

# NetDevOps en la Red de la Universidad de Salamanca

José Manuel Agudo - [jagudo@usal.es](mailto:jagudo@usal.es)



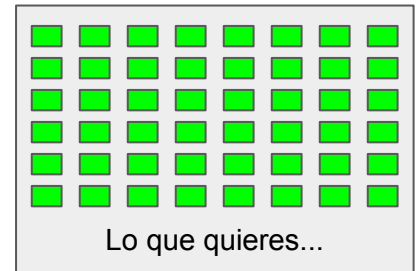
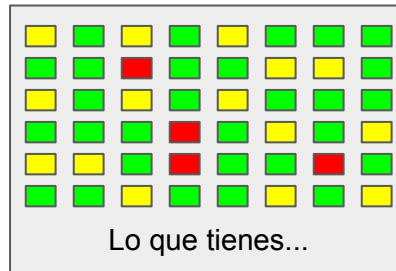
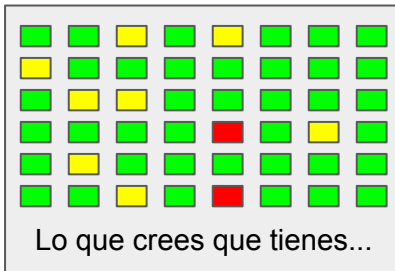
VNIVERSIDAD  
DSALAMANCA  
1218 - 2018

# NetDevOps - Definición

“Incorporar la cultura, métodos, estrategias y mejores prácticas de DevOps a las redes” -  
Hank Preston - Cisco Network  
Programmability Evangelist

# NetDevOps - La vida antes de...

- Configurar cientos de equipos de forma manual (Red de Campus de la USAL = 400 switches)
- Cambios individuales para solucionar problemas en un switch no son aplicados a switch similares (con lo que el problema reaparece en otro sitio)
- Cambios generales implican un gran esfuerzo para su despliegue
- Falta de consistencia que se va acumulando:



# NetDevOps - Cultura del Miedo al Cambio

## “Culture of Fear” Reinforcing Loop



# NetDevOps - Cultura del Cambio

## “Culture of Change” Loop



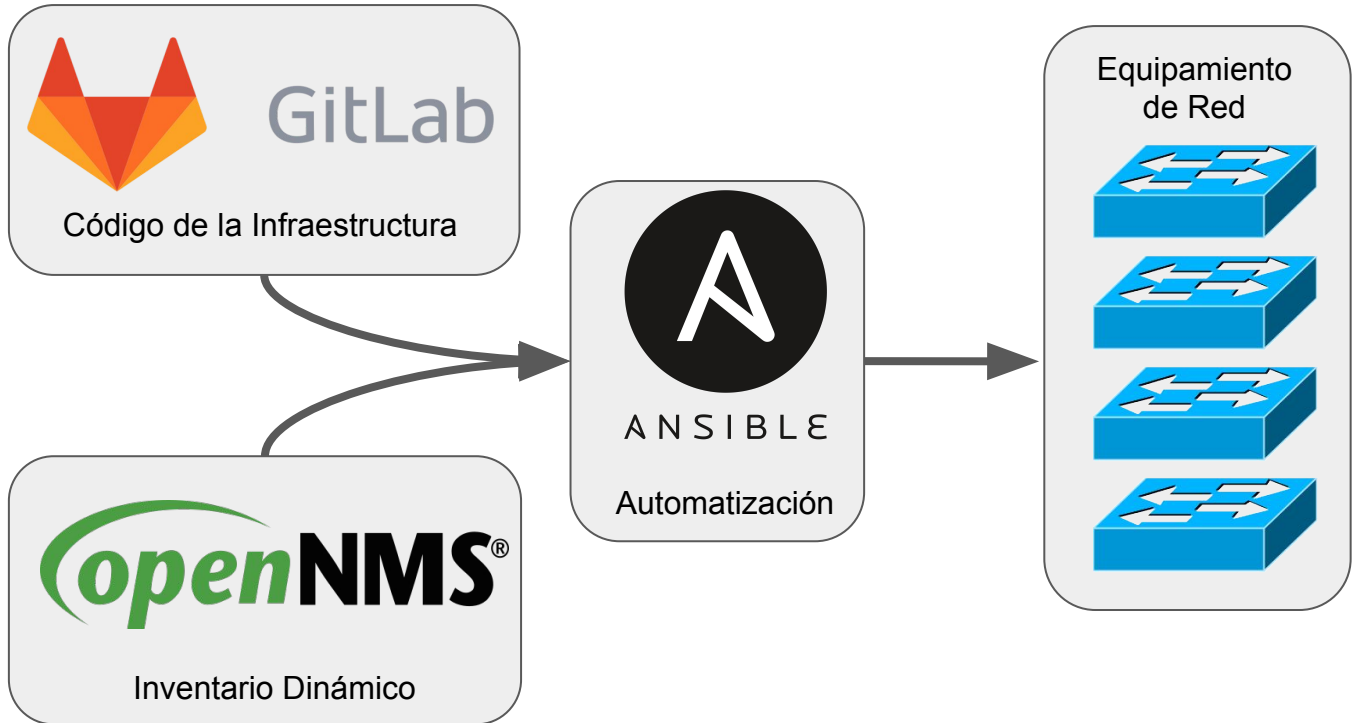
# NetDevOps - Métodos y Estrategias

- Automatización:
  - Reduciendo la intervención directa en los equipos de forma que se minimice el tiempo dedicado a tareas repetitivas y los posibles errores humanos a la hora de aplicar cambios en cientos de equipos
  - Reduciendo el tiempo de aplicación de los cambios
  - Aumentando la consistencia de las configuraciones
- Infraestructura como Código:
  - Repositorio central de configuración
  - Estableciendo políticas unificadas para todos los elementos de red
- Integración Continua:
  - Realización de pruebas y despliegues automatizados

# NetDevOps - Métodos y Estrategias

- Verificación Continua:
  - Estableciendo comprobaciones periódicas del estado de la infraestructura
  - Configuración automatizada de la monitorización
  - Cambios ligados al estado de la red
- Evolución hacia una red basada en la intención (Intent-Based Networking) que permita aplicar políticas generales que se traduzcan y apliquen en el equipamiento de red directamente por las herramientas de automatización:
  - Definición de políticas de acceso a recursos en función de los roles
  - Definición de recursos global aplicada a diferente equipamiento: balanceadores, firewalls, switches, ...

# NetDevOps - Solución

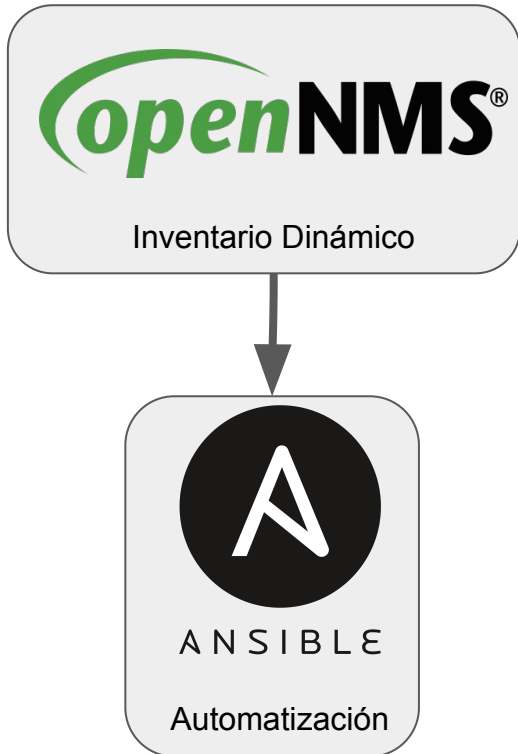




# NetDevOps - Ansible

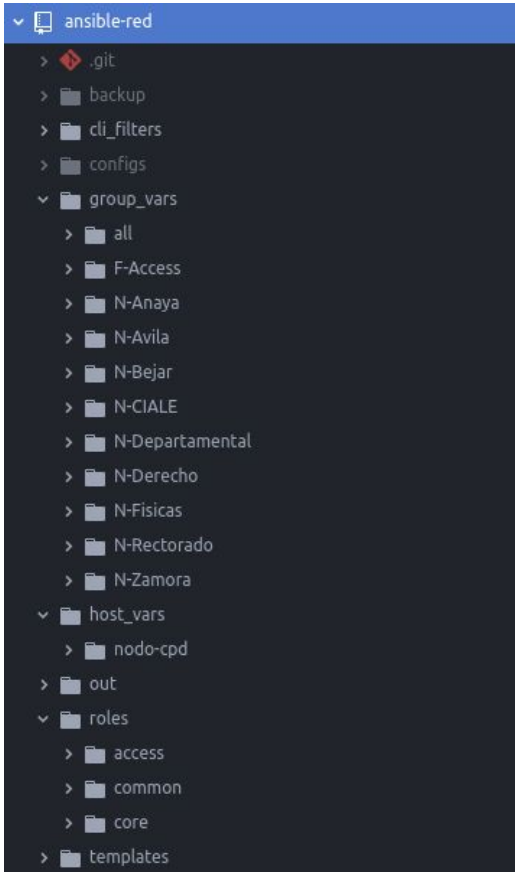
- Herramienta de Gestión de la Configuración y Automatización
- Enfocada en la simplicidad y la facilidad de uso
- Sin agentes, conexiones con los equipos por: SSH, API, ...
- Extensible mediante módulos, con cientos de ellos disponibles para los principales fabricantes de equipamiento de red, en concreto con soporte para los que necesitábamos:
  - Cisco (Catalyst / Nexus)
  - Paloalto
  - F5
- También permite: automatización de sistemas, aplicaciones, cloud, ...

# NetDevOps - Inventario



- Herramienta de Gestión de Red OpenSource
- Desarrollo de un script en Python de inventario dinámico usando el API RESTful de OpenNMS
- Proporciona información sobre cada equipo: nombre, IP, edificio, ciudad, modelo, versión sistema operativo
- Categorías a las que pertenece:
  - Función: F-Access, F-Distribution, F-Core
  - Tipo: T-Switch, T-Router, T-Firewall
  - Nodo: N-Anaya, N-Bejar, N-Avila, N-Zamora

# NetDevOps - Ansible Estructura



- Variables por categoría y específicas por equipo:
  - comunidades snmp
  - servidores de logs
  - servidores de tiempo
  - vlans
- Roles aplicados por categoría:
  - **common**: configuración general (usuarios, snmp, logs, ntp)
  - **access**: configuración equipos de acceso
  - **core**: configuración equipos de core

# NetDevOps - Ansible PlayBooks = Tareas

```
# file: roles/access/tasks/main.yml
- name: Generar Configuracion General IOS 12
  template:
    src: ios-access-general-ios12.j2
    dest: configs/{{inventory_hostname}}/ios-access-general.txt
  tags: general
  when: description | search("Version 12")

- name: Generar Configuracion General IOS 15
  template:
    src: ios-access-general-ios15.j2
    dest: configs/{{inventory_hostname}}/ios-access-general.txt
  tags: general
  when: description | search("Version 15")
```

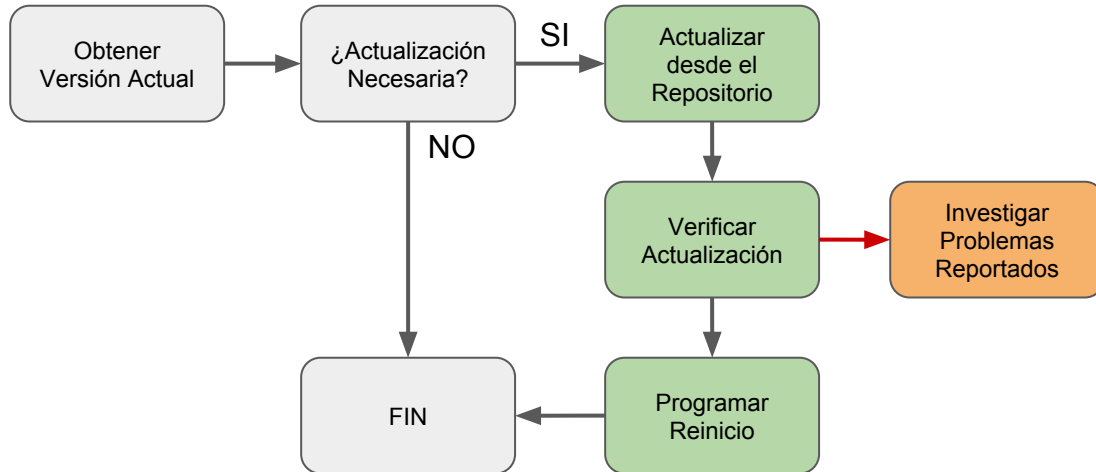
```
- name: Configuracion General
  ios_config:
    src: configs/{{inventory_hostname}}/ios-access-general.txt
    provider: "{{ ios_cli }}"
    diff_against: running
    backup: yes
  notify:
    - ios_save
  tags: general
```

# NetDevOps - Ansible Templates = Plantillas

```
service timestamps debug datetime localtime
service timestamps log datetime localtime
service password-encryption
clock summer-time SMET recurring
no ip domain-lookup
ip domain-name red.usal.es
logging trap {{ logging_level }}
logging source-interface {{ logging_source }}
{# If snmp_community_ro is defined configure it #}
{% if snmp_community_ro is defined %}
snmp-server community {{ snmp_community_ro }} RO
{% endif%}
{# If snmp_community_rw is defined configure it #}
{% if snmp_community_rw is defined %}
snmp-server community {{ snmp_community_rw }} RW
{% endif%}
{# if snmp_source_interface is defined configure it #}
{% if snmp_source_interface is defined %}
snmp-server trap-source {{ snmp_source_interface }}
{% endif%}
```

```
{# Configure snmp location #}
{% if location is defined %}
snmp-server location {{ city }} - {{ location }}
{% endif%}
{# If snmp_servers are defined configure it #}
{% if snmp_servers is defined %}
{% for snmp_item in snmp_servers %}
snmp-server host {{ snmp_item }} {{ snmp_community_ro }}
{% endfor %}
{% endif%}
{# If ntp_source is defined configure it #}
{% if ntp_source is defined %}
ntp source {{ ntp_source }}
{% endif%}
{# Configure ntp_servers #}
{% if ntp_servers is defined %}
{% for ntp_item in ntp_servers %}
ntp server {{ ntp_item }}
{% endfor %}
{% endif%}
```

# NetDevOps - Actualizaciones de Software (I)



## Variables:

- Modelo
  - Versión
  - Imagen
- Modelo
  - Versión
  - Imagen

## WS-C2960X-48FPD-L:

- version: 15.2(2)E7
- image: c2960x-universalk9-tar.152-2.E7.tar

## WS-C2960-48TC-L:

- version: 15.0(2)SE11
- image: c2960-lanbasek9-tar.150-2.SE11.tar

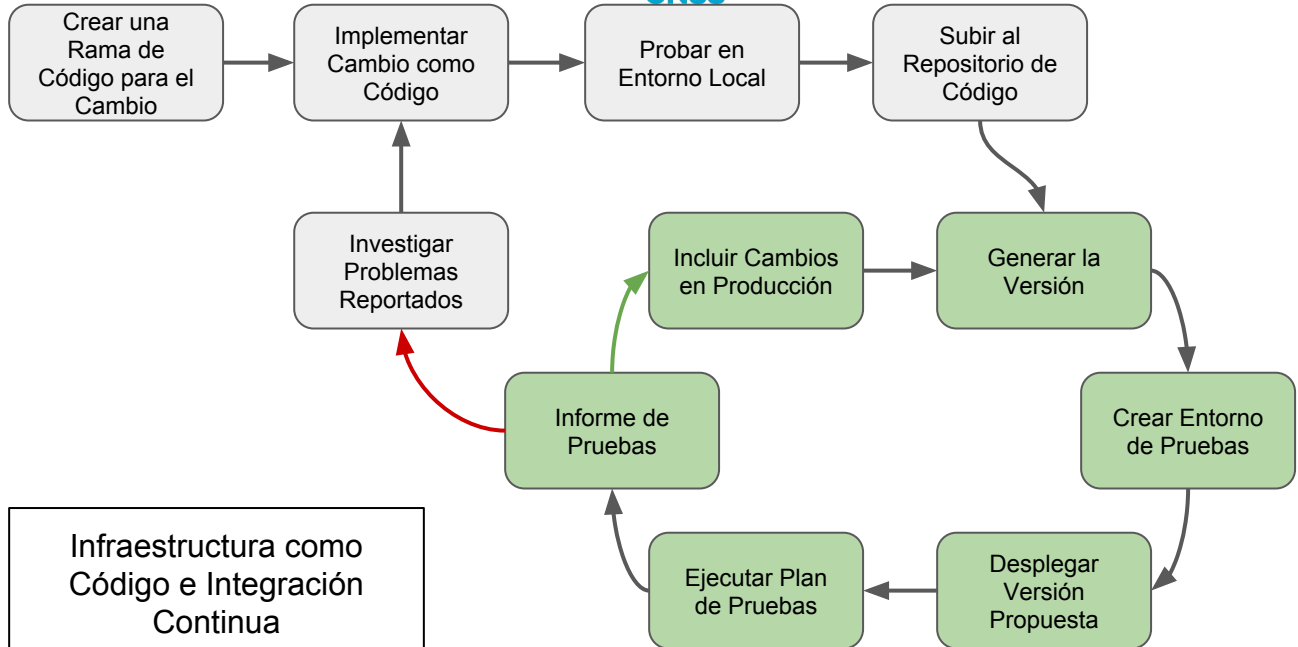
# NetDevOps - Actualización de Software (II)

```
- name: Comprobar Version
set_fact:
  upgrade_needed: true
when:
  - software_images[model] is defined
  - ansible_net_version is defined
  - ansible_net_version != software_images[model]['version']
```

```
- name: Actualizacion Catalyst 2960
ios_command:
  provider: "{{ ios_cli }}"
  timeout: 1800
  commands:
    - archive download-sw /safe
  ftp://{{repo_user}}:{{repo_password}}@{{repo_ip}}/{{software_images[model]['image']}}
  register: ios_upgrade
when:
  - upgrade_needed is defined
  - model | search("WS-C2960")
```

```
- name: Reinicio Programado
ios_config:
  provider: "{{ ios_cli }}"
  src: templates/kron-reload.j2
when:
  - ios_upgrade is defined
  - "'All software images installed.'" in ios_upgrade.stdout"
```

# NetDevOps - Proceso de Configuración





# NetDevOps - Resultados

- Configuración general de los equipos:
  - Usuarios, SNMP, NTP, Syslog
  - Opciones generales en función del tipo: F-Core, F-Distribution, F-Access
  - Documentación dinámica de la red mediante definiciones automatizadas de las descripciones de los interfaces de red
- Configuraciones basadas en roles (Acceso, Distribución, Core) y en localizaciones físicas (por nodo, por ciudad, ...)
- Configuración de equipos de acceso (F-Access):
  - Plantillas para puertos en modo trunk y de acceso
  - VLANs, Spanning Tree
  - Seguridad nivel 2: DHCP Snooping, PortSecurity, StormControl
- Gestión de actualizaciones de software

# NetDevOps - Sigüientes Pasos

- Mejorar la integración continua y las pruebas automatizadas mediante: GitLab CI y GitLab Runner
- Ejecución continua de playbooks de verificación y configuración mediante AWX (versión OpenSource de Ansible Tower)
- Configuración de equipos de F-Core y F-Distribution:
  - Gestión de ACLs
  - Interfaces nivel 3
  - Routing Dinámico
- Configuración de equipos centrales:
  - Switches Cisco Nexus
  - Firewalls PaloAlto
  - Balanceadores F5

# Gracias



VNIVERSIDAD  
DSALAMANCA  
1218 - 2018