Scaling networks in DC networks using FOSS

Hendrik Sokolowski
DevOps, hosting.de

Summary

- Challenges in DC networks
- The TCP/IP Stack
- Buzzwords and marketing phrases
- Downsides of large networks
- Ingredients
- Let's put it all together

Challenges in DC networks

- Virtual Private Cloud
- DC Interconnect
- On-demand creation of new networks
- Multi-Homing
- Scaling for east-west traffic
- “The Customer”

But first...

The (slightly reduced) TCP/IP Stack

4 Application Layer	HTTP	FTP	MQTT	SMTP	IMAP	POP3
	TFTP	Telnet	DNS	mDNS	SSH	SNTP
3 Transport Layer	TCP		UDP		RAW	
2 Internet Layer	IPv4			IPv6		
1 Link Layer	ARP	NDP	SLAAC	ICMP	IGMP	DHCP
	Ethernet			WiFi		

The (slightly reduced) TCP/IP Stack

Ethernet (link layer):

- Format: MAC-address: aa:bb:cc:dd:ee:ff (6 byte / 48 bit, hex)
  - 2^48 -> 281,474,976,710,656
- addresses a device within a network domain
- 802.1Q / VLAN: segmentation into 4096 domains (12 bit)
- 802.1ad / QinQ: adds second 12 bit header
  - up to 4096 x 4096 = 16,777,216 domains
  - Scenario: "The Customer" wants his own vlan-capable network
  - Scenario: I am Google and have Apple as customer
- 16,777,216 domains just not enough

The (slightly reduced) TCP/IP Stack

IP (internet layer):

- Format:
  - IPv4: 192.168.0.1 (4 byte / 32 bit, decimal)
    - 2^32 -> 4,294,967,296
  - IPv6: 2001:0db8::0370:7344 (16 byte / 128 bit, hex)
    - 2^128 -> 340,282,366,920,938,463,463,374,607,431,768,211,456
- addresses a host throughout the internet
-

Buzzwords and marketing phrases

VPC: Virtual private Cloud

- standard cloud use-case
- customer gets private network
  - usually a single non-vlan capable domain
  - often multicast traffic is limited
  - not every ip-address can be chosen
- customer gets highly-available Proxy / Firewall / VPN Gateway
  - Allows for encrypted connections
  - Allows exposing single private services to the outside
    - eg. HTTP(S) Port 80 / 443

DC interconnect

- VPC spanned over multiple geological regions
- L3 traffic is send to ip-addresses but devices have mac-addresses IPv4: ARP IPv6: NDP ARP uses L2 broadcast NDP uses L3 multicast Multiple gateways to optimize traffic-flow from different locations

High-Availability / Multi-Homing

- link layer
  - a single device addressed by it’s mac-address can be reached over multiple physical links
  - eg. server to switch in datacenter
- internet layer
  - an ip-address is reachable over multiple routes
- difficulties
  - performance
    - 40 / 100 GBit/s links widespread
    - up to 400 GBit/s links affordable
  - SLA
    - more nines more better

Downsides of large networks

- data-plane learning: Hosts are resolved using ARP / NDP
  - packets are duplicated to any host within ethernet domain
  - Example:
    - host wants to resolve default gateway (192.168.0.5 -> 192.168.0.1)
    - The host does an arp request, asking for the mac-address of a known ip-address
    - ARP is broadcast traffic /
    - Every Arp frame is replicated to every reachable target
    - Arp information are aging out within 5 to 15 minutes

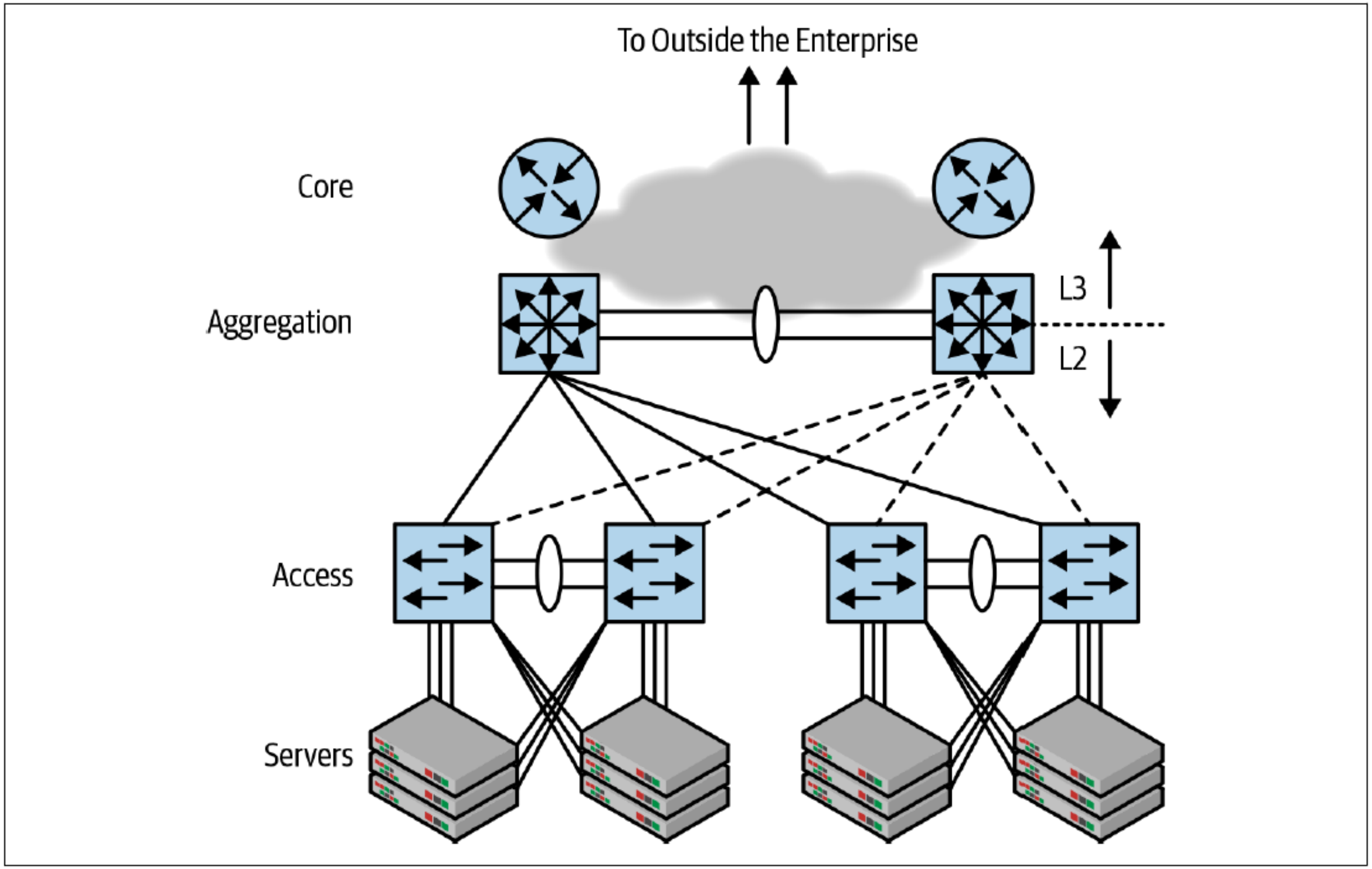
Downsides of large networks

- Multihoming complicated
  - For redundancy reasons you want your servers to be connected to at least 2 different switches
  - In Link Layer loops are not allowed
  - Often done using LACP / 802.3ad
  - On the switch side you need proprietary technologies like port-channel or virtual-chassis
  - Always requires the switches talking LACP to be interconnected with each other
  - Error-prone, as the two devices are no longer fully independent
  - Often only two links per LAG are supported

Downsides of large networks

- Scaling even more advanced
  - Every link has a limited performance (current max: 400 Gbit/s, widely found: 40 or 100 Gbit/s)
  - What if your storage cluster needs moARRRrr bandwidth?
  - You cannot just add more links, remember?

Downsides of large networks



Source: https://silvanogai.github.io/assets/images/access-aggregation-core.png

Ingredients

- Recent linux kernel > 5.0
- BGP Routing Daemon
  - FRR
  - goBGP
  - ...
- Network links with VxLAN support
- ToR Switch / Router with BGP / EVPN support

VxLAN?

- Encapsulation technique
    - ethernet frames + VxLAN header are packed into UDP packets
    - packet is send to remote target, decapsulated and injected into target L2 domain
    - up to 2^32 distinct L2-domains are supported
    - each L2-domain is tagged with a VNI (virtual network identifiert) that is encoded in the VxLAN packet
    - ethernet frames can be equipped with 802.1Q / 802.1ad header
        - possible, lacks hardware support
    - hosts en- and decapsulating VxLAN traffic are called VTEP
        - VTEP: virtual tunnel endpoint

BGP?

- TCP Protocol to exchange routes between autonomous systems
- Runs the internet for decades now
- Supports multiple address families (type of routes)
  - IPv4
  - IPv6
  - EVPN
  - ...

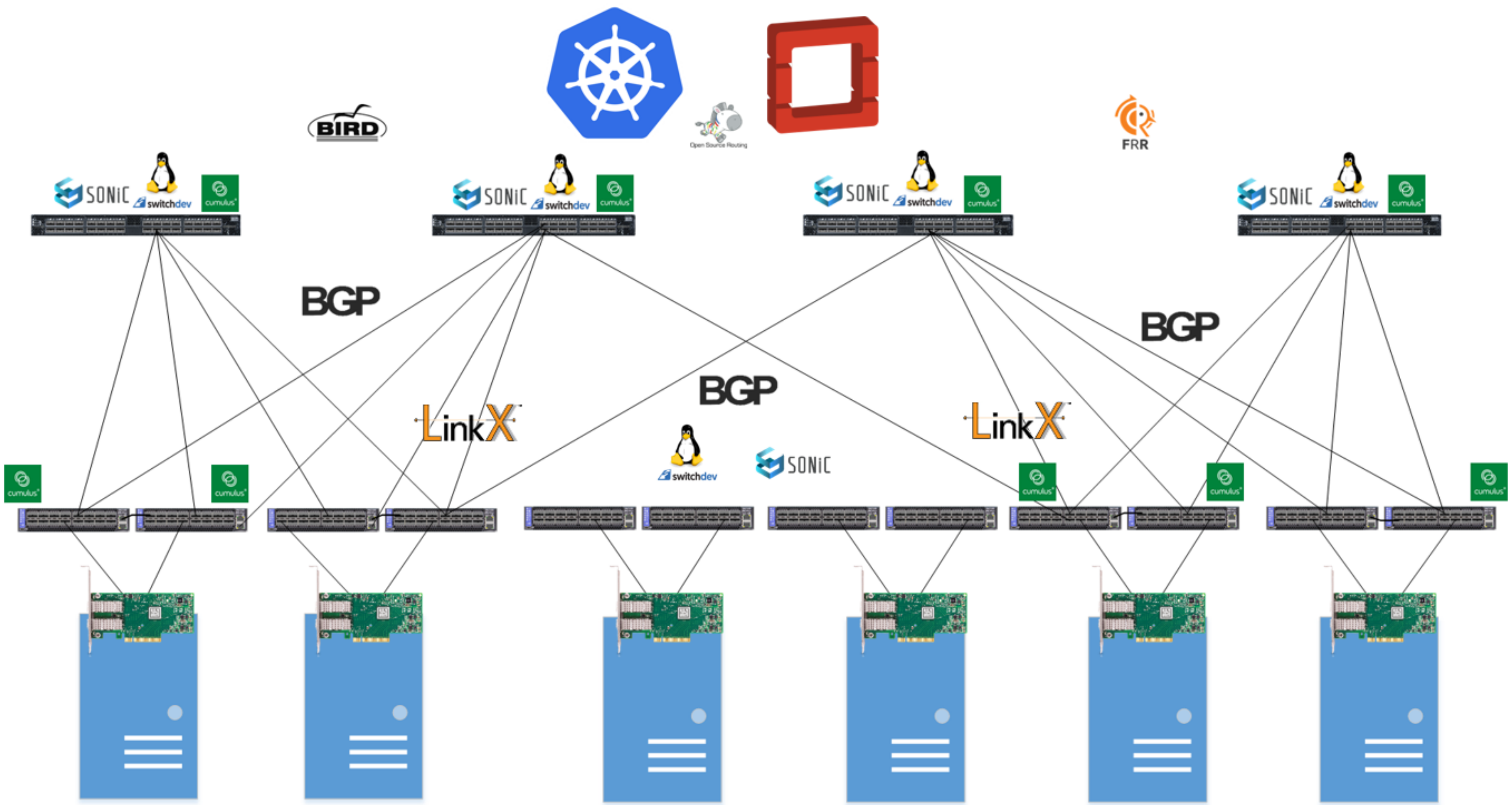
EVPN?

- a BGP "address-family"
- special kinds of routes are exchanged
  - MAC / MAC-IP route (type 2)
  - VTEP discovery (type 3)
  - Ethernet Segment route (type 4)
  - ...
- standardized (RFC 7432 / 8365)
  - adopted by every large network manufacturer
  - available in several open source bgp implementations

Let's put it all together

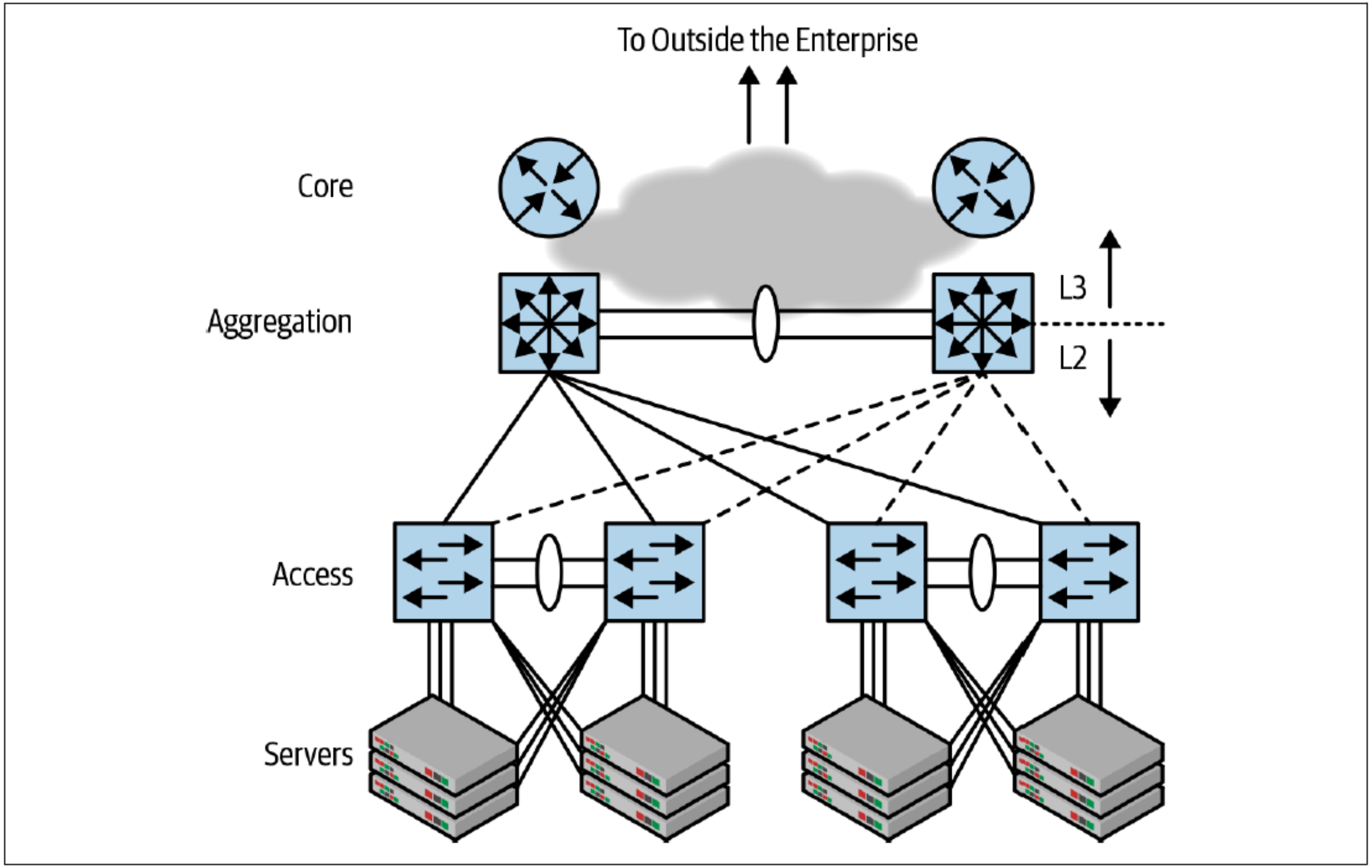
- we have BGP / EVPN to exchange information about attached hosts
  - ARP / NDP traffic can be filtered out as it is not needed anymore
- we have VxLAN to make ethernet traffic routable
  - ethernet frames are encapsulated with a VxLAN header into a UDP packet
- we route traffic as early as possible
  - we let every VTEP act as gateway
  - we move the problem of high-availability to the internet layer instead of link layer

How does it look like?



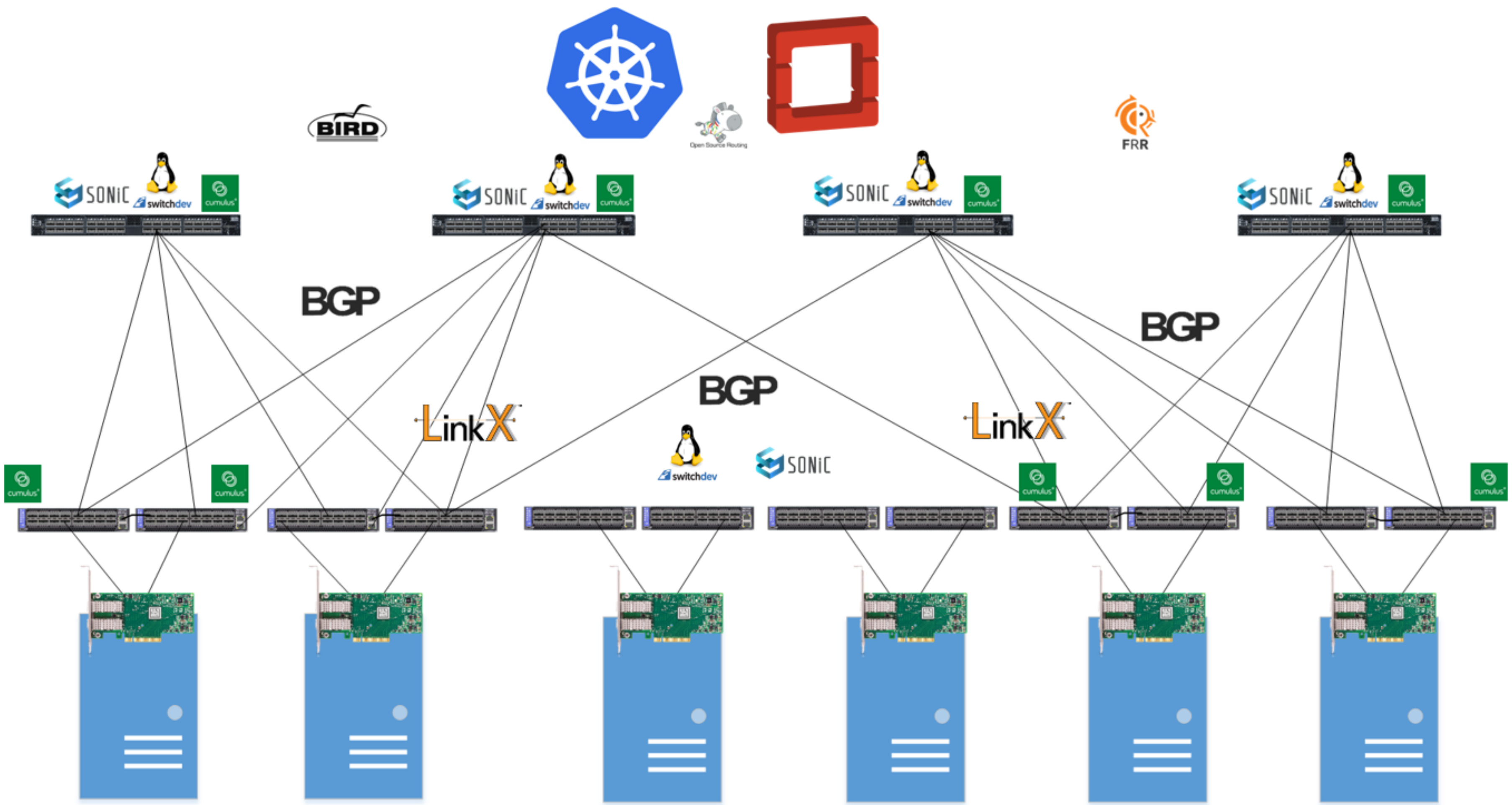
Source: https://blog.mellanox.com/wp-content/uploads/diagram-1-1.png

Compared to a classic network...



Source: https://silvanogai.github.io/assets/images/access-aggregation-core.png

... this is awesome



Source: https://blog.mellanox.com/wp-content/uploads/diagram-1-1.png

How does that make my network scalable?

- instead of relying on vlans we use a routed underlay network done via BGP v4/v6
  - every switch acts as a router now
  - hypervisors / servers will be equipped with an EVPN capable routing daemon
- the servers listen to their local bridges and learn what hosts are attached
  - when the server learns about a new host it publishes an EVPN type-2 route containing MAC / IP
    - MAC
    - IP
    - VNI
  - ARP / NDP can effectively be filtered
- EVPN routes are published through BGP

Ah and what about Multi-homing?

- use case: Hypervisor
  - routing is done on the hypervisor
  - links to Rack Switch are used for independent BGP Sessions
  - every link can fail independently without the network going down
- use case: Classic server
  - routing is done on the Rack Switch
  - the links to the server are setup using LACP
  - supported by almost every OS
  - EVPN Type-1 / Type-4 routes are used to replace proprietary LACP implementations
  - no switch interconnection necessary anymore

Done

Done?

Caveats

- BUM traffic
- Complexity

Thank you

Hendrik Sokolowski
DevOps, hosting.de
hendrik.sokolowski@gssws.de (mailto:hendrik.sokolowski@gssws.de)
https://github.com/hensoko (https://github.com/hensoko)
@hensoko (http://twitter.com/hensoko)