Routing Area Working Group
Internet-Draft
Intended status: Standards Track
Expires: April 25, 2019

J. Heitz
K. Majumdar
Cisco
October 22, 2018

Automatic discovery and configuration of the network fabric in Massive Scale Data Centers
draft-heitz-idr-msdc-fabric-autoconf-00

Abstract

A switching fabric in a massive scale data center can comprise many 10,000's of switches and 100,000's of IP hosts. To connect and configure a network of such size needs automation to avoid errors. Zero Touch Provisioning (ZTP) protocols exist. These can configure IP devices that are reachable by the ZTP agents. A method to combine BGP, DHCPv6 and SRv6 with ZTP that can be used to configure an entire network of devices is described. It is designed to scale well, because each networked device is not required to know about more than its directly connected neighborhood.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on April 25, 2019.

Table of Contents

1. Introduction

[RFC7938] defines a massive scale data center as one that contains over one hundred thousand servers. It describes the advantages of using BGP [RFC4271] as a routing protocol in a Clos switching fabric that connects these servers. A fabric design that scales to one million servers is considered enough for the forseeable future and is the design goal of this document. Of course, the design should also work for smaller fabrics. A switch fabric to connect one million servers will consist of between 35000 and 130000 switches and 1.5 million to 8 million links, depending on how redundantly the servers are connected to the fabric and the level of oversubscription in the fabric. A switch that needs to store, send and operate on hundreds of routes is clearly cheaper than one that needs to store, send and operate on millions of links.

Such a network requires significant configuration on each switch and many cables to connect. This is an onerous task without automation.

2. Requirements

To configure a fabric network for massive scale data centers.

To detect every wiring error. For example, a spine switch that has a different number of links into one pod than into another pod in a Clos fabric.

One or multiple controllers exist to control a network. Multiple controllers are used for redundancy and to improve operation in partitioned networks.

Any devices with equivalent functionality should be interchangeable without requiring configuration changes. That means if a device breaks, it can be replaced by any other device of equivalent functionality without any changes to its configuration. Even if a replacement device already has configuration, it should still work in its new position.

A device may have configuration, but such configuration MUST NOT depend on the location of the device in the network. Therefore, no IP addresses should be pre-configured on any devices. No fabric tier should be needed.

For scalability, every device must not need to know how to reach every other device. Only a controller should be expected to know the entire topology.

If two such auto-discovering/auto-configuring networks are connected together, the function of discovery/configuration in one network must not disturb this function in the other network.

A device must accept configuration only from a well-defined set of controllers.

Separate cabling for a management network must not be required.

The network should function even if the controllers are disconnected. Link failures and restoration should be dealt with. Device failure should be dealt with. Device restoration should be dealt with as long as it does not require new configuration. A controller should only be needed to discover and configure new devices to the network.

The protocol does not need to be fast.

A controller must be able to reach any device if there is any way at all to reach it, even if that is multiple hops between spine switches or any other path that may be disallowed in a normal Clos network.

At the same time, normal traffic must remain restricted to allowable paths.

The routing protocol for normal traffic must be fast and efficient.

The network must scale to 1 million connected servers and 8 million links in the fabric.

3. Solution Overview

DHCPv6 [RFC3315] and ZTP are used to discover and configure devices reachable by the controller. As the controller configures devices, it configures them to be DHCP relay agents. This makes more devices reachable by the new DHCP relay agents, allowing the new devices to be configured. As this configuration process proceeds further away from the controller, it configures BGP to ensure reachabillity to all devices even if links were to fail. Reachability needs to be device to controller and controller to device. Every device does not need to be able to reach every other device during the discovery/ configuration process. Devices close to the controller will be used to forward packets to many more distant devices. These close devices should not store routes to reach all those more distant devices. A possible idea to reduce the routing table on close devices is to aggregate addresses of more distant devices. This is difficult and unreliable, because before discovery completes, the number of devices behind any given device is unknown. Also, if links fail, suddenly, a large number of devices could appear behind a different device, making the previous addressing structure non-aggregatable with the new topology. The chosen method to route traffic from controller to device is segment routing. The controller knows the topology. With that knowledge, it can build a segment list to reach any device.

In certain environments, it is required for devices to authenticate the network and for the network to authenticate devices. DHCPv6 provides a method to authenticate in both directions using shared keys. TCP-AO [RFC5925] can be used to authenticate BGP sessions. SZTP [I-D.ietf-netconf-zerotouch] provides for authentication during the ZTP process.

4. Solution Details

Each device needs a unique identifier. This may be printed on the device. For easy servicability, a device must have a single identifier, visible on the outside of the device and by the controller. This will be the DUID in the DHCPv6 Client Identifier Option.

In order to discover the topology, a controller needs to know every link in the topology. This means the device ID and interface ID or interface address at each end of every link. DHCPv6 can be used to obtain that information. For each link, one end of the link is the device that requests an address. The other end of the link is either the controller itself or a DHCP relay agent. The DHCP relay agent relays all client requests back to the controller.

Configuration proceeds in waves. Each controller may take part in configuring the network. The waves of configuration propagate away from each controller. In the first wave, a controller allocates a routable ipv6 address to each device directly connected to the controller. These devices comprise the first wave. The controller will then configure each of these devices using a ZTP protocol, such as [I-D.ietf-netconf-zerotouch]. The configuration for each device will include the following items:

- A routable Ipv6 address for each of its interfaces that have not already acquired one by DHCP.

- A routable Ipv6 address for the loopback interface.

- Configuration to act as a DHCPv6 relay agent for the next wave of devices.

- Configuration for a BGP session to each of its connected neighbors. That BGP session will initially be down, but will establish once the neighbors are connected and configured.

- Configuration for a BGP session to the controller.

The controller will allocate a different IP address for each interface for each device in the network. When the controller receives DHCP requests from DHCP relay agents, it will recognize the DHCP relay agent end of the link from the link-address field in the relay-forward message. The controller will note the DUID in the DHCP request to keep track of the device making the request. Because it already knows the DUID of the DHCP relay agent from its IP address, it can tie the two devices together by their DUID.

The controller must keep track of the DUID in every DHCP request, so that it can recognize different interfaces on the same device. This is needed to detect looped cables and to prevent the controller attempting to use ZTP to configure a single device through multiple links at the same time.

Two devices A and B may be connected by a link and be configured at the same time, each through a different link. At this time, the

controller does not yet know about the link A-B. In this case, neither A nor B will send a DHCP request across the link A-B. The interfaces on each end will not come up either, because the IP interface addresses will not have a common prefix. This case can be detected, because both A and B will send periodic router-advertisement messages on the link, announcing their interface IP addresses. The device with the lower address MUST send a DHCPv6 request to the other device to get a new address.

A device SHOULD use the DHCPv6 User Class Option to identify the network it is attempting to reach. This is to prevent the controller from configuring devices attached to the network that are not part of the network to be configured. A string should be used that is not likely to match that of any other network that this network is connecting to. However, even if it matches by some small chance, the DHCPv6 authentication key will likely not match or the subsequent ZTP will fail. Inadvertently getting an IP address is not a terrible thing.

The controller should allocate a different BGP AS number for each device. There are plenty of private 4-octet ASNs available.

The controller will advertise its own loopback address to all the directly connected BGP neighbors with a community to identify it as a controller address. This IP address will be advertised by all devices to their directly connected BGP neighbors. The devices will use this BGP route to route back to the controller.

Each device will announce its interface addresses to the BGP connections of its directly connected neighbors tagged with a community. These routes will be re-announced only to the BGP session to the controller and not to directly connected neighbors. The BGP connections can be made to fail upon interface down or BFD down. BFD should only operate on the BGP sessions to directly connected neighbors, not on the session to the controller.

The devices will be segment-routing V6 (SRv6) [I-D.ietf-6man-segment-routing-header] capable. When a device receives an Ipv6 packet, it will first inspect the SRv6 extension header and be able to forward the packet to the next segment. If there is no SRv6 extension header or no more segments, then the packet should be for itself or for a directly connected neighbor or for a controller. If none of those match, then it must drop the packet.

The controller, knowing the topology, will be able to send a packet to any device in the network by building the appropriate SRv6 SID

list. Thus each device in the network does not need to store a route for every other device.

Once the controller has learnt the whole network topology, or at least a large recognizable part of it, it can complete the configuration of the network. This depends on the network. The controller will be programmed with a description of the expected network and applicable constraints. As discovery proceeds, the controller will try to match the discovered topology with the programmed description. An example of a data center description is: "A number of pods. Each pod consists of 384 TORs and 32 spines. Each TOR has 32 south facing ports and 32 north facing ports. Each spine has 384 south facing ports and 192 north facing ports. Super-spines connect the pods. Some of the pods are DCI pods. The devices need aggregatable addresses and BGP sessions." The controller should be able to recognize all the switches, the servers and the DCI routers and match the discovered topology to the description. It should then create configurations for all the devices and report inconsistencies. How the controller does this is out of scope of this document.

When a new device joins the network, the controller will detect it, because it will receive a DHCP request from it, relayed by its neighboring DHCP relay agent.

5. Security Considerations

TBD

6. IANA Considerations

TBD

7. Acknowldgements

8. References

8.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC3315] Droms, R., Ed., Bound, J., Volz, B., Lemon, T., Perkins, C., and M. Carney, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 3315, DOI 10.17487/RFC3315, July 2003, <https://www.rfc-editor.org/info/rfc3315>.

[RFC4271] Rekhter, Y., Ed., Li, T., Ed., and S. Hares, Ed., "A Border Gateway Protocol 4 (BGP-4)", RFC 4271, DOI 10.17487/RFC4271, January 2006, <https://www.rfc-editor.org/info/rfc4271>.

[RFC5925] Touch, J., Mankin, A., and R. Bonica, "The TCP Authentication Option", RFC 5925, DOI 10.17487/RFC5925, June 2010, <https://www.rfc-editor.org/info/rfc5925>.

8.2. Informative References

[I-D.ietf-6man-segment-routing-header] Filsfils, C., Previdi, S., Leddy, J., Matsushima, S., and d. daniel.voyer@bell.ca, "IPv6 Segment Routing Header (SRH)", draft-ietf-6man-segment-routing-header-14 (work in progress), June 2018.

[I-D.ietf-netconf-zerotouch] Watsen, K., Abrahamsson, M., and I. Farrer, "Zero Touch Provisioning for Networking Devices", draft-ietf-netconf-zerotouch-25 (work in progress), September 2018.

[RFC7938] Lapukhov, P., Premji, A., and J. Mitchell, Ed., "Use of BGP for Routing in Large-Scale Data Centers", RFC 7938, DOI 10.17487/RFC7938, August 2016, <https://www.rfc-editor.org/info/rfc7938>.

Authors' Addresses

Jakob Heitz
Cisco
170 West Tasman Drive
San Jose, CA, CA 95134
USA

Email: jheitz@cisco.com

Kausik Majumdar
Cisco
170 West Tasman Drive
San Jose, CA, CA 95134
USA

Email: kmajumda@cisco.com