



The bridge to possible

An Introduction to SONiC on Cisco 8000

Suhaib Ahmad
Technical Marketing Engineer
DEVNET-1086

CISCO *Live!*

#CiscoLive

Cisco Webex App

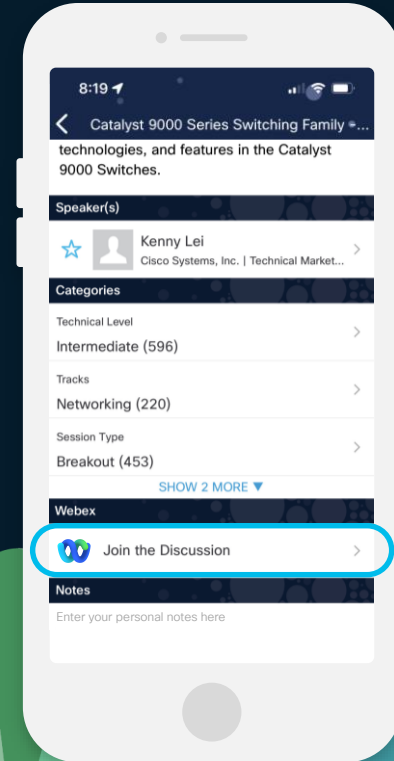
Questions?

Use Cisco Webex App to chat with the speaker after the session

How

- 1 Find this session in the Cisco Live Mobile App
- 2 Click “Join the Discussion”
- 3 Install the Webex App or go directly to the Webex space
- 4 Enter messages/questions in the Webex space

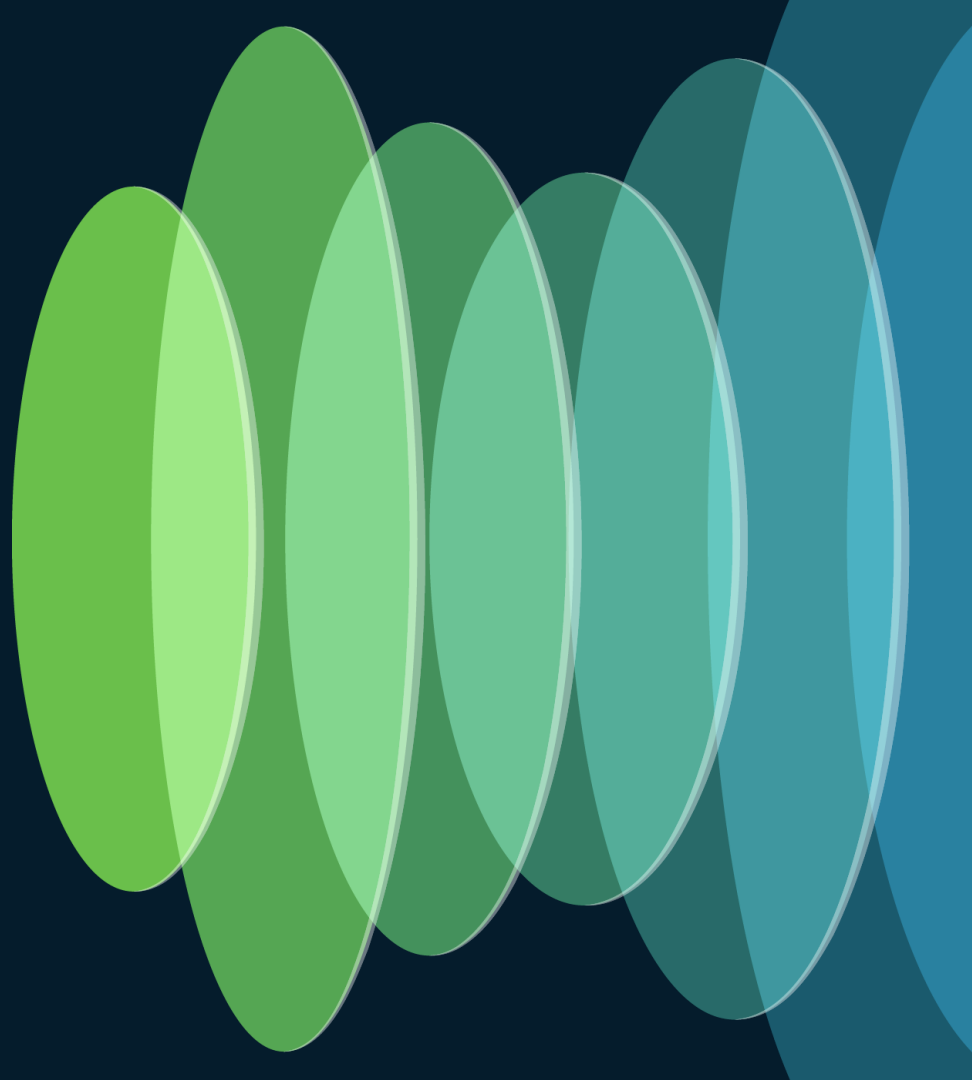
Webex spaces will be moderated by the speaker until June 7, 2024.



Agenda

- Introduction to SONiC, Design Goals, and Use Cases
- Architecture and Components Deep-dive
- SONiC Operational Considerations
- SONiC and Cisco 8000 routers
- Conclusion

Introduction to SONiC, Design Goals, and Use Cases



The SONiC Network Operating System

- SONiC – Software for Open Networking in the Cloud is an open-source network operating system based on Debian Linux.
- Initially created by Microsoft for use in their Azure datacenters. Open-sourced by Microsoft in 2016.
- It became part of the Linux Foundation in 2022 which focuses on the software components of SONiC and partners with Open Compute Platform (OCP) on aligning hardware and specifications like the Switch Abstraction Interface.
- <https://github.com/sonic-net/SONiC/wiki>
- <https://sonicfoundation.dev/>



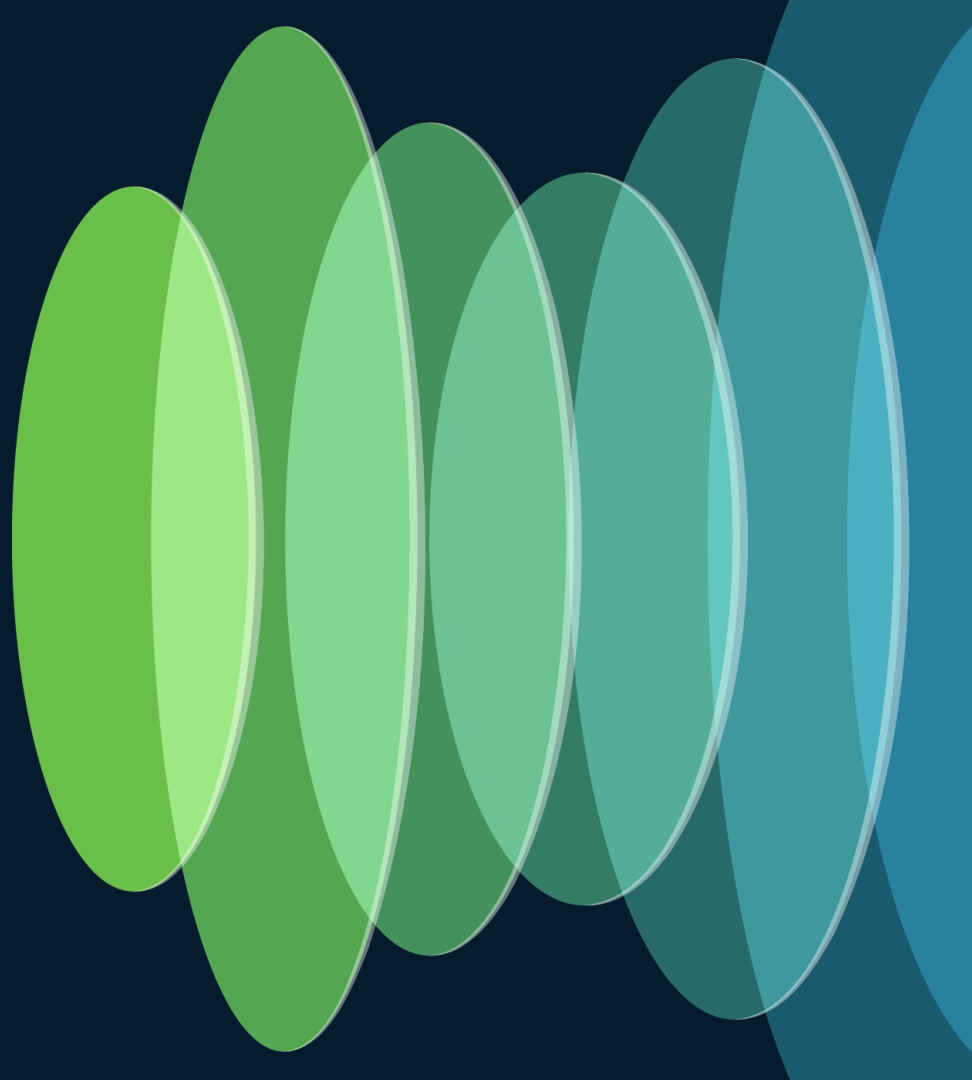
SONiC Design Goals

- **Open-Source Software**: Open-source approach to network operating systems.
- **Hardware Agnostic**: Creating a Network Operating System can be integrated with different ASICs and NPUs.
- **Modularity**: Network Operating System functions separated as docker containers.
- **Operational Ease**: Simplify network operations with support for APIs and configuration management tools.

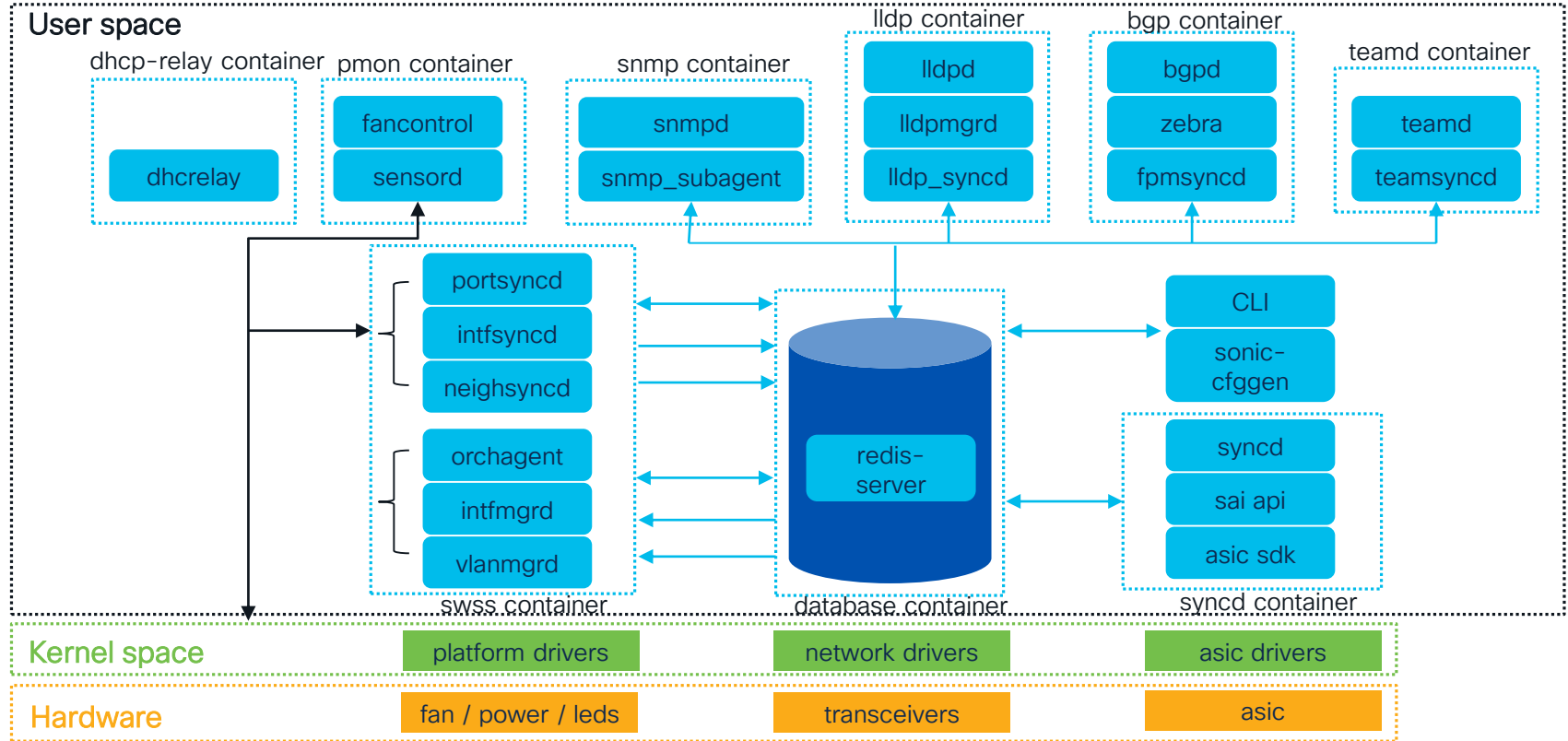
Use Cases

- **Hyperscaler datacenters**
 - IP/BGP Datacenter fabric
 - Overlay fabric with static VXLAN
 - ToR, Leaf, and Fixed-Chassis (or Multi Instance) Spine roles.
- **AI/ML Frontend and backend networks**
 - Backend clusters
 - RDMA over Converged Ethernet v2 (RoCEv2)
 - QoS with Priority Flow Control (PFC) and Explicit Congestion Notification (ECN).
 - BGP Routing, ECMP load balancing
 - Frontend clusters
 - IP/BGP Routing and VXLAN

Architecture and Components Deep-dive

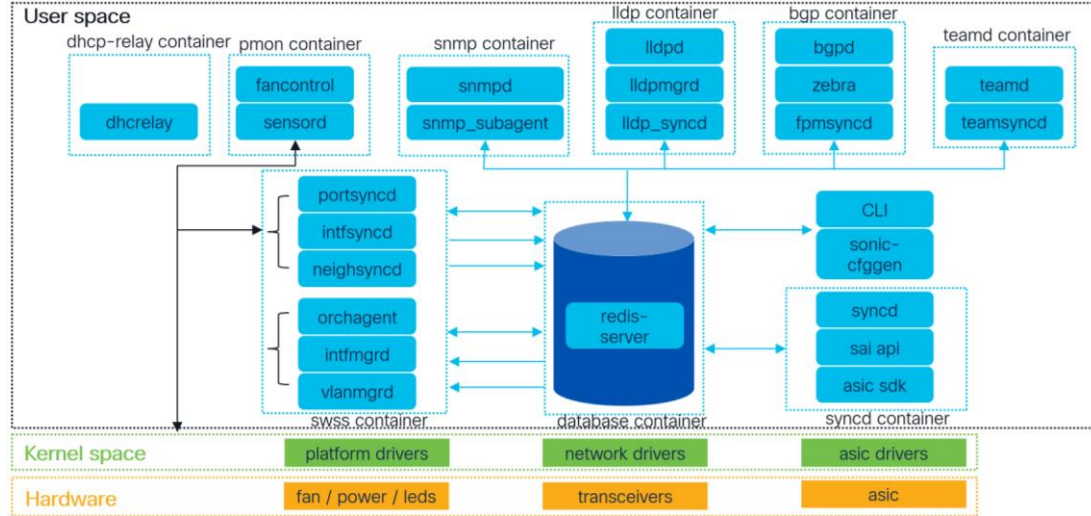


SONiC Architecture



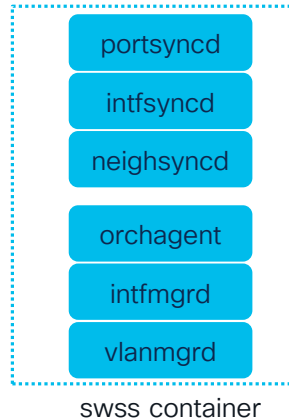
SONiC Architecture

- Networking Operating System (NOS) running on Debian Linux
- NOS components implemented as separate docker containers.
- In-memory Redis databases used by all containers for application state, configuration state, operational state, ASIC state and counters.



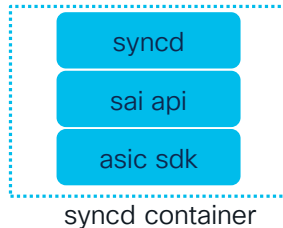
SONiC Components: Switch State Service

- Switch State Service (SwSS) container: This container exists to provide communication services between all SONiC modules.
- *Orchagent*: Reads relevant state from daemons in other containers that need to program the ASIC (fpmSyncd, lldp_syncd, teamsyncd), process it and writes it to the ASIC_DB database in the Redis container.



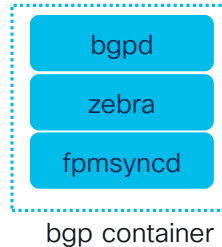
SONiC Components: syncd

- syncd container: Responsible for synchronizing the state present at the SONiC NOS layer and the actual forwarding ASIC in the platform.
- *SAI API*: Switch Abstraction Interface API defines a vendor-agnostic set of endpoints for managing the forwarding ASIC in the platform.
- *syncd*: Subscribes to the ASIC_DB database (which is updated by the SWSS container) and programs the ASIC using the ASIC SDK.



SONiC Components: bgp

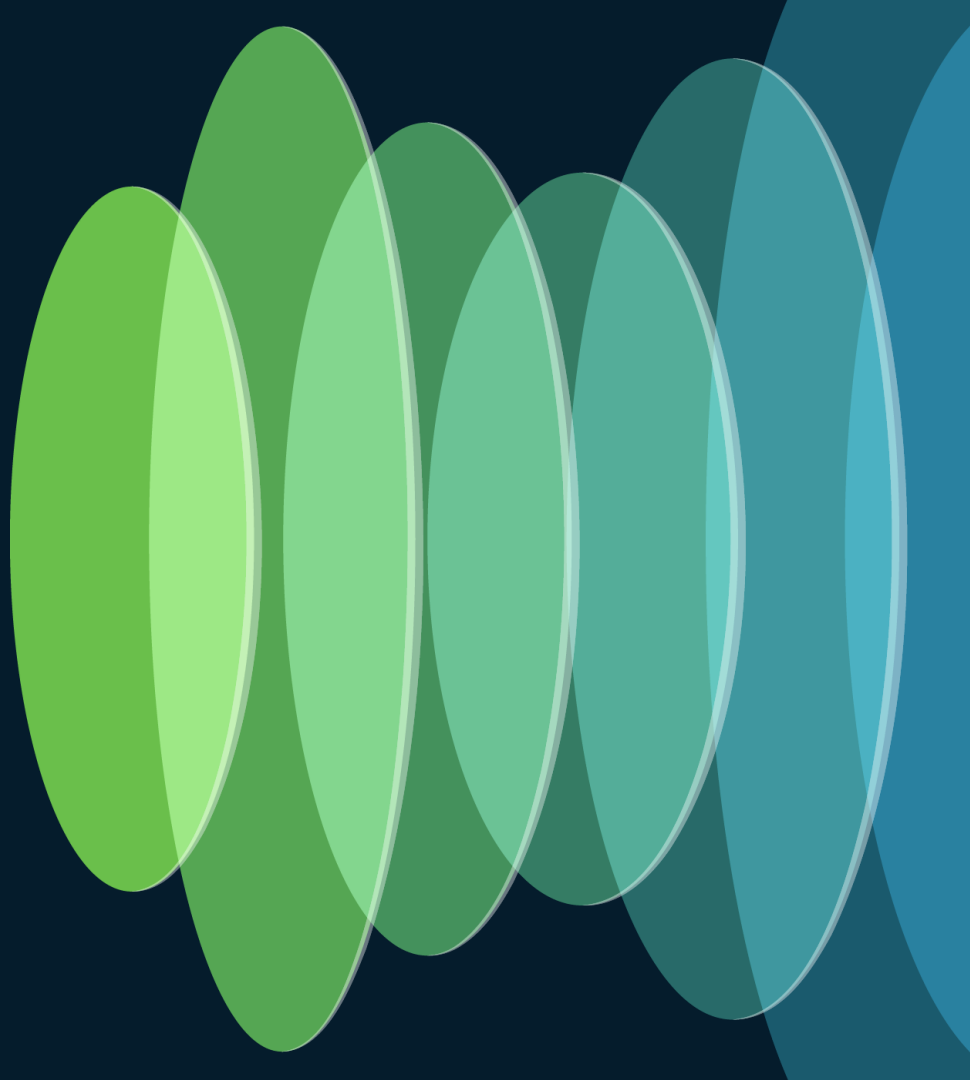
- bgp container: The SONiC routing stack is implemented in this container. It is based on the open-source Free Range Routing (FRR) protocol suite.
 - *bgpd*: Collects routing updates from BGP neighbors.
 - *zebra*: Provides kernel routing table updates, interface lookups, and redistribution of routes between different routing protocols.
 - *fpmsyncd*: Writes FIB state calculated by zebra to the Redis container.



SONiC Components: Other containers

- Platform Monitor (pmon) container: Collects sensor readings from platform sensors. Raises alarms (if any) and controls fan speeds on the basis of sensor readings.
- SNMP (snmp) container: Implements an SNMP server for SONiC.
- Teamd (teamd) container: Provides link aggregation functionality in SONiC. Based on teamd, an open-source LAG protocol.
- LLDP (lldp) container: Implements Link Layer Discovery Protocol for SONiC.

SONiC Operational Considerations



SONiC Install

- Open Network Install Environment (ONIE) is SONiC's supported install environment.
- It combines a boot loader with a Linux kernel and Busybox and Provides boot options such as iPXE.
- The busybox environment allows user-configurable install actions as well as password recovery
- SONiC CLI also supports install upgrade and rollback actions.

SONiC Install: ONIE

GNU GRUB version 2.02~beta2+e4a1fe391

```
+-----+
|*ONIE: Install OS
| ONIE: Rescue
| ONIE: Uninstall OS
| ONIE: Update ONIE
| ONIE: Embed ONIE
+-----+
```



Use the ^ and v keys to select which entry is highlighted.
Press enter to boot the selected OS, `e' to edit the commands
before booting or `c' for a command-line.

SONiC Install: sonic-installer

```
root@sonic:/home/admin# sonic-installer list
```

```
Current: SONiC-OS-202305.1.2-6514cfa26
```

```
Next: SONiC-OS-202205.3.3-7c68be04e
```

```
Available:
```

```
SONiC-OS-202205.3.3-7c68be04e
```

```
SONiC-OS-202305.1.2-6514cfa26
```



SONiC Install: sonic-installer

```
root@sonic:/home/admin# sonic-installer install http://1.63.51.21/sonic/sonic-cisco-8000.202205.3.3/sonic-cisco-8000.202205.3.3.bin
New image will be installed, continue? [y/N]: y
Downloading image...
...99%, 1567 MB, 11301 KB/s, 0 seconds left...
efi not supported - exiting without verification

Installing image SONiC-OS-202205.3.3-7c68be04e and setting it as default...
Command: bash /tmp/sonic_image
Verifying image checksum ... OK.
Preparing image archive ... OK.
Installing SONiC in SONiC
ONIE Installer: platform: x86_64-cisco-8000-r0
onie_platform: x86_64-8201_32fh_o-r0
Installing SONiC to /host/image-202205.3.3-7c68be04e
Archive: fs.zip
  creating: /host/image-202205.3.3-7c68be04e/boot/
  inflating: /host/image-202205.3.3-7c68be04e/boot/vmlinuz-5.10.0-18-2-amd64
  inflating: /host/image-202205.3.3-7c68be04e/boot/initrd.img-5.10.0-18-2-amd64
  inflating: /host/image-202205.3.3-7c68be04e/boot/System.map-5.10.0-18-2-amd64
  inflating: /host/image-202205.3.3-7c68be04e/boot/config-5.10.0-18-2-amd64
  extracting: /host/image-202205.3.3-7c68be04e/fs.squashfs
Switch CPU vendor is: GenuineIntel
Switch CPU cstates are: disabled
EXTRA_CMDLINE_LINUX=
Installed SONiC base image SONiC-OS successfully
```

SONiC Install: sonic-installer

```
root@sonic:/home/admin# sonic-installer set-next-boot SONiC-OS-202305.1.2-6514cfa26
```

```
Command: grub-reboot --boot-directory=/host 1
```

```
root@sonic:/home/admin#
```



Sets the image to be loaded on the next router reload

SONiC Configuration

- The device configuration is stored as a json file and can be accessed from `/etc/sonic/config_db.json`
- Contents of `/etc/sonic/config_db.json` are loaded into the Redis database on startup or by issuing a `config load` or `config reload` command
- SONiC also has a simple Python Click based CLI interface.
- “show” commands can be executed by any user whereas configuration commands require sudo privileges
- Configuration actions can be made by either:
 - Modifying the `config_db.json` file using the `sonic-cfggen` utility
 - CLI commands in the Linux shell
 - vty shell in FRR
- SONiC configuration management can be simplified by using open-source tooling such as Ansible and Terraform.
- gRPC/gNMI also supported, however, gNMI implementation is recent and does not cover all endpoints

SONiC show CLI examples

```
admin@sonic:~$ show platform summary
```

```
Platform: x86_64-8201_32fh_o-r0
```

```
HwSKU: 32x400Gb
```

```
ASIC: cisco-8000
```

```
ASIC Count: 1
```

```
Serial Number: FLM27200ADU
```

```
Model Number: 8201-32FH-O
```

```
Hardware Revision: 1.0
```



SONiC show CLI examples

```
admin@sonic:~$ show version
```

```
SONiC Software Version: SONiC.202305.1.2-6514cfa26
SONiC OS Version: 11
Distribution: Debian 11.9
Kernel: 5.10.0-23-2-amd64
Build commit: 6514cfa26
Build date: Wed Mar 13 09:33:29 UTC 2024
Built by: sonicci@sonic-ci-8-lnx
```

```
Platform: x86_64-8201_32fh_o-r0
HwSKU: 32x400Gb
ASIC: cisco-8000
ASIC Count: 1
Serial Number: FLM27200ADU
Model Number: 8201-32FH-O
Hardware Revision: 1.0
Uptime: 22:25:57 up 31 days, 4:27, 1 user, load average: 1.32, 1.14, 1.04
Date: Fri 31 May 2024 22:25:57
```

```
<...>
```

```
<...>
```

Docker images:	TAG	IMAGE ID	SIZE
REPOSITORY			
docker-macsec	latest	f7a9a7d4ff58	340MB
docker-dhcp-relay	latest	4bc1d2cc1192	328MB
docker-ixeserver-cisco	202305.1.2-6514cfa26	6d594511b9ea	352MB
docker-ipxeserver-cisco	latest	6d594511b9ea	352MB
docker-gbsyncd-cisco	202305.1.2-6514cfa26	29a320c79090	386MB
docker-gbsyncd-cisco	latest	29a320c79090	386MB
docker-sonic-telemetry	202305.1.2-6514cfa26	0974c2d9654e	407MB
docker-sonic-telemetry	latest	0974c2d9654e	407MB
docker-teamd	202305.1.2-6514cfa26	e8e364924112	338MB
docker-teamd	latest	e8e364924112	338MB
docker-snmp	202305.1.2-6514cfa26	a498e1e14b1c	359MB
docker-snmp	latest	a498e1e14b1c	359MB
docker-sflow	202305.1.2-6514cfa26	bfd6484d7a6d	339MB
docker-sflow	latest	bfd6484d7a6d	339MB
docker-router-advertiser	202305.1.2-6514cfa26	a4ab1e96403f	320MB
docker-router-advertiser	latest	a4ab1e96403f	320MB
docker-platform-monitor	202305.1.2-6514cfa26	087a1cb73d85	443MB
docker-platform-monitor	latest	087a1cb73d85	443MB
docker-orchagent	202305.1.2-6514cfa26	40b5f419bfd3	350MB
docker-orchagent	latest	40b5f419bfd3	350MB
docker-nat	202305.1.2-6514cfa26	3fb6329dc917	341MB
docker-nat	latest	3fb6329dc917	341MB
docker-mux	202305.1.2-6514cfa26	fcaa4514b7af	369MB
docker-mux	latest	fcaa4514b7af	369MB
docker-lldp	202305.1.2-6514cfa26	da9e0a1f94c9	363MB
docker-lldp	latest	da9e0a1f94c9	363MB
docker-fpm-frr	202305.1.2-6514cfa26	bf63f51bab81	369MB
docker-fpm-frr	latest	bf63f51bab81	369MB
docker-eventd	202305.1.2-6514cfa26	6fc2fd9a7bc4	320MB
docker-eventd	latest	6fc2fd9a7bc4	320MB
docker-database	202305.1.2-6514cfa26	52c1c0481fdf	320MB
docker-database	latest	52c1c0481fdf	320MB
docker-synd-cisco	202305.1.2-6514cfa26	fa836211e367	812MB
docker-synd-cisco	latest	fa836211e367	812MB
docker-sonic-mgmt-framework	202305.1.2-6514cfa26	45c7465d9c8b	415MB
docker-sonic-mgmt-framework	latest	45c7465d9c8b	415MB

SONiC Containers

```
admin@sonic:/home/admin# docker container ls
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
945bcbff6d13	docker-sonic-telemetry:latest	"/usr/local/bin/supe..."	4 weeks ago	Up 4 weeks		telemetry
5cce97df9b4	docker-snmp:latest	"/usr/local/bin/supe..."	4 weeks ago	Up 4 weeks		snmp
8a9a6b19e1cc	docker-platform-monitor:latest	"/usr/bin/docker_ini..."	4 weeks ago	Up 4 weeks		pmon
f8fa2d960729	docker-sonic-mgmt-framework:latest	"/usr/local/bin/supe..."	4 weeks ago	Up 4 weeks		mgmt-framework
aacff75b4d6c	docker-lldp:latest	"/usr/bin/docker-lld..."	4 weeks ago	Up 4 weeks		lldp
f96849265f1a	4bc1d2cc1192	"/usr/bin/docker_ini..."	4 weeks ago	Up 4 weeks		dhcp_relay
4e44e2828dbf	docker-fpm-frr:latest	"/usr/bin/docker_ini..."	4 weeks ago	Up 4 weeks		bgp
7fe7884038ba	docker-router-advertiser:latest	"/usr/bin/docker_ini..."	4 weeks ago	Up 4 weeks		radv
cbb436366c53	docker-syncd-cisco:latest	"/usr/local/bin/supe..."	4 weeks ago	Up 4 weeks		syncd
ce9b7d9b6b0f	docker-teamd:latest	"/usr/local/bin/supe..."	4 weeks ago	Up 4 weeks		teamd
9aa4cfff8175	docker-orchagent:latest	"/usr/bin/docker_ini..."	4 weeks ago	Up 4 weeks		swss
499884e6c90f	docker-eventd:latest	"/usr/local/bin/supe..."	4 weeks ago	Up 4 weeks		eventd
91474e3f9111	docker-database:latest	"/usr/local/bin/dock..."	4 weeks ago	Up 4 weeks		database

SONiC show CLI examples

admin@sonic:~\$ show interfaces status

Interface	Lanes	Speed	MTU	FEC	Alias	Vlan	Oper	Admin	Type	Asym	PFC
Ethernet0	2304, 2305, 2306, 2307, 2308, 2309, 2310, 2311	400G	9100	N/A	etp0	routed	down	up	N/A		N/A
Ethernet8	2320, 2321, 2322, 2323, 2324, 2325, 2326, 2327	400G	9100	N/A	etp1	routed	down	up	N/A		N/A
Ethernet16	2312, 2313, 2314, 2315, 2316, 2317, 2318, 2319	400G	9100	N/A	etp2	routed	down	up	N/A		N/A
Ethernet24	2056, 2057, 2058, 2059, 2060, 2061, 2062, 2063	400G	9100	N/A	etp3	routed	down	up	N/A		N/A
Ethernet32	1792, 1793, 1794, 1795, 1796, 1797, 1798, 1799	400G	9100	N/A	etp4	routed	down	up	N/A		N/A
Ethernet40	2048, 2049, 2050, 2051, 2052, 2053, 2054, 2055	400G	9100	N/A	etp5	routed	down	up	N/A		N/A
Ethernet48	2560, 2561, 2562, 2563, 2564, 2565, 2566, 2567	400G	9100	N/A	etp6	routed	down	up	N/A		N/A
Ethernet56	2824, 2825, 2826, 2827, 2828, 2829, 2830, 2831	400G	9100	N/A	etp7	routed	down	up	N/A		N/A
Ethernet64	2832, 2833, 2834, 2835, 2836, 2837, 2838, 2839	400G	9100	N/A	etp8	routed	down	up	N/A		N/A
Ethernet72	2816, 2817, 2818, 2819, 2820, 2821, 2822, 2823	400G	9100	N/A	etp9	routed	down	up	N/A		N/A
Ethernet80	2568, 2569, 2570, 2571, 2572, 2573, 2574, 2575	400G	9100	N/A	etp10	routed	down	up	N/A		N/A
Ethernet88	2576, 2577, 2578, 2579, 2580, 2581, 2582, 2583	400G	9100	N/A	etp11	routed	down	up	N/A		N/A
Ethernet96	1536, 1537, 1538, 1539, 1540, 1541, 1542, 1543	400G	9100	N/A	etp12	routed	down	up	N/A		N/A
Ethernet104	1800, 1801, 1802, 1803, 1804, 1805, 1806, 1807	400G	9100	N/A	etp13	routed	down	up	N/A		N/A
Ethernet112	1552, 1553, 1554, 1555, 1556, 1557, 1558, 1559	400G	9100	N/A	etp14	routed	down	up	N/A		N/A
Ethernet120	1544, 1545, 1546, 1547, 1548, 1549, 1550, 1551	400G	9100	N/A	etp15	routed	down	up	N/A		N/A
Ethernet128	1296, 1297, 1298, 1299, 1300, 1301, 1302, 1303	400G	9100	N/A	etp16	routed	down	up	N/A		N/A
Ethernet136	1288, 1289, 1290, 1291, 1292, 1293, 1294, 1295	400G	9100	N/A	etp17	routed	down	up	N/A		N/A
Ethernet144	1280, 1281, 1282, 1283, 1284, 1285, 1286, 1287	400G	9100	N/A	etp18	routed	down	up	N/A		N/A
Ethernet152	1032, 1033, 1034, 1035, 1036, 1037, 1038, 1039	400G	9100	N/A	etp19	routed	down	up	N/A		N/A
Ethernet160	264, 265, 266, 267, 268, 269, 270, 271	400G	9100	N/A	etp20	routed	down	up	N/A		N/A
Ethernet168	272, 273, 274, 275, 276, 277, 278, 279	400G	9100	N/A	etp21	routed	down	up	N/A		N/A
Ethernet176	16, 17, 18, 19, 20, 21, 22, 23	400G	9100	N/A	etp22	routed	down	up	N/A		N/A
Ethernet184	0, 1, 2, 3, 4, 5, 6, 7	400G	9100	N/A	etp23	routed	down	up	N/A		N/A
Ethernet192	256, 257, 258, 259, 260, 261, 262, 263	400G	9100	N/A	etp24	routed	down	up	N/A		N/A
Ethernet200	8, 9, 10, 11, 12, 13, 14, 15	400G	9100	N/A	etp25	routed	down	up	N/A		N/A
Ethernet208	1024, 1025, 1026, 1027, 1028, 1029, 1030, 1031	400G	9100	N/A	etp26	routed	down	up	N/A		N/A
Ethernet216	768, 769, 770, 771, 772, 773, 774, 775	400G	9100	N/A	etp27	routed	down	up	N/A		N/A
Ethernet224	520, 521, 522, 523, 524, 525, 526, 527	400G	9100	N/A	etp28	routed	down	up	N/A		N/A
Ethernet232	776, 777, 778, 779, 780, 781, 782, 783	400G	9100	N/A	etp29	routed	down	up	N/A		N/A
Ethernet240	512, 513, 514, 515, 516, 517, 518, 519	400G	9100	N/A	etp30	routed	down	up	N/A		N/A
Ethernet248	528, 529, 530, 531, 532, 533, 534, 535	400G	9100	N/A	etp31	routed	down	up	N/A		N/A



Configuring SONiC: CLI

```
root@sonic:/home/admin# config interface ip add Ethernet0 192.1.2.2/30
root@sonic:/home/admin# config interface ip add Ethernet8 192.1.2.6/30
root@sonic:/home/admin# config interface ip add Ethernet16 192.1.2.10/30
root@sonic:/home/admin# config interface ip add Ethernet24 192.1.2.14/30

root@sonic:/home/admin# config route add prefix 10.0.0.1/24 nexthop
192.1.2.1

root@sonic:/home/admin# config save -y
```



Saves modified config into
/etc/sonic/config_db.json
to preserve it across device
reloads

Configuring SONiC: sonic-cfggen utility

- The sonic-cfggen is a utility to read SONiC config from various input types and then write the config to the config database in Redis, print as a json or render a jinja2 config template.

```
root@sonic:/home/admin# sonic-cfggen -j config.json --write-to-db  
root@sonic:/home/admin# config save -y
```



Saves modified config into
/etc/sonic/config_db.json
to preserve it across device
reloads

Configuring SONiC: sonic-cfggen utility

```
{ sample-cfg.json x
Users > suahmad > Library > CloudStorage > OneDrive-Cisco > clus-24 > { sample-cfg.json > ...
1  {
2
3      "INTERFACE": {
4          "Ethernet0": {},
5          "Ethernet0|192.1.2.2/30": {},
6          "Ethernet8": {},
7          "Ethernet8|192.1.2.6/30": {},
8          "Ethernet16": {},
9          "Ethernet16|192.1.2.10/30": {},
10         "Ethernet24": {},
11         "Ethernet24|192.1.2.14/30": {}
12     },
13     "MGMT_INTERFACE": {
14         "eth0|172.24.1.2/24": {
15             "gwaddr": "172.24.1.1"
16         }
17     },
18     "MGMT_VRF_CONFIG": {
19         "vrf_global": {
20             "mgmtVrfEnabled": "true"
21         }
22     }
23 }
```

sonic-cfggen -j config.json --write-to-db
will write this config to **config_db.json**

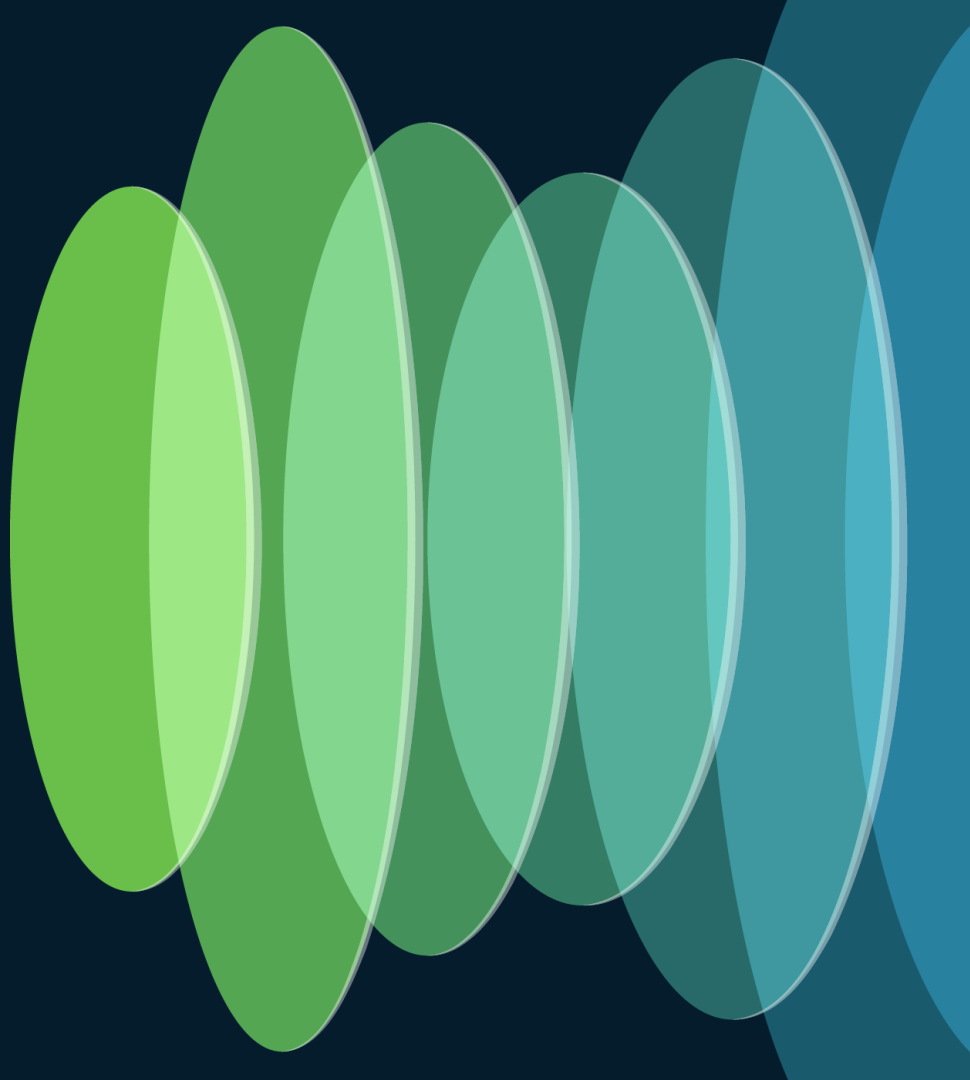
Configuring SONiC: Ansible

```
sample-playbook.yml ×
leaf > sample-playbook.yml
1 - name: Leaf IP configuration
2   gather_facts: false
3   hosts: leaf_switches
4   tasks:
5     - name: Configure IP address for Ethernet0
6       ansible.builtin.shell: config interface ip add Ethernet0 192.1.2.2/30
7
8     - name: Configure IP address for Ethernet8
9       ansible.builtin.shell: config interface ip add Ethernet8 192.1.2.6/30
10
11    - name: Configure IP address for Ethernet16
12      ansible.builtin.shell: config interface ip add Ethernet16 192.1.2.10/30
13
14    - name: Configure IP address for Ethernet24
15      ansible.builtin.shell: config interface ip add Ethernet24 192.1.2.14/30
16
17    - name: Save config
18      ansible.builtin.shell: config save -y
```

```
! sample-playbook.yml ×
Users > suhahmad > Library > CloudStorage > OneDrive-Cisco > clus-24 > ! sample-playbook.yml
1 - name: Leaf IP configuration with sonic-cfggen
2   gather_facts: false
3   hosts: leaf_switches
4   tasks:
5     - name: Copy sample-cfg.json
6       ansible.builtin.copy: /home/user/configs/sample-cfg.json
7       dest: /home/admin/sample-cfg.json
8       owner: admin
9       mode: '0644'
10
11    - name: Write sample-cfg.json to running config
12      ansible.builtin.shell: sonic-cfggen -j sample-cfg --write-to-db
13      args:
14        chdir: /home/admin/
15
16    - name: Save running config to /etc/sonic/config_db.json
17      ansible.builtin.shell: config save -y
```

Alternatively, community SONiC collections are also openly available at <https://github.com/ansible-collections/community.sonic>

SONiC support on Cisco 8000 routers

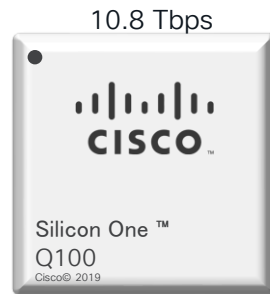
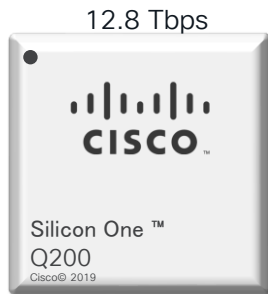
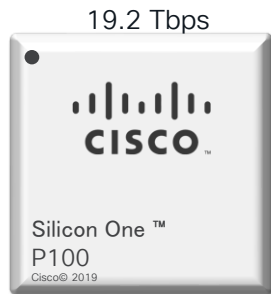


SONiC on Cisco 8000 routers

- Cisco 8000 is a family of routers based on Cisco's Silicon One ASICs designed for deployment in service provider or hyperscaler environments.
- SONiC is supported on certain routers in the Cisco 8000 family.
- Cisco 8000 routers supporting SONiC come with device drivers and a Base Support Package (BSP) which Controls board, fans, drivers, FPD, LED, etc.
- These routers ship with SONiC installed on them. New images and upgrades are available at www.software.cisco.com
- Cisco is an active participant in SONiC's open-source community and upstreams code into the SONiC open-source project.

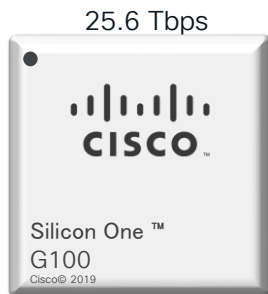
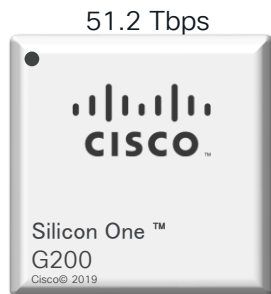
Cisco Silicon One ASICs

Cisco
8800/8600/8200
Series (w/ deep buffers)
running IOS-XR



Routing

Cisco 8100
Series
(w/o deep buffers)
running SONiC



Web Scale Switching

Core

Edge

Aggregation

Core

Peering

DCI

Large
Enterprise

Service
Provider/Web Scale
Provider

Spine

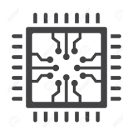
Leaf

TOR

Web Scale Provider

Cisco 8000 Routers supporting SONiC

	8101-32H	8101-64H	8101-32FH	8111-32EH	8122-64EH
FCS	Shipping	Shipping	Shipping	Shipping	Q3 2024
Ports	32x 100GbE	64x 100GbE	32x 400GbE	32x 800GbE	64x800GbE
Throughput	3.2Tbps	6.4Tbps	12.8Tbps	25.6Tbps	51.2Tbps
ASIC	Q202L (7nm)	Q201L (7nm)	Q200L (7nm)	G100 (7nm)	G200 (5nm)



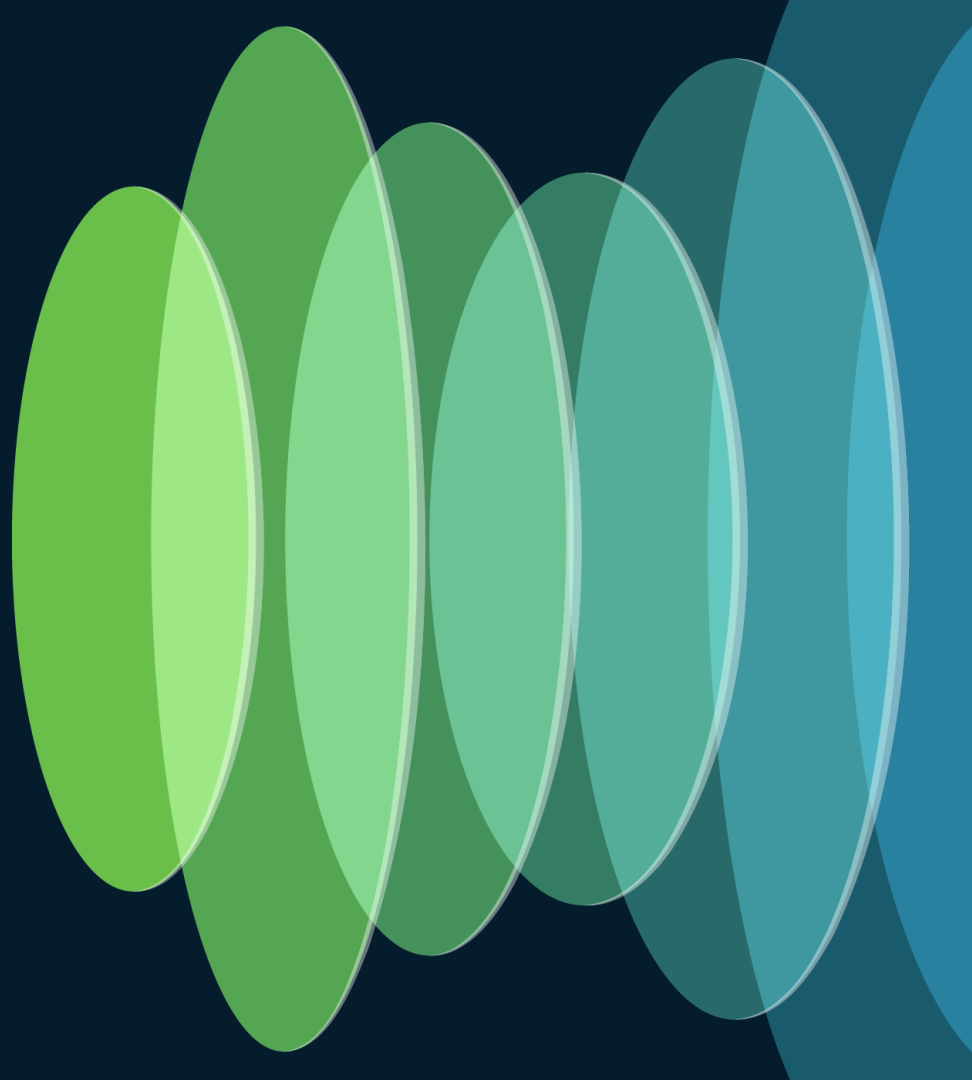
Single ASIC
Systems



Power Optimized



Conclusion



Conclusion

- SONiC is an open-source, hardware agnostic, modular Network Operating System.
- It was created by hyperscale cloud providers for their datacenter and AI/ML environments and the feature support on SONiC is still dominated by these use-cases.
- SONiC's open-source approach extends into its components which are based on other popular open-source projects such as FRR, teamd, etc.
- SONiC has support for a variety of management tools. However, advanced configuration may need multiple configuration touchpoints (e.g., ConfigDB, FRR CLI).
- Cisco supports SONiC on certain routers in the 8000 family and ships these routers with the SONiC image.

Further Reading

- SONiC on Cisco 8000 routers:
<https://www.cisco.com/c/en/us/products/routers/8000-series-routers/sonic.html>
- SONiC on Cisco 8000 at DevNet:
<https://developer.cisco.com/docs/sonic/>
- SONiC Learning Lab:
<https://developer.cisco.com/learning/labs/sonic-notebooks/getting-started-with-notebooks-for-sonic-on-the-cisco-8000-emulator/>

Complete Your Session Evaluations



Complete a minimum of 4 session surveys and the Overall Event Survey to be entered in a drawing to **win 1 of 5 full conference passes** to Cisco Live 2025.



Earn 100 points per survey completed and compete on the Cisco Live Challenge leaderboard.



Level up and earn **exclusive prizes!**



Complete your surveys in the **Cisco Live mobile app**.

Continue your education

- Visit the Cisco Showcase for related demos
- Book your one-on-one Meet the Engineer meeting
- Attend the interactive education with DevNet, Capture the Flag, and Walk-in Labs
- Visit the On-Demand Library for more sessions at www.CiscoLive.com/on-demand

Contact me at: suhahmad@cisco.com



The bridge to possible

Thank you

CISCO *Live!*

#CiscoLive