# Batfish使用报告

## 一、Start with Batfish

1. **初始化网络和快照**

*bf = Session(host="localhost")*

*# Assign a friendly name to your network and snapshot*

*NETWORK\_NAME = "example\_network"*

*SNAPSHOT\_NAME = "example\_snapshot"*

*SNAPSHOT\_PATH = "networks/trans"*

*# Now create the network and initialize the snapshot*

*bf.set\_network(NETWORK\_NAME)*

*bf.init\_snapshot(SNAPSHOT\_PATH, name=SNAPSHOT\_NAME, overwrite=True)*

1. **查询网络模型**

Batfish 创建了一个厂商无关的综合设备和网络模型，可查询有关设备、接口、VRF（虚拟路由转发）、路由等信息。它提供了一系列问题来查询该模型。

*# You can also use tab-completion on the Batfish question module - bf.q. -> press TAB key,*

*# uncomment and try on the following line*

*# bf.q.*

*# In IPython and Jupyter you can use the "?" shorthand to get help on a question*

*?bf.q.nodeProperties*

*# help(bf.q.nodeProperties) # in standard Python console*

1. **获取已解析文件的状态**

*parse\_status = bf.q.fileParseStatus().answer().frame()*

*# View the parse status results*

*parse\_status*

*# An example: use a filter on the returned dataframe to see which files failed to parse completely*

*parse\_status[parse\_status['Status'] != 'PASSED'] # change '!=' to '==' to get the files which passed*

*# View details if some of the files were not parsed completely*

*bf.q.parseWarning().answer().frame()*

1. **提取网络实体的属性**

网络中的实体指的是节点、接口、路由进程和 VRF 等。Batfish 可以以不影响供应商的方式提取这些实体的配置属性。

*# Extract the properties of all nodes whose names contain 'border'*

*node\_properties = bf.q.nodeProperties(nodes="/border/").answer().frame()*

*# View what columns (properties) are present in the answer*

*node\_properties.columns*

*# To extract only a subset of properties, use the properties parameter*

*bf.q.nodeProperties(nodes="/border/", properties="Interfaces").answer().frame()*

Batfish 支持以下配置属性：

|  |  |
| --- | --- |
| ***名称*** | ***解释*** |
| *Node* | *设备节点，表示网络中的一个物理或逻辑设备* |
| *AS\_Path\_Access\_Lists* | *BGP的AS路径访问控制列表，用于匹配或过滤BGP路由的AS路径* |
| *Authentication\_Key\_Chains* | *认证密钥链，常用于路由协议（如EIGRP、OSPF）进行报文验证* |
| *Community\_Match\_Exprs* | *BGP社群匹配表达式，用于匹配路由社群属性* |
| *Community\_Set\_Exprs* | *BGP社群集合表达式，用于定义如何修改社群属性* |
| *Community\_Set\_Match\_Exprs* | *匹配特定的BGP社群集合，用于控制策略选择* |
| *Community\_Sets* | *定义的BGP社群集合，包含一组社群标签* |
| *Configuration\_Format* | *配置文件格式，如Cisco IOS、Juniper Junos等* |
| *DNS\_Servers* | *配置的域名解析服务器地址* |
| *DNS\_Source\_Interface* | *用于发送DNS请求的源接口* |
| *Default\_Cross\_Zone\_Action* | *不同安全区域之间默认的流量处理策略（如permit或deny）* |
| *Default\_Inbound\_Action* | *接口或防火墙的默认入站流量处理策略* |
| *Domain\_Name* | *设备所属的域名，常用于自动生成FQDN等功能* |
| *Hostname* | *设备主机名，用于识别网络设备* |
| *IKE\_Phase1\_Keys* | *IKE第一阶段协商使用的预共享密钥或密钥信息* |
| *IKE\_Phase1\_Policies* | *IKE第一阶段策略，定义加密、认证、DH组等参数* |
| *IKE\_Phase1\_Proposals* | *IKE第一阶段提议，是安全协商过程中提出的参数组合* |
| *IP6\_Access\_Lists* | *IPv6访问控制列表，用于控制IPv6流量* |
| *IP\_Access\_Lists* | *IPv4访问控制列表，用于控制IPv4流量* |
| *IPsec\_Peer\_Configs* | *IPsec对等体配置，定义VPN连接的远端信息* |
| *IPsec\_Phase2\_Policies* | *IPsec第二阶段策略，定义隧道加密和认证参数* |
| *IPsec\_Phase2\_Proposals* | *IPsec第二阶段提议，定义要协商使用的加密参数组合* |
| *Interfaces* | *网络接口配置，包括IP地址、状态、类型等信息* |
| *Logging\_Servers* | *配置的日志服务器地址，用于接收系统日志* |
| *Logging\_Source\_Interface* | *用于发送日志消息的源接口* |
| *NTP\_Servers* | *配置的网络时间协议（NTP）服务器* |
| *NTP\_Source\_Interface* | *用于发送NTP请求的源接口* |
| *PBR\_Policies* | *策略路由（Policy-Based Routing）策略，定义流量转发依据* |
| *Route6\_Filter\_Lists* | *用于过滤IPv6路由的列表* |
| *Route\_Filter\_Lists* | *用于过滤IPv4路由的列表* |
| *Routing\_Policies* | *路由策略，定义路由匹配与设置规则* |
| *SNMP\_Source\_Interface* | *用于发送SNMP信息的源接口* |
| *SNMP\_Trap\_Servers* | *配置的SNMP Trap服务器地址* |
| *TACACS\_Servers* | *配置的TACACS+认证服务器列表* |
| *TACACS\_Source\_Interface* | *用于发送TACACS流量的源接口* |
| *VRFs* | *虚拟路由与转发实例，实现多租户或隔离的路由表* |
| *Zones* | *安全区域定义，通常用于防火墙策略中的源/目的区域分类* |

1. **接口属性**

*# Fetch specific properties of Loopback interfaces*

*bf.q.interfaceProperties(interfaces="/loopback/", properties="Bandwidth,VRF,Primary\_Address").answer().frame()*

类似的问题可提取其他实体的属性（例如，bgpProcessConfiguration() 可提取 BGP 进程的属性）。

1. **检查配置结构的参照完整性**

网络配置定义并引用route maps, access control lists (ACLs), prefix lists等命名结构。错误配置的两个常见指标包括引用未在任何地方定义的结构（在某些平台上会导致灾难性后果）或已定义的结构未在任何地方引用。由于 Batfish 了解配置的底层语义，因此可以轻松标记此类情况。

*# List references to undefined structures*

*bf.q.undefinedReferences().answer().frame()*

1. **检查拓扑**

*# Get layer 3 edges*

*bf.q.layer3Edges(nodes="as1border1").answer().frame()*

*# Get BGP edges*

*bf.q.bgpEdges(nodes="as1border1").answer().frame()*

1. **探索路由和转发**

Batfish 可根据快照数据本身计算网络的路由表和转发表（又称 RIB 和 FIB）。通过检查这些表，可以了解网络的路由和转发行为。

检查这种行为的一种方法是使用虚拟跟踪路由。与实时网络跟踪路由不同，Batfish 显示了网络中所有可能的流量路径，并确定了导致每个跳转的路由条目。

*# Do a traceroute from host1 to 1.0.2.2*

*tr\_frame = bf.q.traceroute(startLocation="host1", headers=HeaderConstraints(dstIps="1.0.2.2")).answer().frame()*

*# Display results using customizations to handle large string values*

*show(tr\_frame)*

另一种详细了解路由行为的方法是直接检查路由表。

*# Fetch the routing table of all VRFs on all nodes in the snapshot*

*routes\_all = bf.q.routes().answer().frame()*

(对于大型网络，第一次运行需要数据平面的问题时，获取答案可能需要几分钟时间）。由于生成的数据平面由 Batfish 保存，因此后续问题的处理速度很快）。

如上文所述，routes() 问题可以生成大量结果。您可以使用问题中的参数来限制输出--若要将结果限制在核心路由器上，请使用 nodes = "/core/"；若要将结果限制在 90.90.90.0/24 前缀上，请使用 \*\*network=90.90.90.0/24"。

*# Get all routes for the network 90.90.90.0/24 on core routers*

*bf.q.routes(nodes="/core/", network="90.90.90.0/24").answer().frame()*

## BGP Analysis

## Route Computation and Analysis

## Analyzing ACLs and firewall rules

网络和安全工程师负责确保网络中的 ACL 和防火墙规则按照预期允许和拒绝流量。这项任务通常需要人工检查或将规则集加载到实验室设备上，以便测试相关示例数据包的行为。这些方法不仅耗时，而且容易出错。

Batfish 可轻松深入分析 ACL 和防火墙规则（我们通常称之为过滤器）。它可以显示过滤器将如何处理给定数据包，直至匹配数据包的那一行；它可以保证过滤器如何处理大量数据包；它还可以对过滤器进行正确性检查，确保其中的每一行都至少匹配某些数据包。

## Analyzing BGP Route Policies

BGP 的路由策略既复杂又容易出错，这就是为什么互联网中一些最严重的网络中断都与路由策略配置错误有关，最终导致路由泄漏或接受了不该接受的路由（例如，BGP 泄漏导致日本等地互联网中断、一个微小的错误如何关闭了美国部分地区的互联网、Telia 工程师错误导致大规模网络中断）。虽然网络工程师通常很清楚路由策略应该做什么或不应该做什么（例如，请参见 MANRS 指南），但确保路由策略的正确实施却是出了名的困难。

为了说明这些功能，我们将使用一个具有两个边界路由器（名为 border1 和 border2）的示例网络。每个路由器都有一个与客户网络的 BGP 会话和一个与提供商网络的 BGP 会话。我们在本笔记本中的目标是验证来自客户的境内路由策略（称为 from\_customer）和到提供商的境外路由策略（称为 to\_provider）。

from\_customer 路由策略的目的是：

* 过滤私有地址
* 仅允许已知前缀的路由（如果它们具有正确的源 AS）
* 使用适当的社区标记允许的路由，并更新本地首选项

to\_provider 路由策略的目的是：

* 宣传我们拥有的所有前缀
* 宣传所有客户路由
* 不宣传任何其他内容

AED