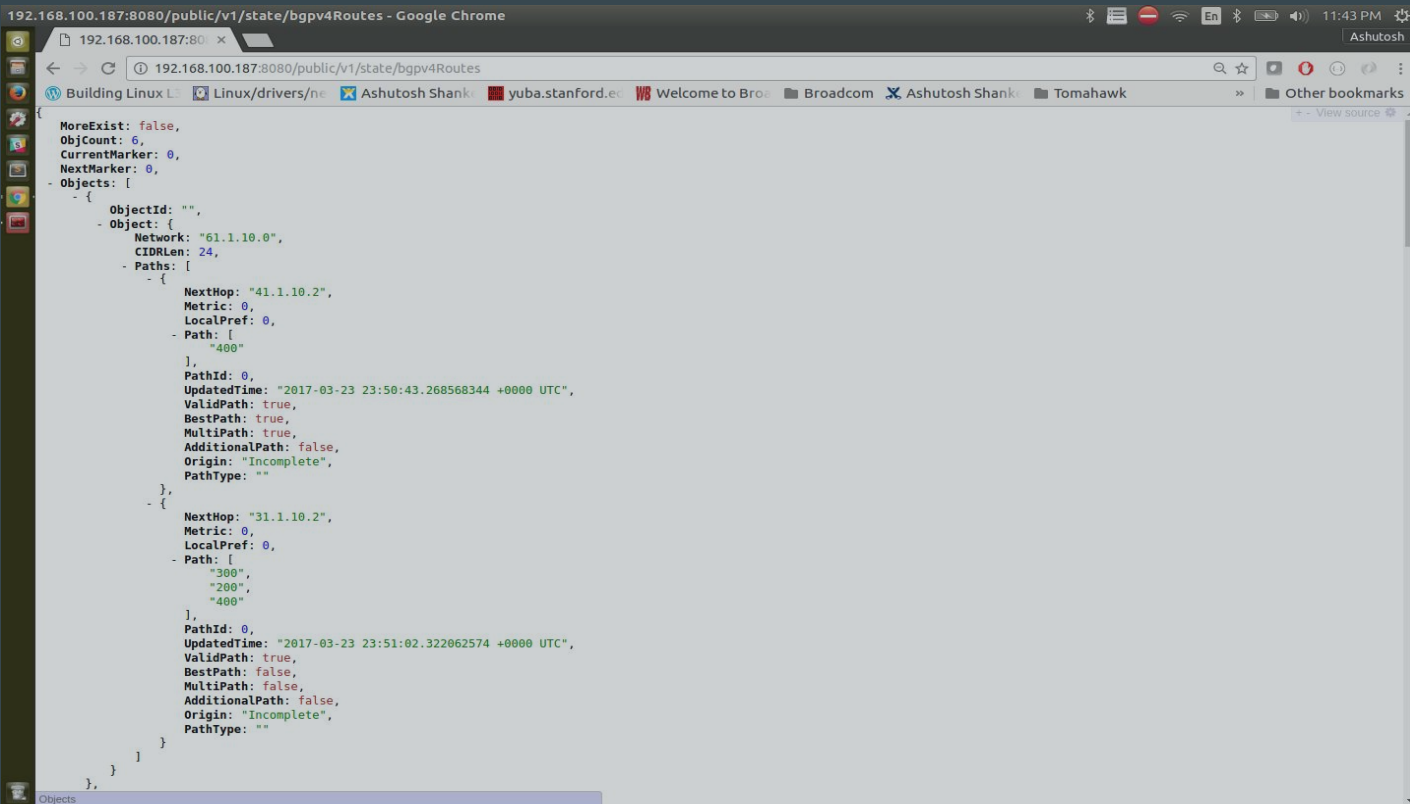
BGP Neighbor State



The screenshot shows a web browser window with the address bar displaying `192.168.100.187:8080/public/v1/state/bgpv4neighbors`. The page content displays a JSON object representing BGP neighbor state. The JSON is partially expanded, showing the `Objects` array. The first object in the array is a BGP neighbor configuration for IP `31.1.10.2`.

```
{
  "MoreExist": false,
  "ObjCount": 2,
  "CurrentMarker": 0,
  "NextMarker": 0,
  "Objects": [
    {
      "ObjectId": "6501e4aa-fcdd-4755-580c-333f68a66a07",
      "Object": {
        "NeighborAddress": "31.1.10.2",
        "IntfRef": "",
        "Description": "",
        "Disabled": false,
        "PeerGroup": "",
        "PeerType": 1,
        "SessionState": 6,
        "PeerAS": "300",
        "LocalAS": "100",
        "UpdateSource": "",
        "NextHopSelf": false,
        "AuthPassword": "",
        "RouteReflectorClusterId": 0,
        "RouteReflectorClient": false,
        "MultiHopEnable": false,
        "MultiHopTTL": 0,
        "ConnectRetryTime": 30,
        "HoldTime": 3,
        "KeepaliveTime": 1,
        "AddPathsRx": false,
        "AddPathsMaxTx": 0,
        "BfdNeighborState": "",
        "MaxPrefixes": 0,
        "MaxPrefixesThresholdPct": 80,
        "MaxPrefixesDisconnect": false,
        "MaxPrefixesRestartTimer": 0,
        "TotalPrefixes": 6,
        "AdjRIBInFilter": "",
        "AdjRIBOutFilter": "",
        "Messages": {
          "Sent": {
            "Update": 1,
            "Notification": 1
          },
          "Received": {
            "Update": 2,
            "Notification": 0
          }
        }
      }
    }
  ]
}
```

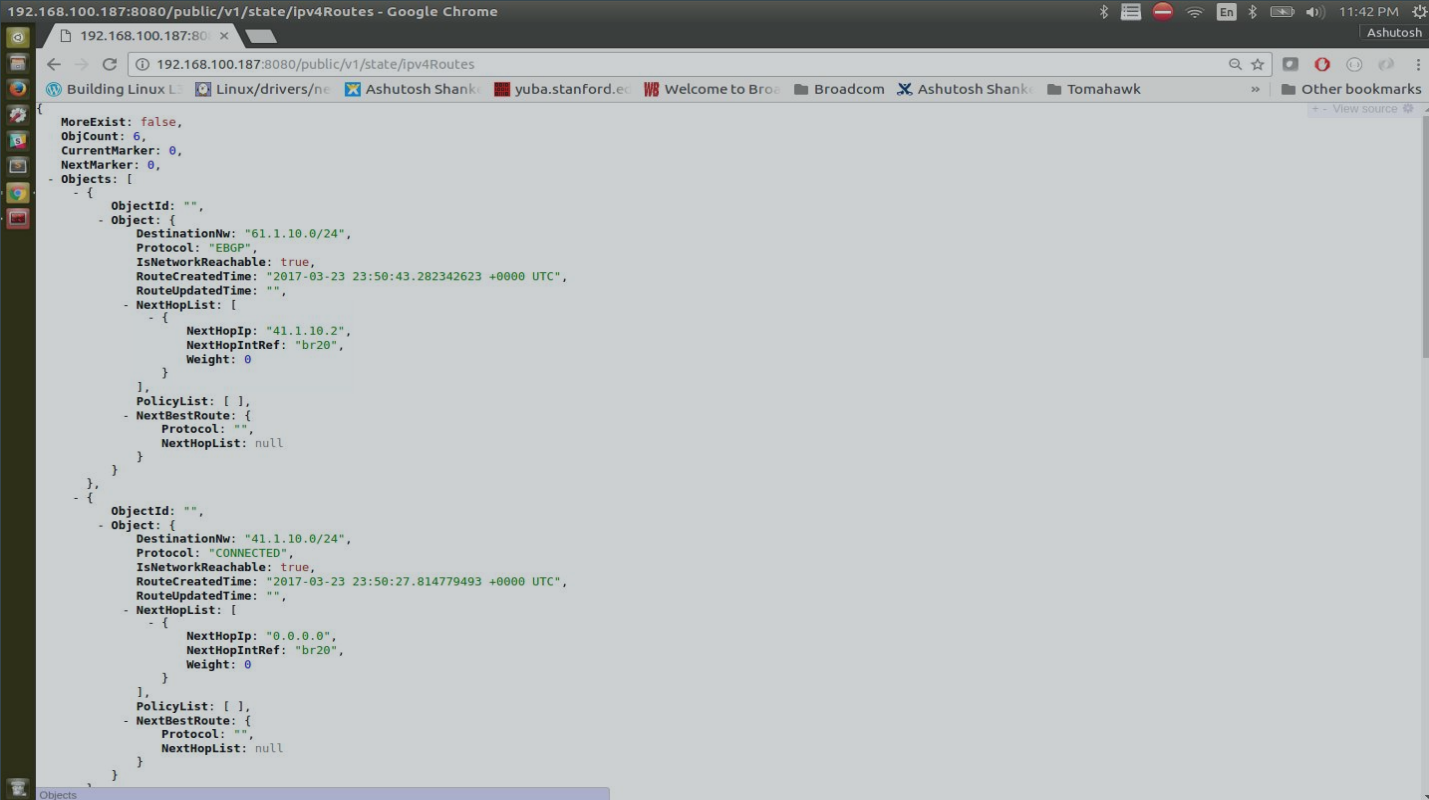
BGP Route State



The screenshot shows a web browser window with the address bar at 192.168.100.187:8080/public/v1/state/bgpv4Routes. The page displays a JSON object representing BGP route state. The object has several fields: MoreExist (false), ObjCount (6), CurrentMarker (0), NextMarker (0), and Objects (an array of route objects). The first route object in the array has an ObjectId of "", a Network of "61.1.10.0", a CIDRLen of 24, and a single Path with NextHop "41.1.10.2", Metric 0, LocalPref 0, PathId 0, and an UpdatedTime of "2017-03-23 23:50:43.268568344 +0000 UTC". The second route object has a NextHop of "31.1.10.2", Metric 0, LocalPref 0, and three Paths with NextHops "300", "200", and "400". The second route object also has a PathId of 0 and an UpdatedTime of "2017-03-23 23:51:02.322062574 +0000 UTC". The JSON is formatted with syntax highlighting and line numbers.

```
{
  MoreExist: false,
  ObjCount: 6,
  CurrentMarker: 0,
  NextMarker: 0,
  Objects: [
    {
      ObjectId: "",
      Object: {
        Network: "61.1.10.0",
        CIDRLen: 24,
        Paths: [
          {
            NextHop: "41.1.10.2",
            Metric: 0,
            LocalPref: 0,
            Path: [
              "400"
            ],
            PathId: 0,
            UpdatedTime: "2017-03-23 23:50:43.268568344 +0000 UTC",
            ValidPath: true,
            BestPath: true,
            MultiPath: true,
            AdditionalPath: false,
            Origin: "Incomplete",
            PathType: ""
          }
        ],
        NextHop: "31.1.10.2",
        Metric: 0,
        LocalPref: 0,
        Paths: [
          "300",
          "200",
          "400"
        ],
        PathId: 0,
        UpdatedTime: "2017-03-23 23:51:02.322062574 +0000 UTC",
        ValidPath: true,
        BestPath: false,
        MultiPath: false,
        AdditionalPath: false,
        Origin: "Incomplete",
        PathType: ""
      }
    }
  ]
}
```

IPv4 Route State



The screenshot shows a web browser window with the address bar displaying `192.168.100.187:8080/public/v1/state/ipv4Routes`. The browser's address bar also shows the URL `192.168.100.187:8080/public/v1/state/ipv4Routes`. The browser's tabs include "Building Linux L...", "Linux/drivers/ne...", "Ashutosh Shank...", "yuba.stanford.ec...", "Welcome to Bro...", "Broadcom", "Ashutosh Shank...", and "Tomahawk". The browser's status bar shows the time "11:42 PM" and the name "Ashutosh". The browser's content area displays a JSON object representing the IPv4 route state. The JSON object is a list of two objects, each representing a route. The first object has the following properties: `MoreExist: false`, `ObjCount: 6`, `CurrentMarker: 0`, `NextMarker: 0`, `Objects: [`, `- {`, `Objectid: "",`, `Object: {`, `DestinationNw: "61.1.10.0/24",`, `Protocol: "EBGP",`, `IsNetworkReachable: true,`, `RouteCreatedTime: "2017-03-23 23:50:43.282342623 +0000 UTC",`, `RouteUpdatedTime: "",`, `NextHopList: [`, `- {`, `NextHopIp: "41.1.10.2",`, `NextHopIntRef: "br20",`, `Weight: 0`, `},`, `],`, `PolicyList: [],`, `NextBestRoute: {`, `Protocol: "",`, `NextHopList: null`, `}`, `}`, `- {`, `Objectid: "",`, `Object: {`, `DestinationNw: "41.1.10.0/24",`, `Protocol: "CONNECTED",`, `IsNetworkReachable: true,`, `RouteCreatedTime: "2017-03-23 23:50:27.814779493 +0000 UTC",`, `RouteUpdatedTime: "",`, `NextHopList: [`, `- {`, `NextHopIp: "0.0.0.0",`, `NextHopIntRef: "br20",`, `Weight: 0`, `},`, `],`, `PolicyList: [],`, `NextBestRoute: {`, `Protocol: "",`, `NextHopList: null`, `}`, `}`, `]`, `}`. The browser's status bar shows the text "Objects".

```
{
  MoreExist: false,
  ObjCount: 6,
  CurrentMarker: 0,
  NextMarker: 0,
  Objects: [
    - {
      Objectid: "",
      Object: {
        DestinationNw: "61.1.10.0/24",
        Protocol: "EBGP",
        IsNetworkReachable: true,
        RouteCreatedTime: "2017-03-23 23:50:43.282342623 +0000 UTC",
        RouteUpdatedTime: "",
        NextHopList: [
          - {
            NextHopIp: "41.1.10.2",
            NextHopIntRef: "br20",
            Weight: 0
          }
        ],
        PolicyList: [ ],
        NextBestRoute: {
          Protocol: "",
          NextHopList: null
        }
      }
    },
    - {
      Objectid: "",
      Object: {
        DestinationNw: "41.1.10.0/24",
        Protocol: "CONNECTED",
        IsNetworkReachable: true,
        RouteCreatedTime: "2017-03-23 23:50:27.814779493 +0000 UTC",
        RouteUpdatedTime: "",
        NextHopList: [
          - {
            NextHopIp: "0.0.0.0",
            NextHopIntRef: "br20",
            Weight: 0
          }
        ],
        PolicyList: [ ],
        NextBestRoute: {
          Protocol: "",
          NextHopList: null
        }
      }
    }
  ]
}
```



DEMO
Time

References

- <http://www.openswitch.net/>
- <https://github.com/open-switch>
- <https://github.com/open-switch/flx-docs>
- <https://github.com/open-switch/flx-docs/blob/master/README.md>
- <https://github.com/open-switch/opx-flx13>
- <https://github.com/open-switch/opx-flx12>
- <https://github.com/open-switch/opx-flxconfig>
- <https://github.com/open-switch/opx-flxcps>
- https://github.com/open-switch/flx-docs/blob/master/opx_flx_ebgp_demo.sh
- <https://github.com/open-switch/opx-docs>

Q & A





THANK
You 😊