



CISCO

Techtorial

The Power of the **Platform**





Konfig-guru Ansible-lel

Hogyan varázsoljunk el Cisco eszközöket?

Czifra János
SCI-Network
Network Engineer

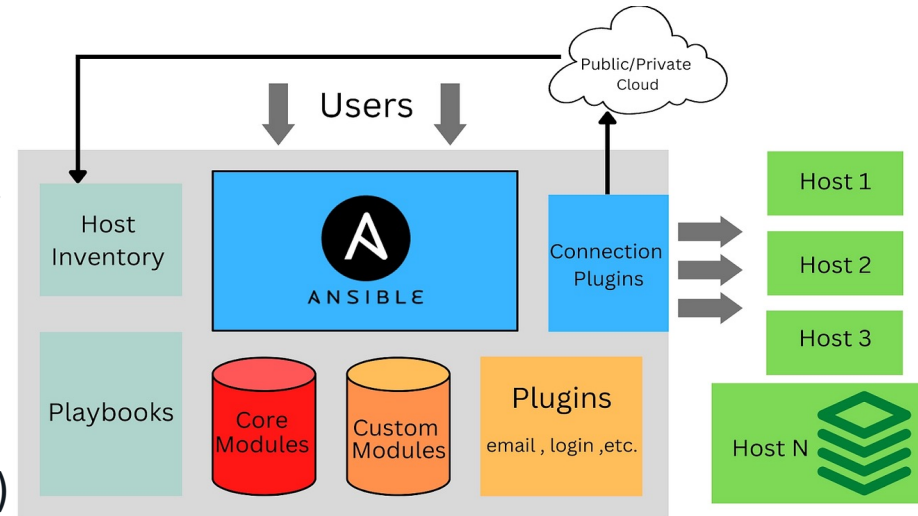
Mi az Ansible?

- Automatizálási eszköz: Segít automatizálni a konfigurációs feladatokat, alkalmazások üzembe helyezését és IT rendszerek menedzselését.
- YAML alapú leíró nyelv: Az Ansible egyszerű, emberi olvasásra is könnyen érthető YAML fájlokat (Playbook-okat) használ a feladatok leírásához.
- Agentless (ügynöknélküli): Nincs szükség külön kliensszoftver telepítésére a célrendszereken, mivel SSH vagy API kapcsolaton keresztül működik.
- Idempotens működés: Az Ansible biztosítja, hogy a rendszeren végrehajtott változtatások csak akkor történnek meg, ha szükséges, ezzel elkerülve a redundáns változtatásokat.
- Skálázhatóság: Képes egyszerre több száz vagy akár több ezer eszközt kezelni és konfigurálni.
- Feltételes végrehajtás: Képes feltételeket használni, amelyek alapján dönt, hogy végrehajt egy műveletet vagy sem (pl. csak akkor fut, ha az eszközön egy adott konfiguráció még nem létezik).

Ansible Playbook felépítése

Playbook egy YAML fájl, amely meghatározza a futtatandó feladatokat.

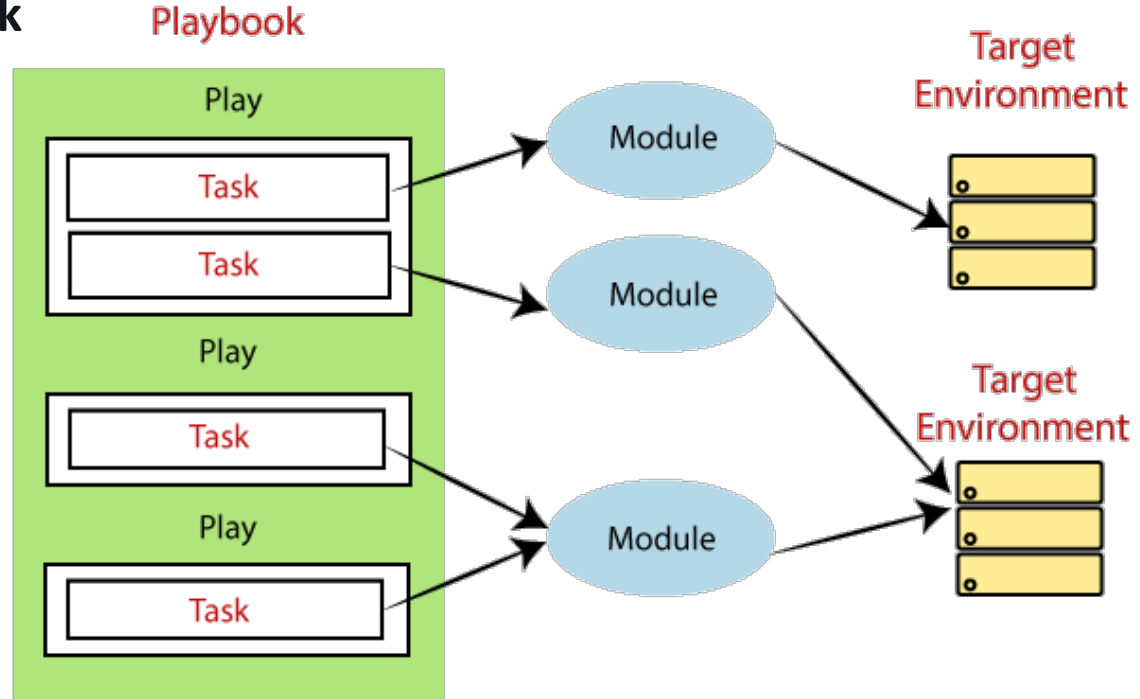
- Fontos részei:
 - hosts
 - tasks
 - modules (pl. `ios_command`, `ios_config`)



Cisco Modulok az Ansible-ben

Támogatott Cisco modulok

- [cisco.aci](#)
- [cisco.asa](#)
- [cisco.dnac](#)
- [cisco.intersight](#)
- [cisco.ios](#)
- [cisco.iosxr](#)
- [cisco.ise](#)
- [cisco.meraki](#)
- [cisco.mso](#)
- [cisco.nxos](#)
- [cisco.ucs](#)
- [community.network.ftd_configuration](#) (CiscoDevnet community)



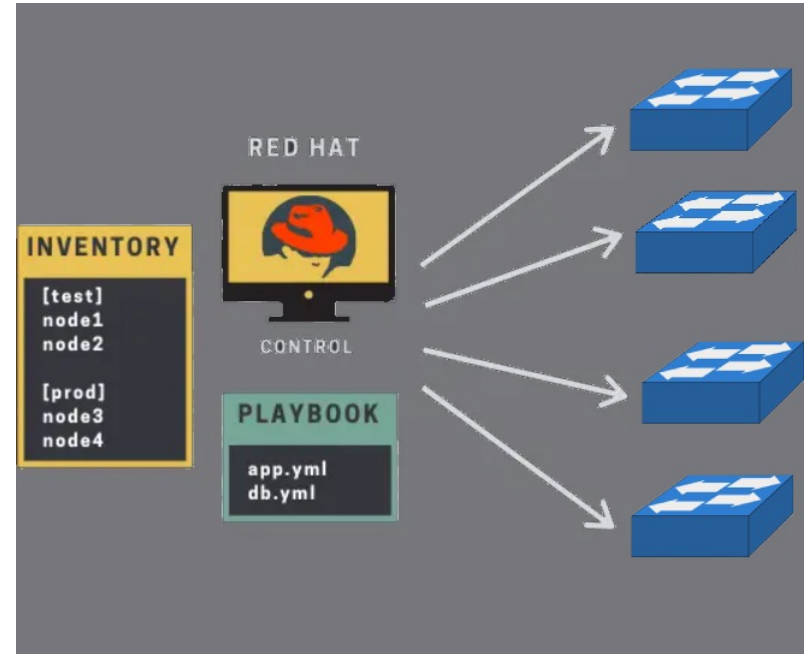
```
---
- name: Switches Config Backup
  hosts: switches
  gather_facts: false

  tasks:
    - name: Backup
      ios_command:
        commands:
          - show run
          - show inventory
          - show rep topology
          - sh spanning-tree sum
          - sh vlan brief
          - sh version
      register: output

    - name: Save Config
      copy:
        content: "{{ output.stdout[0] }}" | "{{ output.stdout[1] }}" | "{{ output.stdout[2] }}" | "{{ output.stdout[3] }}" | "{{ output.stdout[4] }}"
        dest: "/backups/switches/{{ inventory_hostname }}.txt"
```

Mi kell egy sikeres Ansible playbook futtatáshoz?

- Inventory fájl: eszközök listája, amelyeket kezelni szeretnénk.
- SSH kapcsolat, eszközök elérhetősége
- Modulok, pl. `ios_config`, `ios_command`.
- Hitelesítési adatok az eszközökhöz.



Hitelesítési adatok kezelése

- **Ansible Vault:** Ez egy beépített eszköz az érzékeny adatok titkosítására. A Vault használatával elrejthetjük a jelszavakat és más titkos adatokat a Playbook-fájlokban.
 - AES256 titkosítás
 - Könnyű integráció a playbook-ban
 - Több kulcs és jelszó kezelése
 - GDPR és ISO kompatilis

Copy

```
$ cat encrypt_me.txt
```

Output

```
$ANSIBLE_VAULT;1.1;AES256  
66633936653834616130346436353865303665396430383430353366616263323161393639393136  
3737316539353434666438373035653132383434303338640a396635313062386464306132313834  
34313336313338623537333332356231386438666565616537616538653465333431306638643961  
3636663633363562320a613661313966376361396336383864656632376134353039663662666437  
39393639343966363565636161316339643033393132626639303332373339376664
```

```
bash
```

```
ansible-vault encrypt group_vars/all.yml
```

Hibaelhárítás és naplózás (Logging és Debugging)

- **Ansible naplók:** Ansible futtatásának részletes naplózását (-vvv opció a futtatás során).
- **Dry run (check mode):** Ansible rendelkezik egy “ellenőrzési móddal” (check mode), amely során a változtatások végrehajtása nélkül csak ellenőrzi, hogy mi fog történni. A **cisco.ios.ios_config** modul például **csak kiértékeli** a konfigurációs parancsokat. Az Ansible ellenőrzi, hogy a megadott konfigurációk eltérnek-e a jelenlegi állapottól, de nem ír bele a konfigurációs fájlokba, és **nem hajt végre tényleges konfigurációs parancsokat**.

```
bash
```

```
ansible-playbook playbook.yml --check
```

DEMO TIME: IOS frissítés Ansible segítségével

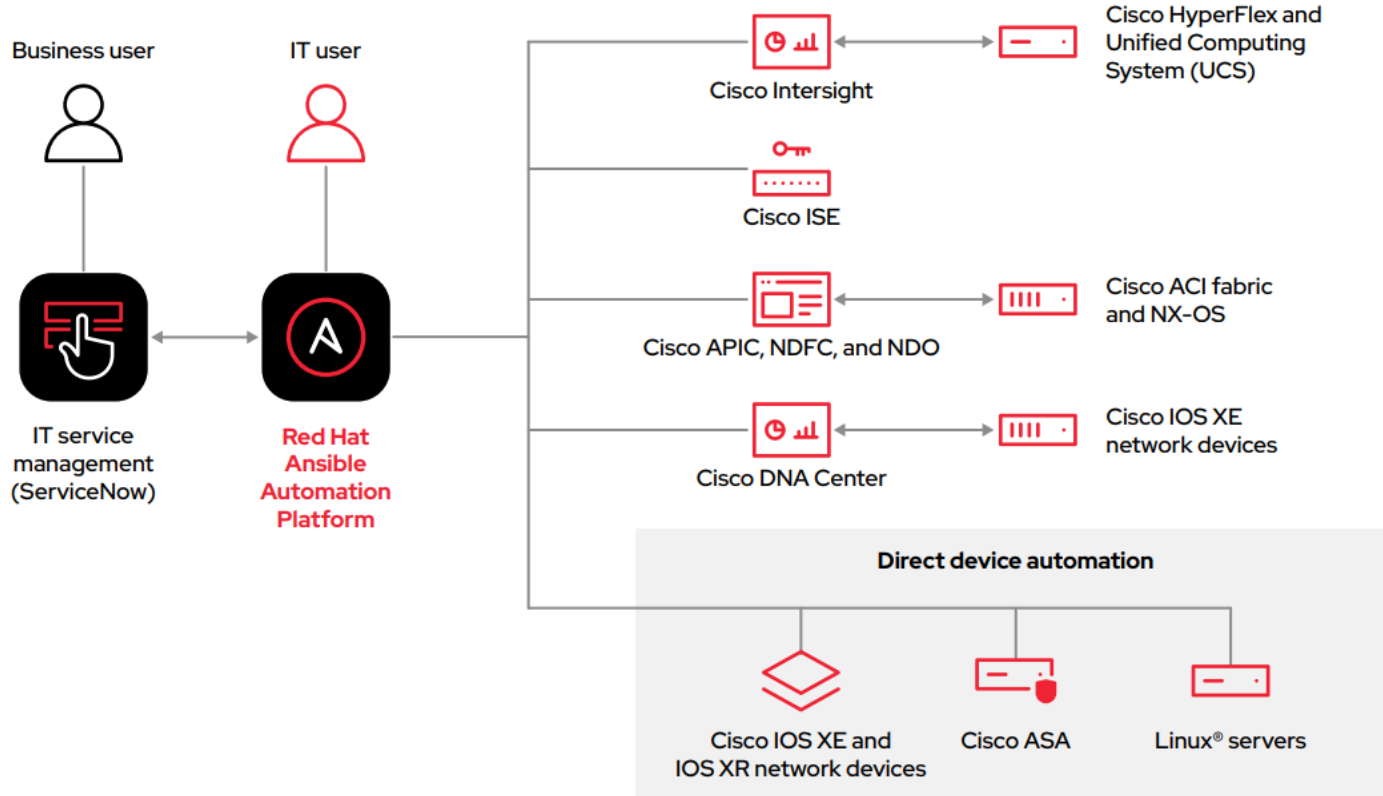
DEMO TIME:

Konfiguráció mentés és show parancsok futtatása

DEMO TIME: Normalizáló konfiguráció



RedHat Ansible Automation Platform

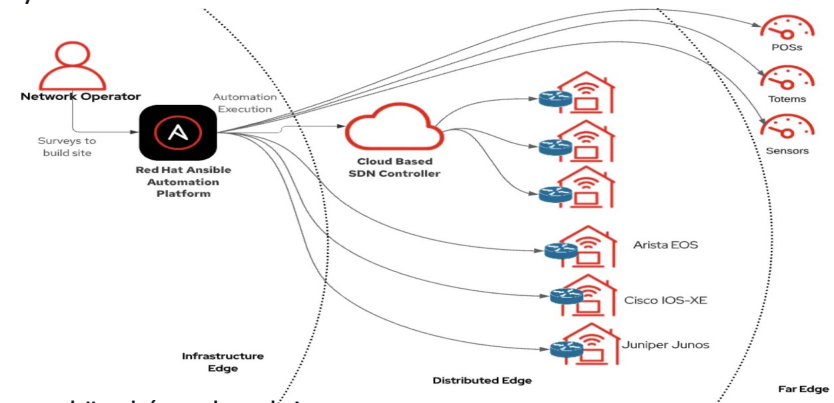


Főbb különbségek

	Ansible-playbook	Ansible Automation Platform
Használati szint	Parancssori eszköz, egyszerű futtatás Playbook-okhoz	Teljeskörű automatizálási platform, GUI-val
Menedzsment	Nincs központosított menedzsment	Központi irányítás, felhasználók kezelése, RBAC
Ütemezés	Manuális futtatás	Ütemezhető automatizációs feladatok
Felhasználói felület	Parancssor (CLI)	Grafikus felhasználói felület (GUI)
Skálázhatóság	Kisebb környezetek, egyedi futtatások	Nagy környezetek skálázása, auditálása
Integrációk	Korlátozott, manuálisan scriptelhető	API és natív integrációk más eszközökkel
Futtatás ellenőrzése	Nincs beépített monitorozás	Monitoring, jelentések, analitika

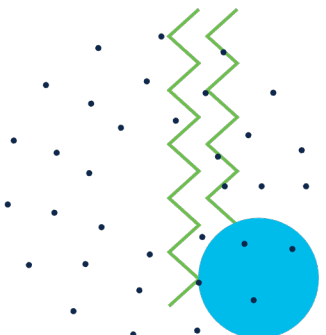
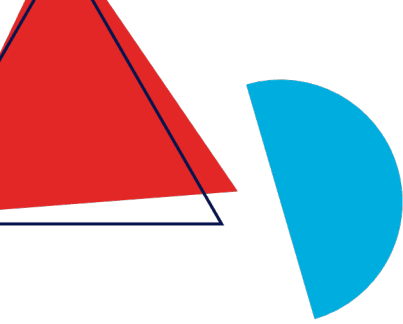
Ansible Automation Platform előnyei

- **Átfogó automatizálási megoldás**
 - Teljes körű platform az Ansible automatizálás kezelésére, fejlesztésére és bevezetésére.
 - Központosított irányítás a különböző Playbook-ok, feladatok és környezetek felett.
- **Főbb komponensek:**
 - **Automation Controller (korábban Tower):** Vizualizált felügyelet, jogkezelés, ütemezés és API hozzáférés.
 - **Automation Hub:** Megosztott és újrahasznosítható automatizációs tartalmak letöltése és kezelése.
 - **Analytics:** Automatizációs feladatok monitorozása, jelentéskészítés és teljesítmény optimalizálás.
- **Előnyök:**
 - **Skálázhatóság:** Nagyobb környezetekben is alkalmazható, több ezer eszközt képes kezelni.
 - **RBAC (Role-Based Access Control):** Felhasználói jogok finomhangolása, biztonságos hozzáférés.
 - **Egyszerű integráció:** Könnyen összekapcsolható más DevOps eszközökkel (pl. Jenkins, Git, CI/CD).
- **Célzott automatizálás minden szinten:**
 - IT adminisztráció, hálózatok, alkalmazások és biztonság automatizálása egy platformon keresztül.





Köszönöm a figyelmet!



Techtorial

CISCO Újra együtt

