

7.1 Ansible Network Automation Trends



Changes in 2.2

- ▶ <https://www.ansible.com/blog/ansible-network-updates>
- ▶ Template module verging into Config modules, with Config modules adding more features.
- ▶ Return structured text after command execution
- ▶ Working on more Network Facts
- ▶ More community, more testing, more documentation

Changes in 2.3

- ▶ https://docs.ansible.com/ansible/2.5/porting_guides/porting_guide_2.3.html#networking
- ▶ Support of ProxyCommands
- ▶ Persistent SSH Connection

Changes in 2.4

- ▶ <https://www.ansible.com/blog/networking-features-in-ansible-2-4>
- ▶ More declarative modules
- ▶ Aggregate Resources
- ▶ Additional platforms and support

Changes in 2.5

- ▶ <https://www.ansible.com/blog/coming-soon-networking-features-in-ansible-2.5>
- ▶ network_cli and netconf connection types
- ▶ Network Facts
- ▶ Persistent SSH connection
- ▶ Network Best Practices documentation
- ▶ Better logging
- ▶ More declarative intent
- ▶ Additional Platforms and Modules

Next: Declarative Example



Ansible Modules

- ▶ Scripts that accomplish a task
- ▶ Mostly in Python, small percentage of them are in PowerShell for Windows.
- ▶ Self-made modules can be in any language
- ▶ Lots of Network Modules

Ansible Run Methods

- ▶ Use ssh to copy the Python module to remote host, run, clean up.
- ▶ Run locally and use Paramiko to ssh to host or call remote host API (This is the way most network modules are run).
- ▶ AdHoc one-liner

Ansible Playbooks

- ▶ Modules are your tools, Playbook is the instruction manual for which tool to use, when to use them, and in what order.
- ▶ No need to learn all the features at once, start small and pick up more features as you need them.

Playbook

```
1  --
2  - name: First Network Playbook
3    connection: network_cli
4    hosts: all
5    tasks:
6      - name: show version
7        ios_command:
8          commands: show version
9
10       register: output
11
12     - name: show output
13       debug:
14         var: output.stdout
15
16     - name: copy output to file
17       copy: content="{{ output }}" dest=./output/{{ inventory_hostname }}.txt
18  ...
```

Playbook

Play

Task

Modules

YAML

- ▶ Starts with “---” and end with “...”
- ▶ All members of a list are lines with the same indentation level with ‘-’ and a space
- ▶ A Dictionary are lines with the same indentation level with (key: value) pair
- ▶ Indentation is important

YAML

```
1  --
2  - name: First Network Playbook
3    connection: network_cli
4    hosts: all
5    tasks:
6      - name: show version
7        ios_command:
8          commands: show version
9
10       register: output
11
12       - name: show output
13         debug:
14           var: output.stdout
15
16       - name: copy output to file
17         copy: content="{{ output }}" dest=./output/{{ inventory_hostname }}.txt
18  ...
```

Dictionary

List

YAML Beginning / End

List

Resources

▷ Intro to Playbooks

http://docs.ansible.com/ansible/playbooks_intro.html

▷ YAML Syntax

<http://docs.ansible.com/ansible/YAMLSyntax.html>

▷ Working with Modules

http://docs.ansible.com/ansible/latest/user_guide/modules.html

▷ YAML Lint

<http://www.yamllint.com/>

Suggestion

Keep in simple, follow a pattern 😊

.... at least for now

Playbook

```
1  --
2  - name: First Network Playbook
3    connection: network_cli
4    hosts: all
5    tasks:
6      - name: show version
7        ios_command:
8          commands: show version
9
10       register: output
11
12     - name: show output
13       debug:
14         var: output.stdout
15
16     - name: copy output to file
17       copy: content="{{ output }}" dest=./output/{{ inventory_hostname }}.txt
18  ...
```

Playbook

Play

Task

Modules

YAML

```
1  --
2  - name: First Network Playbook
3    connection: network_cli
4    hosts: all
5    tasks:
6      - name: show version
7        ios_command:
8          commands: show version
9
10       register: output
11
12       - name: show output
13         debug:
14           var: output.stdout
15
16       - name: copy output to file
17         copy: content="{{ output }}" dest=./output/{{ inventory_hostname }}.txt
18  ...
```

Dictionary

List

YAML Beginning / End

List

OSPF Configuration Workflow

- ▶ Configure Basic OSPF
- ▶ Verifying OSPF Adjacencies
- ▶ Verifying the OSPF Database

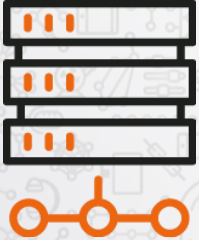
New in Version 2

“Up and Running” with Network Devices

Save Time + Be More Productive

Prerequisites

- ▶ (optional) Basic Networking Knowledge




Why Ansible?


Ansible has thousands of users, hundreds of customers and over 2,400 community contributors

600+ ANSIBLE MODULES	29,000+ GITHUB STARS	230+ MEETUPS WORLDWIDE
----------------------------	----------------------------	------------------------------

[JOIN OUR COMMUNITY](#)




“ The work the Ansible team is doing... is something the entire industry should be paying attention to.




Lew Tucker
VP & CTO, Cloud Computing
Cisco

[OpenStack Integration](#)



“ Red Hat Ansible Tower allows us to easily streamline the delivery of applications and services to both OpenStack & Amazon Clouds in a cost effective, simple, & secure manner.



Jeremy Pruitt
Cloud & Automation
Juniper

[IT Automation with Ansible Tower](#)

Credit: <https://www.ansible.com/>

Come Join me to save time and be more
productive as Network Engineers!

OSPF Configuration Workflow

- ▶ Configure Basic OSPF
- ▶ Verifying OSPF Adjacencies
- ▶ Verifying the OSPF Database

Tracking Topology Changes

▶ When a new LSA is received it is checked against the database for changes such as...

- Sequence number is used to:
 - track new vs old LSAs
- Age is used to:
 - Keep information new
 - Withdraw old information
 - Periodic flooding occurs after 30 minutes
 - “paranoid” update
 - LSAs that reach MaxAge (60 minutes) are withdrawn
- Checksum
 - Used to avoid transmission & memory corruption

Tracking Topology Changes

- ▶ When a new LSA is received it is checked against the database for changes such as...
 - Sequence number
 - Age
 - Used to keep information new
 - Withdraw old information
 - Periodic flooding occurs after 30 minutes
 - LSAs that reach MaxAge (60 minutes) are withdrawn
 - Checksum
 - Used to avoid transmission & memory corruption

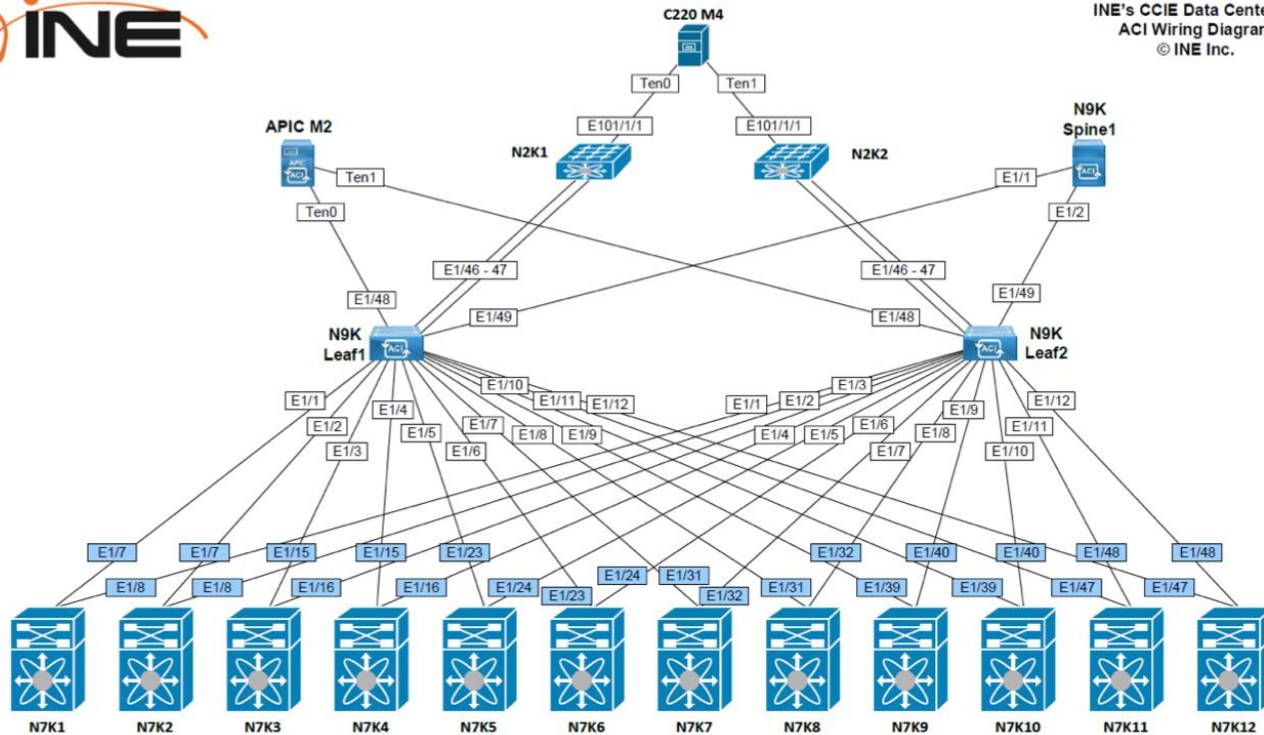
Tracking Topology Changes

- ▶ When a new LSA is received it is checked against the database for changes such as...
 - Sequence number
 - Age
 - Used to keep information new and withdraw old information
 - Periodic flooding occurs after 30 minutes
 - LSAs that reach MaxAge (60 minutes) are withdrawn
 - Checksum
 - Used to avoid transmission & memory corruption

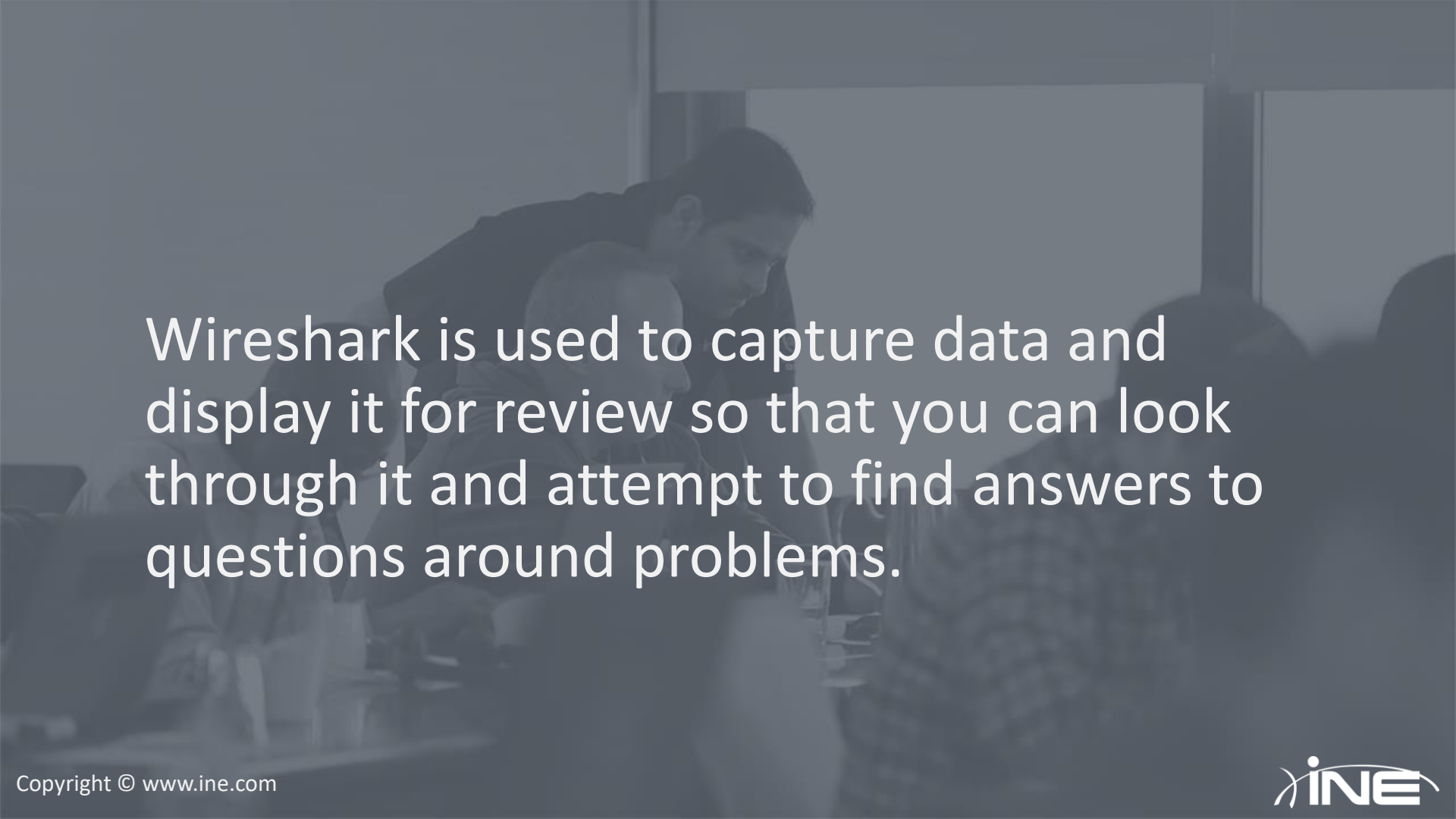
Tracking Topology Changes

▶ When a new LSA is received it is checked against the database for changes such as...

- Sequence number
- Age
- Checksum



Product/Topology Diagram



Wireshark is used to capture data and display it for review so that you can look through it and attempt to find answers to questions around problems.