



# DevNet Experts.

Topic- CCNA- Devnet

 +91 9892028199

 devnetexperts@gmail.com



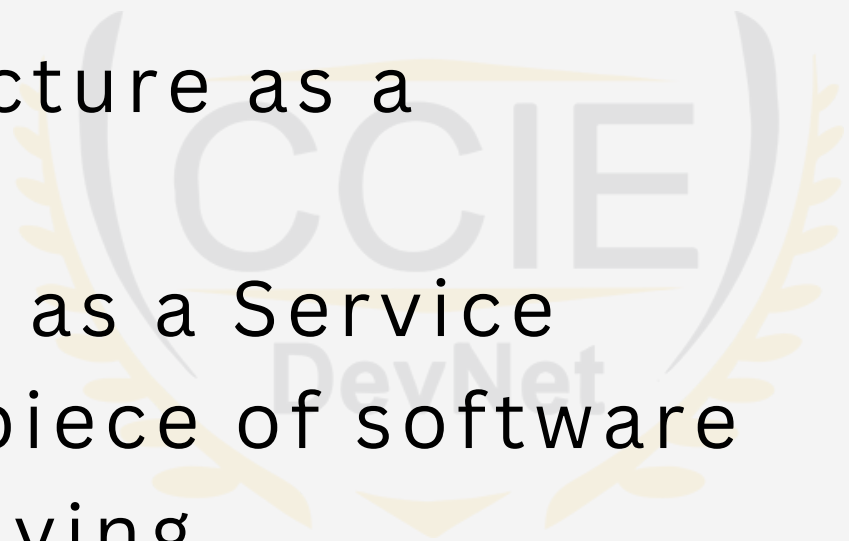
# OVERVIEW

- Software Development and Design
- Understanding and Using APIs
- Cisco Platforms and Development
- Application Deployment and Security
- Infrastructure and Automation
- Network Fundamentals



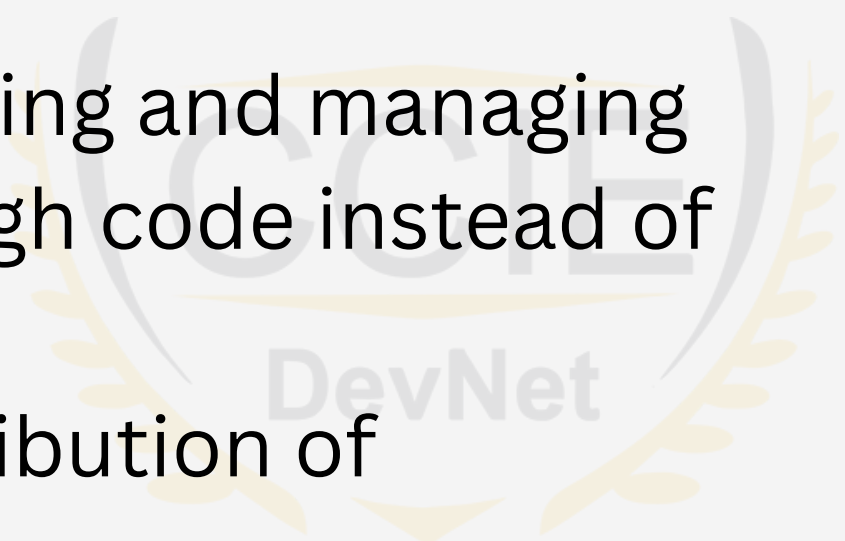
# Software Development & Design.

- With the cloud computing popularity, concepts like IaaS and PaaS are gaining popularity as well
- IaaS stands for Infrastructure as a Service
- PaaS stands for Platform as a Service
- Service here refers to a piece of software that abstracts the underlying infrastructure or platform component
- This makes it easy for them to be provisioned, configured, managed, and decommissioned



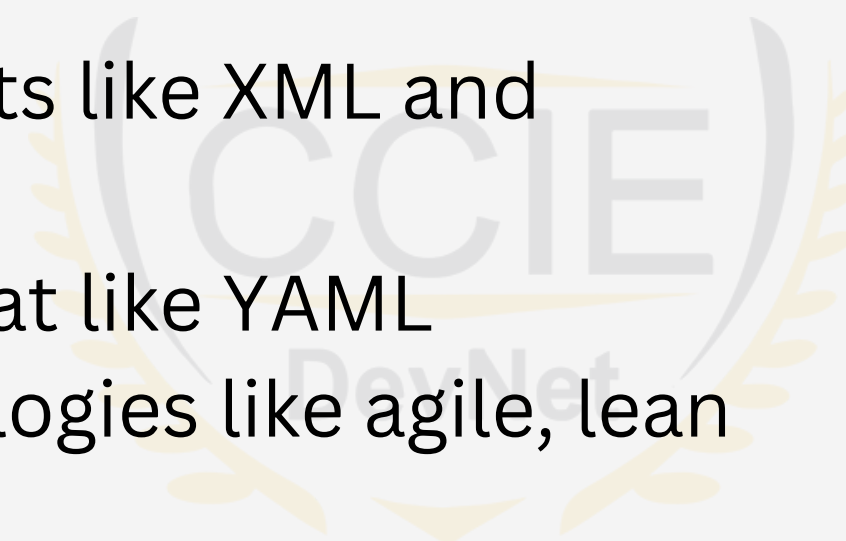
# Software Development & Design (cont.)

- Another related concept is Infrastructure as Code (IaC)
- It is the practice of provisioning and managing infrastructure defined through code instead of manual process
- Allows easy editing and distribution of configuration
- Always ensures that infrastructure is in its desired state
- This results in reliable and reproducible infrastructure configurations



# Software Development & Design (cont.)

- Hence it is necessary for network engineers to be familiar with software development and design principles like:
  1. Usage of data interchange formats like XML and JSON
  2. Configuration specification format like YAML
  3. Software development methodologies like agile, lean and waterfall
  4. Organizing the code into methods, functions, classes and modules
  5. Different design patterns like MVC (Model View Controller) and Observer
  6. Advantages of version controlling the code
  7. Working with version control tools like Git



# Cisco Webex XML API Response

```
<?xml version="1.0" encoding="UTF-8"?>
<serv:message xmlns:serv="http://www.webex.com/schemas/2002/06/service"
  xmlns:com="http://www.webex.com/schemas/2002/06/common"
  xmlns:meet="http://www.webex.com/schemas/2002/06/service/meeting"
  xmlns:att="http://www.webex.com/schemas/2002/06/service/attendee">
  <serv:header>
    <serv:response>
      <serv:result>SUCCESS</serv:result>
      <serv:gsbStatus>PRIMARY</serv:gsbStatus>
    </serv:response>
  </serv:header>
  <serv:body>
    <serv:bodyContent xsi:type="meet:createMeetingResponse"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
      <meet:meetingkey>48591508</meet:meetingkey>
      <meet:iCalendarURL>
        <serv:host>https://www.webex.com/calendarurl1/j.php?ED=48591508&U
          ID=BA24987F&ICS=MIFH&ST=12</serv:host>
        <serv:attendee>https://www.webex.com/calendarurl1/j.php?ED=485915
          08&UID=BA24987F&ICS=MIFA&ST=12</serv:attendee>
      </meet:iCalendarURL>
      <meet:guestToken>f10324e2af4823c278fa1a6efadc426c</meet:guestToken>
    </serv:bodyContent>
  </serv:body>
</serv:message>
```

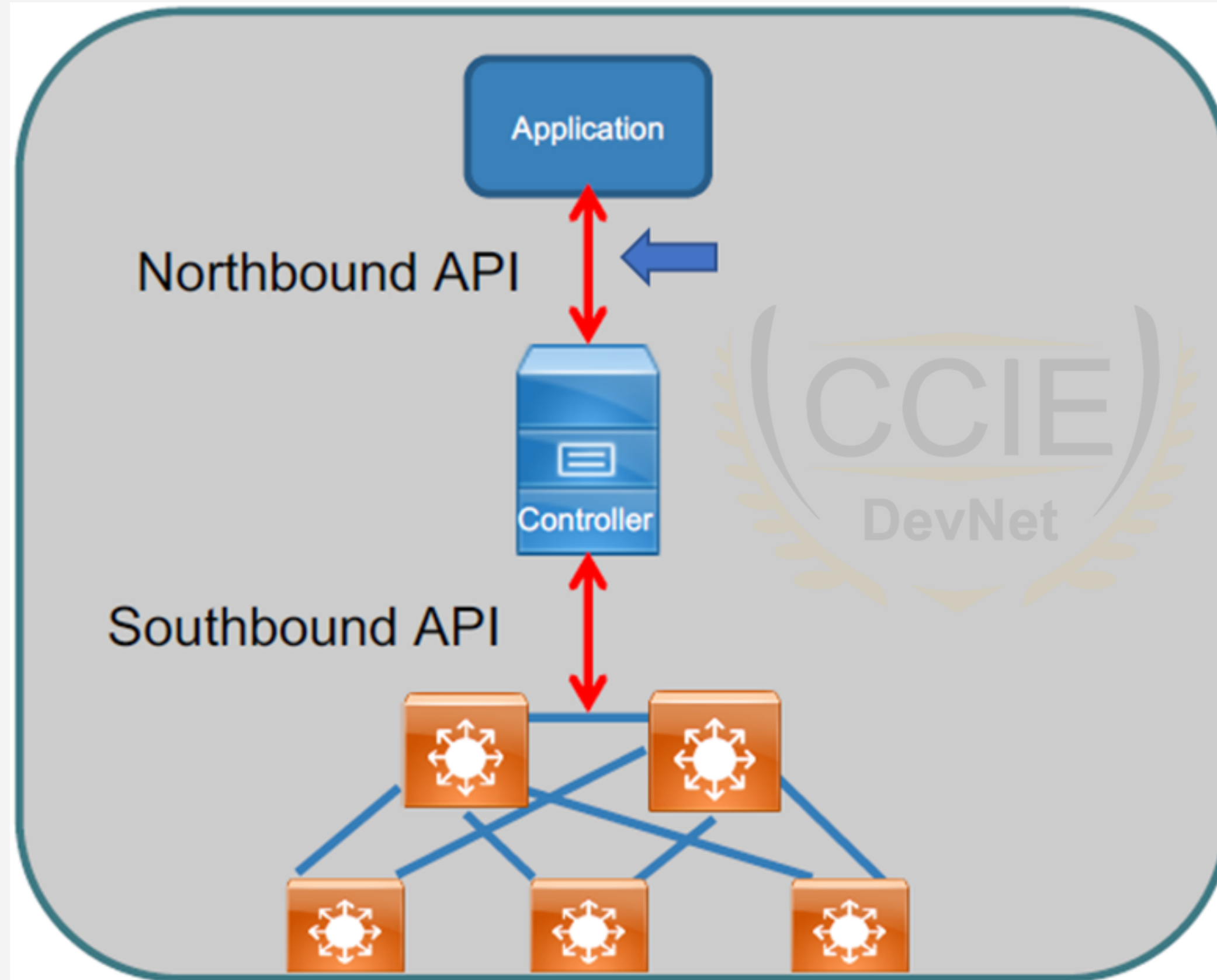


# Understanding and Using APIs

- An Application Programming Interface (API) defines a set of rules that explain how two computers or two applications on those computers communicate with each other
- APIs enable companies to open up their applications' data and functionality to third party developers or business partners
- This was an important milestone in opening up closed devices and software like network infrastructure to a wide range of integration with monitoring or orchestration and other applications



# Understanding and Using APIs (cont.)





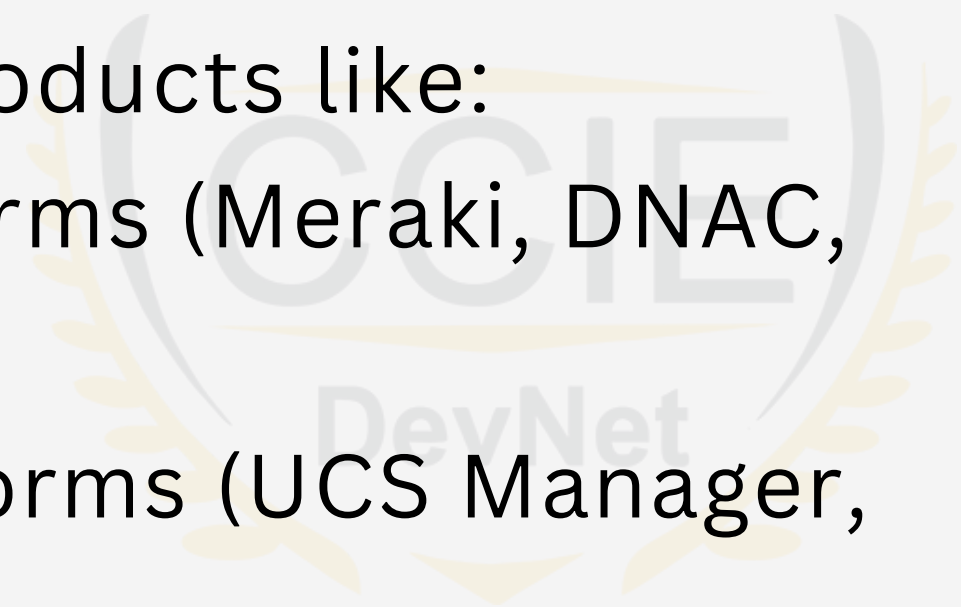
# Understanding and Using APIs (cont.)

- For network engineers to be able to integrate network devices and application with third party tools the following expertise is needed
  1. Ability to understand API documentation and execute API requests
  2. Working with webhooks to consume data received from third party integrations
  3. Troubleshoot API issues based on HTTP error messages, API request and documentation
  4. Understand different API authentication mechanisms (basic, bearer tokens, API keys)
  5. Writing Python scripts to make API calls using requests library



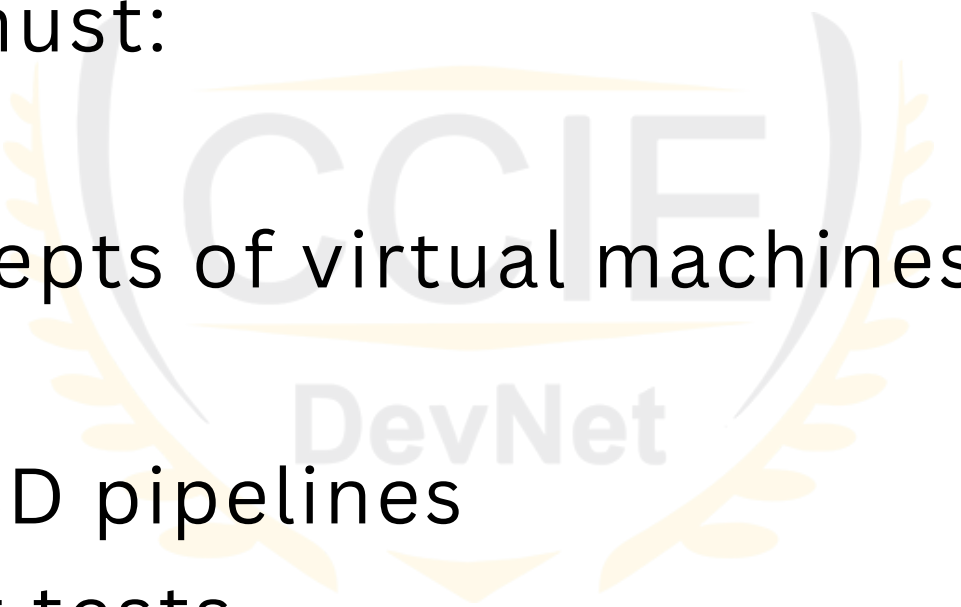
# Cisco Platforms & Development.

- In view of the growing popularity of Software Defined Networking and Infrastructure as Code concepts, Cisco has been providing SDKs, exposing APIs for all their products like:
  1. Network management platforms (Meraki, DNAC, ACI, NSO, SD-WAN)
  2. Compute management platforms (UCS Manager, UCS Director, Intersight)
  3. Collaboration platforms (Webex, Unified Communication Manager, Finesse)
  4. Security platforms (Firepower, Umbress, AMP, ISE)

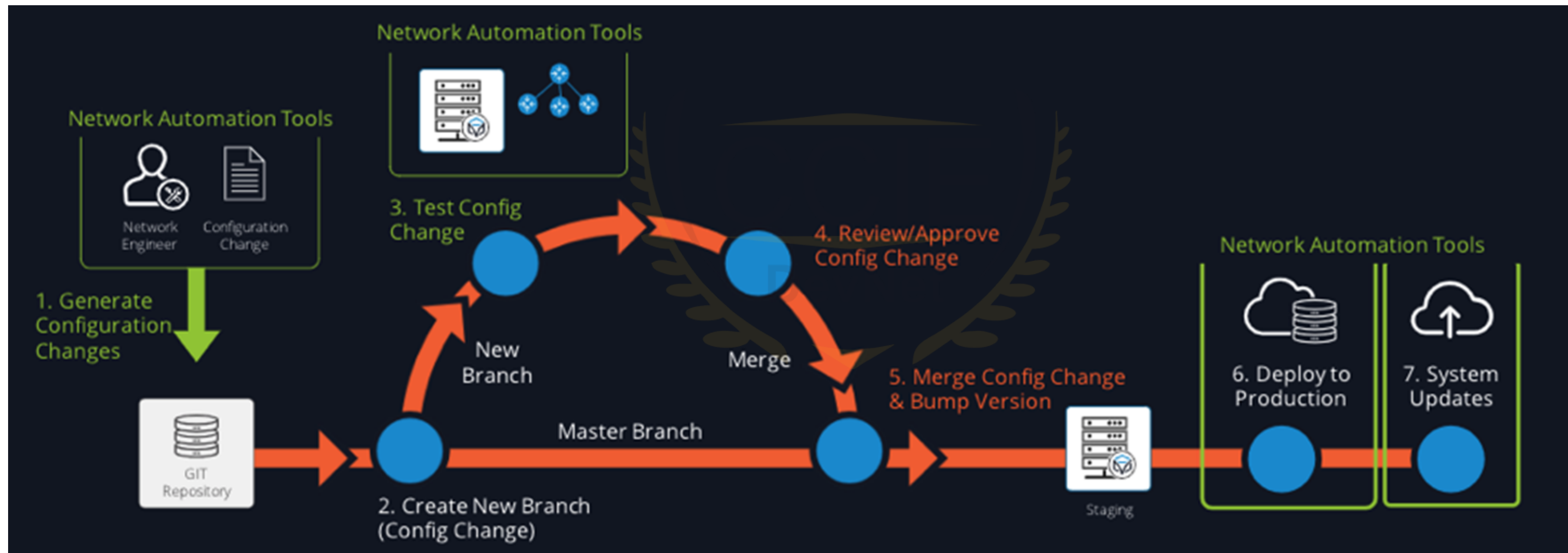


# Application Deployment and Security.

- Once we develop application using the integration points provided by Cisco and other vendors, it's time to think of deploying these applications in a secure way.
- Hence a network engineer must:
  1. Be able to understand concepts of virtual machines, bare metal, containers
  2. Understand concept of CI/CD pipelines
  3. Be able to write Python unit tests
  4. Understand working with Docker
  5. Understand firewalls, DNS, load balancers and reverse proxy
  6. Be able to understand top OWASP threats like XSS, SQL Injection CSRF

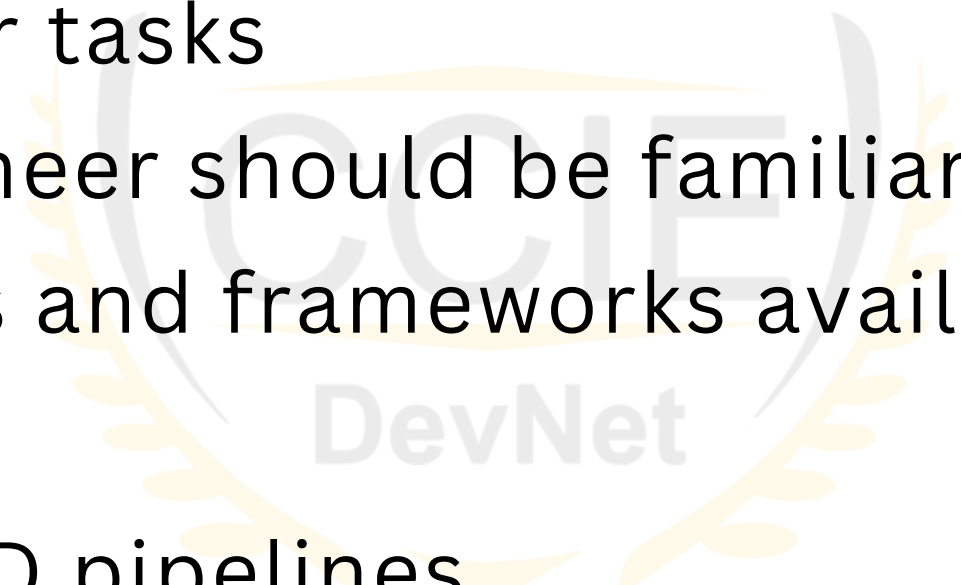


# Pipeline to generate, test, deploy config changes



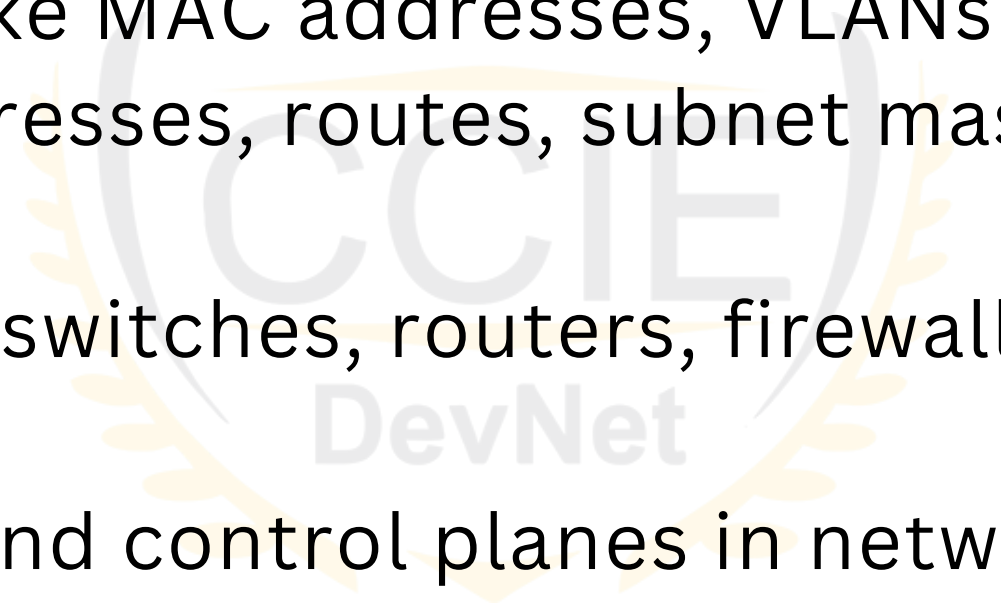
# Infrastructure and Automation.

- Now that we have code which gets deployed on different kinds of infrastructure, it's time to think of automating most of our tasks
- For this the network engineer should be familiar with the automation tools and frameworks available out there:
  1. Infra automation via CI/CD pipelines
  2. Tools such as Ansible, Chef, Puppet, Cisco NSO
  3. Python script using APIs for ACI, DNAC, RESTCONF
  4. Interpreting and using YANG data models
  5. Use network simulation tools such as VIRL/ PyATS



# Network Fundamentals.

- Having said that, the most important element to make all this network automation efforts successful is the domain knowledge
- Needless to say you should be familiar with:
  1. Basic networking concepts like MAC addresses, VLANs
  2. Purpose and usage of IP Addresses, routes, subnet mask, prefix and gateways
  3. Networking components like switches, routers, firewalls and load balancers
  4. Roles of management, data and control planes in network devices
  5. Describing functionalities of DHCP, DNS, NAT, SNMP, NTP
  6. Protocols such as SSH, Telnet, HTTP(S), NETCONF, RESTCONF
  7. Identifying connectivity issues like NAT problem, Ports being blocked
  8. Explaining impact of network constraints on applications



CCNA

DEMO.

