

ANSIBLE

by Red Hat®

自動化無所不在

Ansible 技術概觀與介紹

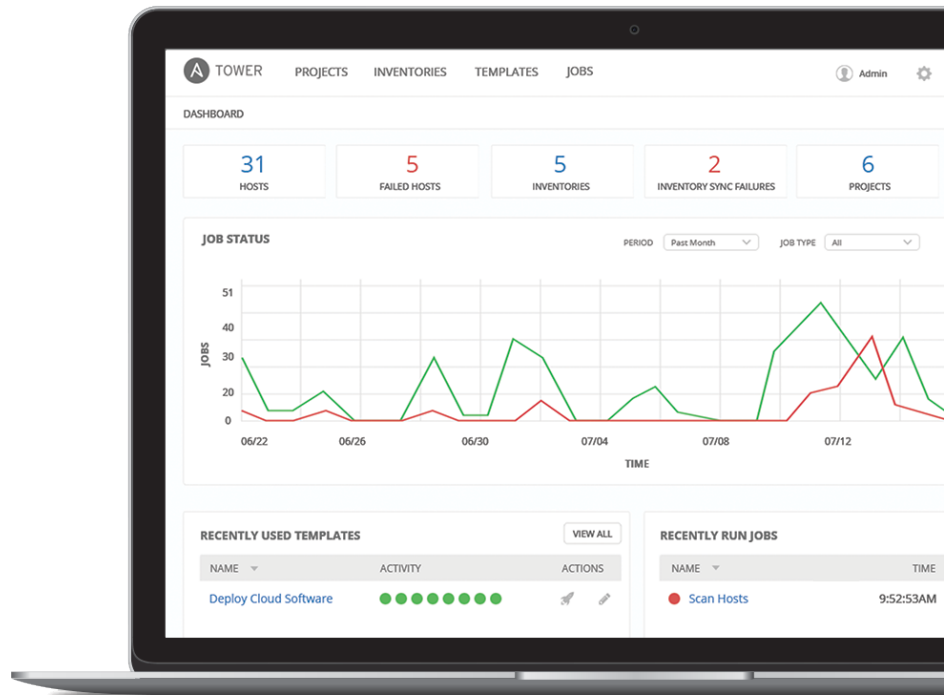
Joe Yu

Senior Solution Architect, Red Hat

沒有一件 **IT** 日常工作 應該手動來完成

Ansible 是由 Red Hat 支援的開放原始碼社群，提供簡單自動化語言，用 Ansible Playbooks 以描述 IT 應用環境

Ansible Tower 是企業管理框架，透過 **UI** 與 **RESTful API** 來控制管理並安全強化 Ansible 的自動化



v1 - Set config file to use on boot

1. Write multiple configuration files
 - For each environment/region
2. Inspect metadata on boot and use the matching config file



v1 - Set config file to use on boot

1. Write multiple configuration files
 - For each environment/region
2. Inspect metadata on boot and use the matching config file

21,000+
Stars on GitHub

1300+
Ansible modules

400,000+
Downloads a month

ANSIBLE 的特色

跨平台

Agentless，支援 Linux, Unix 與 Windows，跨越實體、虛擬、雲端環境以及網路設備。

容易閱讀

可以妥善描述並記錄應用程式環境的每一組成

完美描述應用程式

Playbooks 記錄每一個改變，確認每人的認知都在同一水準上

版本控制

Playbooks 是文本檔案，就像是程式原始碼一般，可利用既有的版本控制軟體進行管理

動態資產

動態獲取伺服器 /IP 資訊，無論其運行位置為何

與其他編配方案協同運作

適應不同的環境與需求，整合其他編配方案，共同完成自動化任務



SIMPLE

- 適合人類閱讀自動化語言
- 不需特殊程式開發技能
- 依序執行工作 (Tasks)

提昇生產效率



POWERFUL

- 應用程式佈署
 - 設定管理
 - 流程與編配
 - 網路自動化
- 編配應用程式生命週期



AGENTLESS

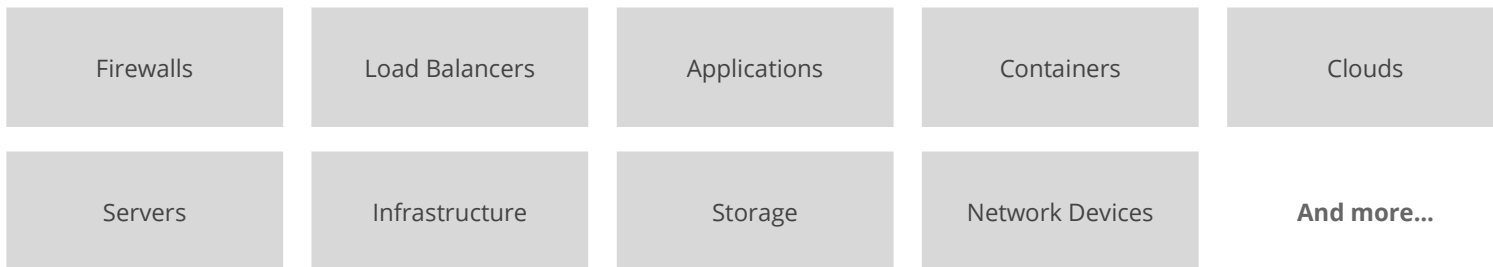
- Agentless 架構
 - 使用既有的 SSH & WinRM
 - 無需考慮 Agent 資安與更新
- 更有效率 & 更加安全

自動化整個 IT 的部署與管理

作這些事 ...



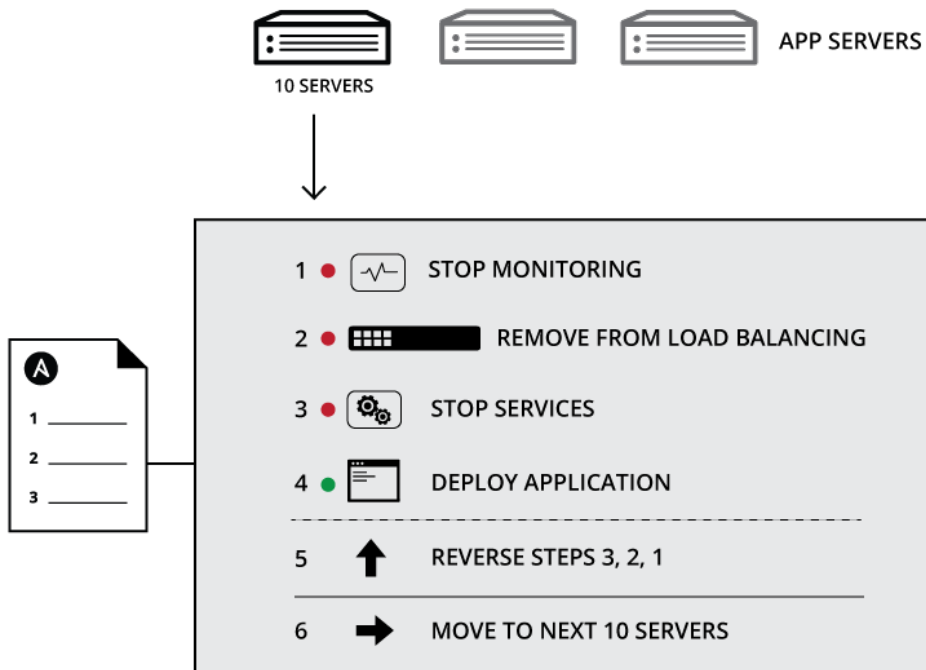
對這些對象 ...

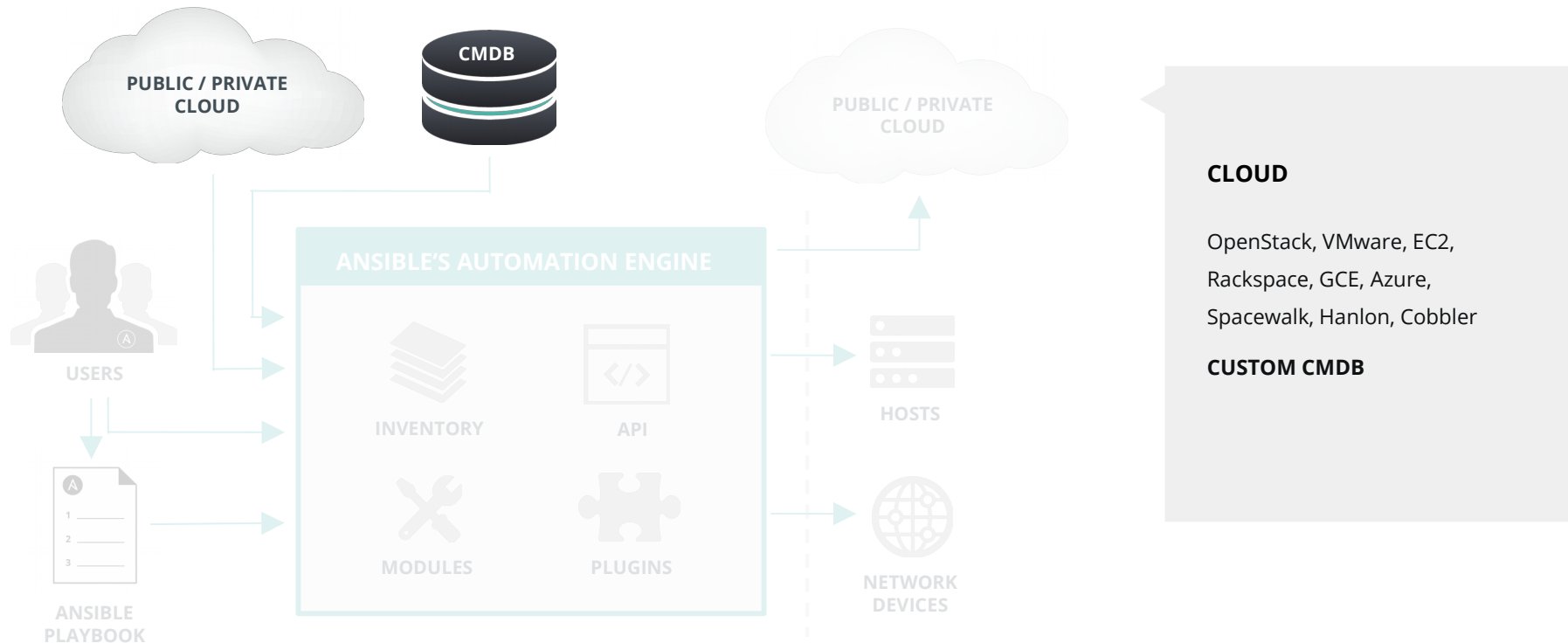


應用程式與系統不僅僅只是一堆設定檔的組合
通常還牽涉一連串的調整優化與工作項目的優先順序，最後才能產出所需的應用服務與環境

Ansible 可以幫你完成：

- 佈署
- 應用佈版
- 設定管理
- 多層編配





ANSIBLE

DEVOPS 的語言



Ansible 是第一個語言能提供跨越 IT 組織的讀寫能力

Ansible 是唯一一個自動化引擎可以自動化整個應用程式生命週與持續交付控制

```
---
- name: install and start apache
  hosts: web
  become: yes
  vars:
    http_port: 80

  tasks:
  - name: httpd package is present
    yum:
      name: httpd
      state: latest

  - name: latest index.html file is present
    copy:
      src: files/index.html
      dest: /var/www/html/

  - name: httpd is started
    service:
      name: httpd
      state: started
```

CLOUD

AWS
Azure
CenturyLink
CloudScale
Digital Ocean
Docker
Google
Linode
OpenStack
Rackspace
And more...

VIRT AND CONTAINER

Docker
VMware
RHEV
OpenStack
OpenShift
Atomic
CloudStack
And more...

WINDOWS

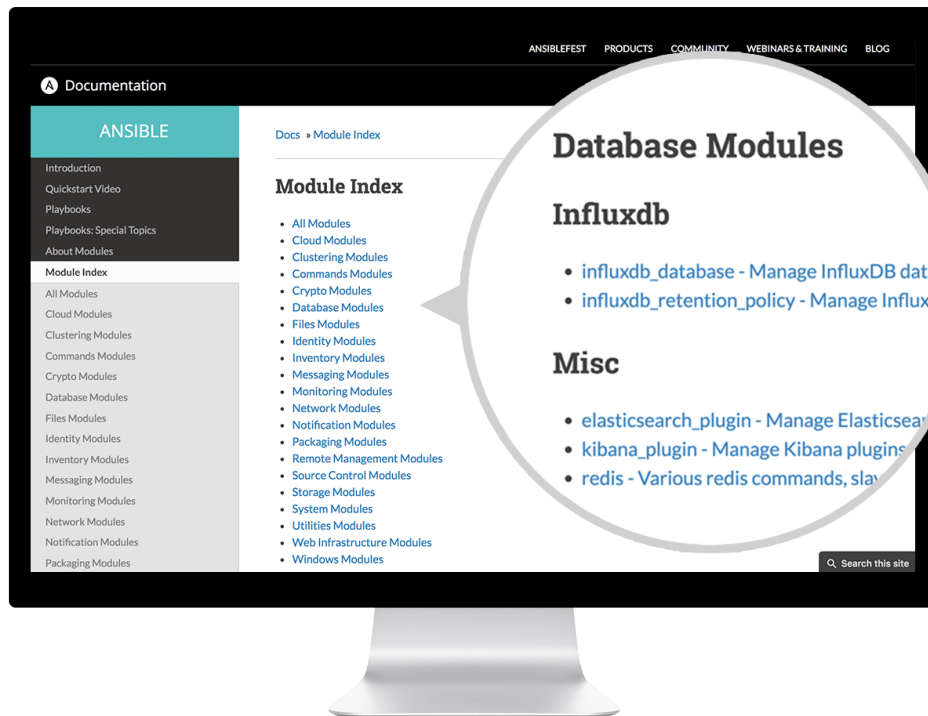
ACLs
Files
Commands
Packages
IIS
Regedits
Shell
Shares
Services
Configs
Users
Domains
And more...

NETWORK

Arista
A10
Cumulus
Big Switch
Cisco
Cumulus
Dell
F5
Juniper
Palo Alto
OpenSwitch
And more...

NOTIFY

HipChat
IRC
Jabber
Email
RocketChat
Sendgrid
Slack
Twilio
And more...



LAMP + HAPROXY + NAGIOS

github.com/ansible/ansible-examples/tree/master/lamp_haproxy

WINDOWS

github.com/ansible/ansible-examples/tree/master/windows

SECURITY COMPLIANCE

github.com/ansible/ansible-lockdown

NETWORK

github.com/privateip/network-demo

MORE...

galaxy.ansible.com

github.com/ansible/ansible-examples



**ANSIBLE
TOWER**
by Red Hat®

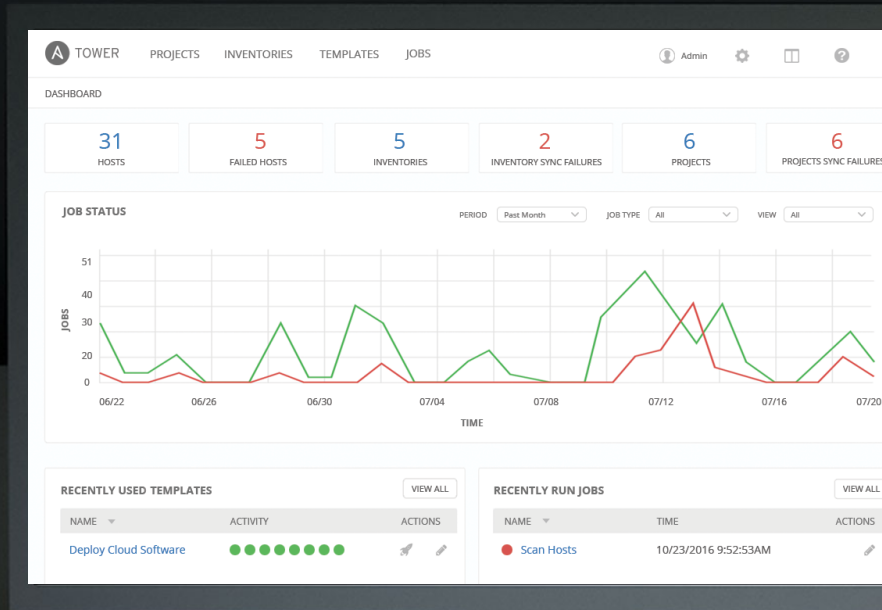
團隊合作實現自動化

Ansible Tower 技術介紹與概觀

什麼是 ANSIBLE TOWER?

Ansible Tower 是企業管理框架，透過 UI 與 RESTful API 來控制管理並安全強化 Ansible 的自動化 A

- 角色為基礎存取控制 (RBAC)
- 實現一鍵式部署或工作
- 所有自動化工作歷程都會被集中的記錄





TOWER EMPOWERS TEAMS TO AUTOMATE

CONTROL

KNOWLEDGE

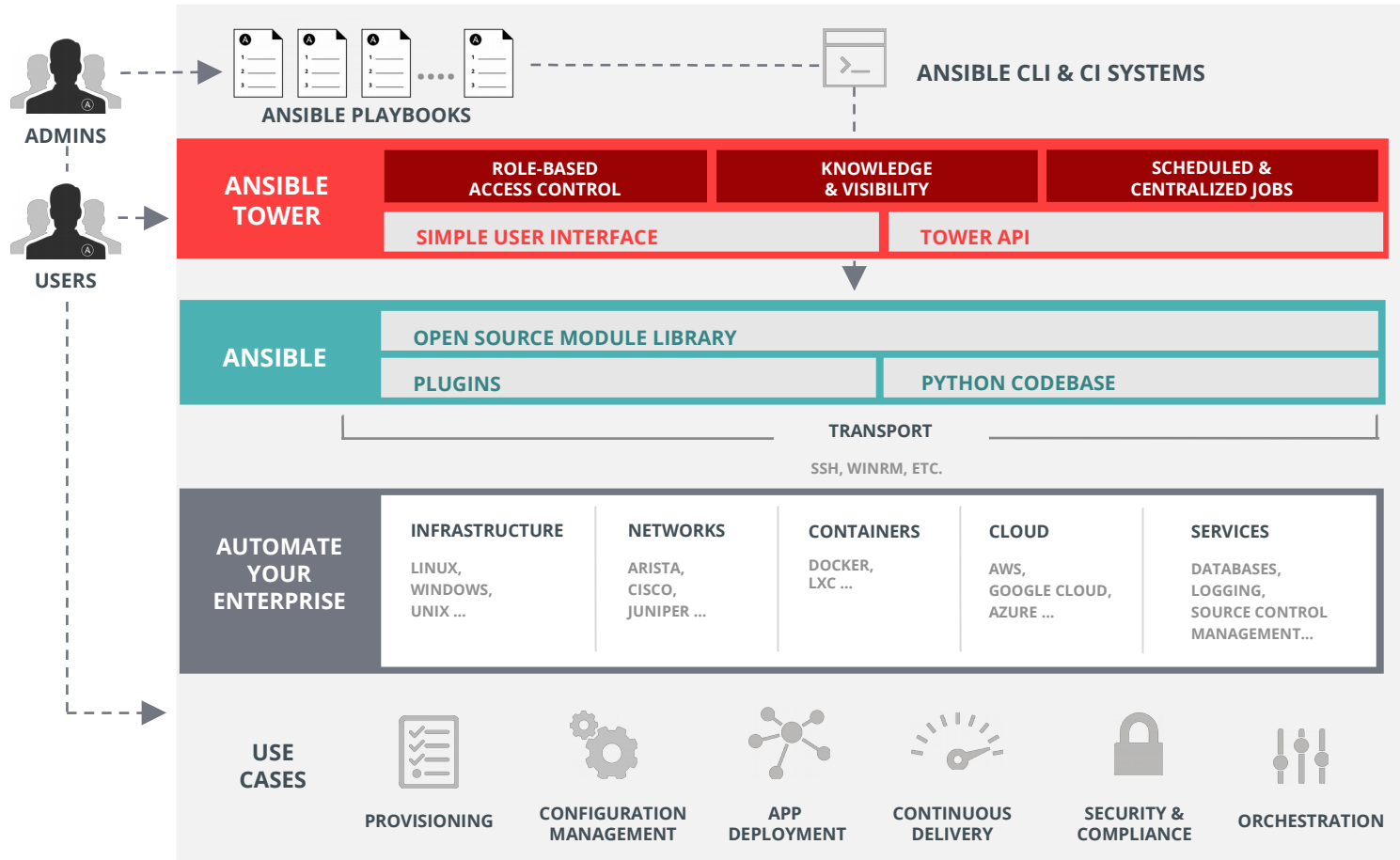
DELEGATION

SIMPLE

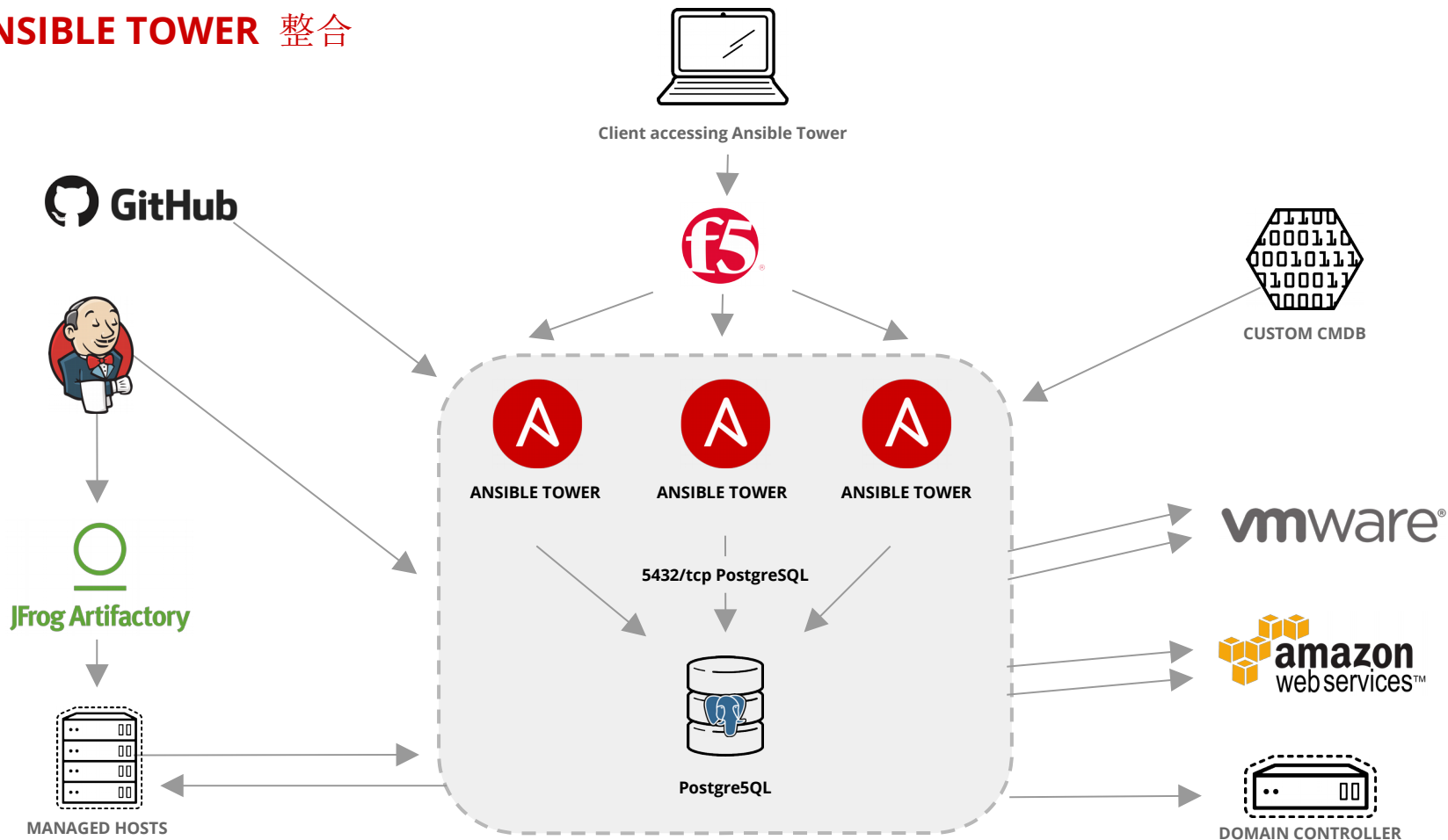
POWERFUL

AGENTLESS

AT ANSIBLE'S CORE IS AN **OPEN-SOURCE** AUTOMATION ENGINE



ANSIBLE TOWER 整合



TOWER 新功能

WORKFLOWS

將工作範本串接載一起，形成全新的工作流程
無須重新編寫 Playbook

SCALE-OUT CLUSTERING

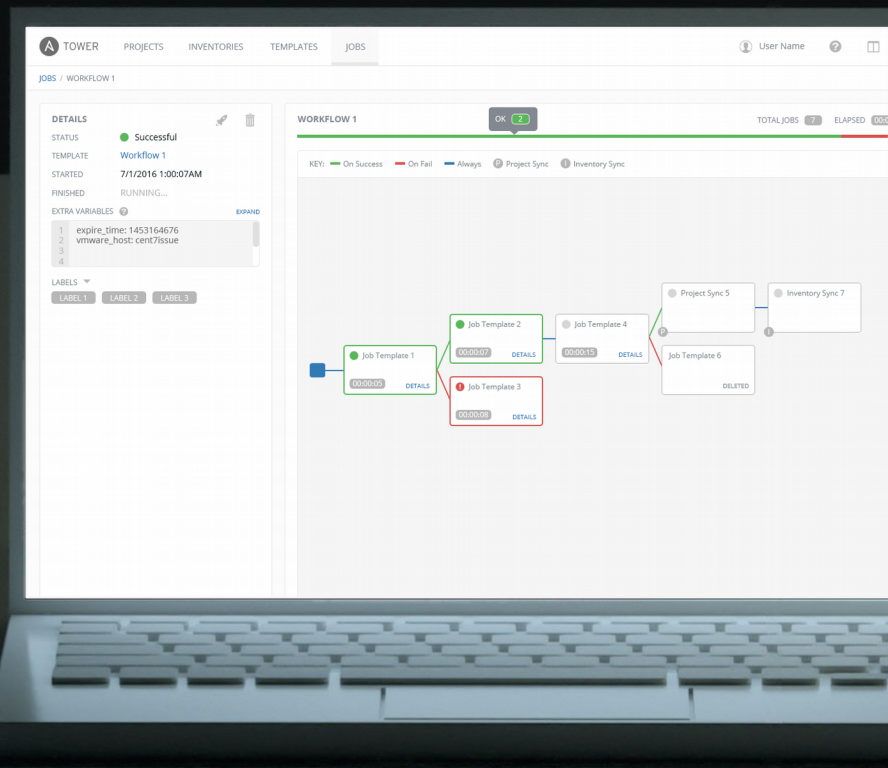
快速新增分工資源與 HA 能力

DETAILED JOB OUTPUT

指尖上的資訊

LOG AGGREGATION AND DELIVERY

自動推送工作產出資料到 Splunk, Sumologic 或其他資訊處理服務



JOB STATUS UPDATE

直覺的 NOC(Notifiacion Of Chnag) 風格
自動化儀表板，顯示 Ansible 環境的每一個
工作步驟

The screenshot displays the Ansible Tower interface for a job titled "REMOVE VMWARE HOST". The job is in a "Successful" state, having started at 1:00:07 AM and finished at 1:00:12 AM. The job was launched by a user and is associated with the "Update License Server" template. The project is "License" and the revision is "00000000". The playbook used is "store.yml" and the credential is "License Server Deployment". The limit is set to "Store" and the verbosity is "Update License Server".

The "EXTRA VARIABLES" section shows the following variables:

```
1 expire_time: 1453164676
2 vmware_host: cen7issue
3
4
```

The "TASKS" section shows the execution progress of the job:

```
1 PLAY [Remove VMWare Host] ..... 00:00:05
2
3 GATHERING FACTS ..... 00:00:01
4
5
6
7
8
9
10
11
12
13
14
15
16 TASK: [ansiblelicense | install required packages via yum] ..... 00:00:01
17 ok: [74.207.226.226]
18 ok: [74.207.226.226]
19
20
21
22
23
24
25
26
27
28
29 TASK: [ansiblelicense | update setuptools] ..... 00:00:01
30 ok: [74.207.226.226]
31 ok: [74.207.226.226]
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
```


ACTIVITY STREAM | ALL ACTIVITY

REFRESH All Activity

INITIATED BY SEARCH

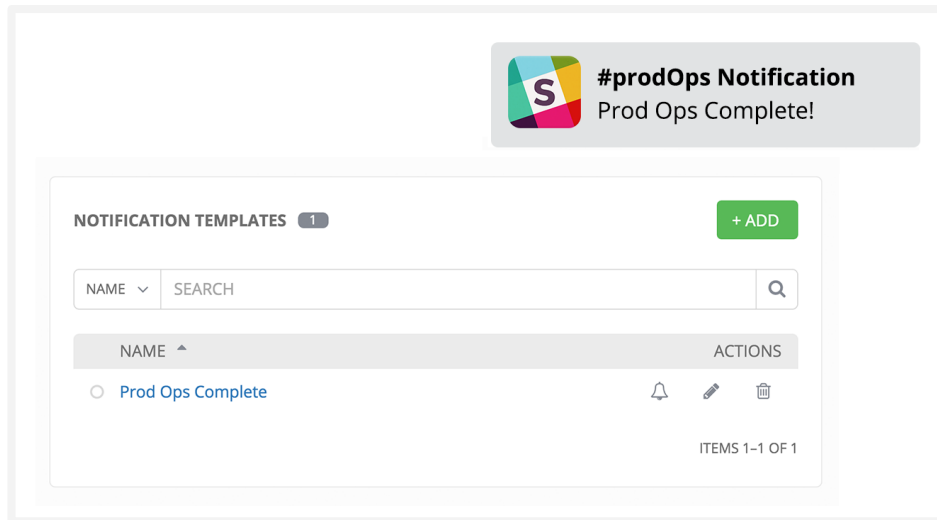
TIME	INITIATED BY	EVENT	ACTIONS
10/3/2016 5:00:52 PM	admin	created schedule Daily remediation	
10/3/2016 4:51:45 PM	admin	deleted schedule Hourly scan	
10/3/2016 4:51:13 PM	admin	created schedule Hourly scan	

ACTIVITY STREAM

安全地存放每一個執行工作，並與允許事後檢視或透過 Tower 的 API 匯出結果

INTEGRATED NOTIFICATIONS




透過整合通知即時通知您自動化狀態
連結 Slack, Hipchat, SMS, email 或其他服務進行



#prodOps Notification
Prod Ops Complete!

NOTIFICATION TEMPLATES 1 + ADD

NAME ▾ SEARCH 🔍

NAME ▲	ACTIONS
<input type="radio"/> Prod Ops Complete	  

ITEMS 1-1 OF 1

The screenshot shows the 'JOB TEMPLATE SCHEDULES / JOB TEMPLATE SCHEDULES.EDIT' page in Ansible Tower. The main configuration area is titled 'DAILY REMEDIATION'. It includes the following fields:

- * NAME:** Daily remediation
- * START DATE (MM/DD/YYYY):** 10/03/2016
- * START TIME (HH24:MM:SS):** 01:23:45
- * LOCAL TIME ZONE:** America/New_York
- * REPEAT FREQUENCY:** Day

Below these are the 'FREQUENCY DETAILS' section:

- * EVERY:** 1 DAYS
- * END:** Never

The 'SCHEDULE DESCRIPTION' section shows:

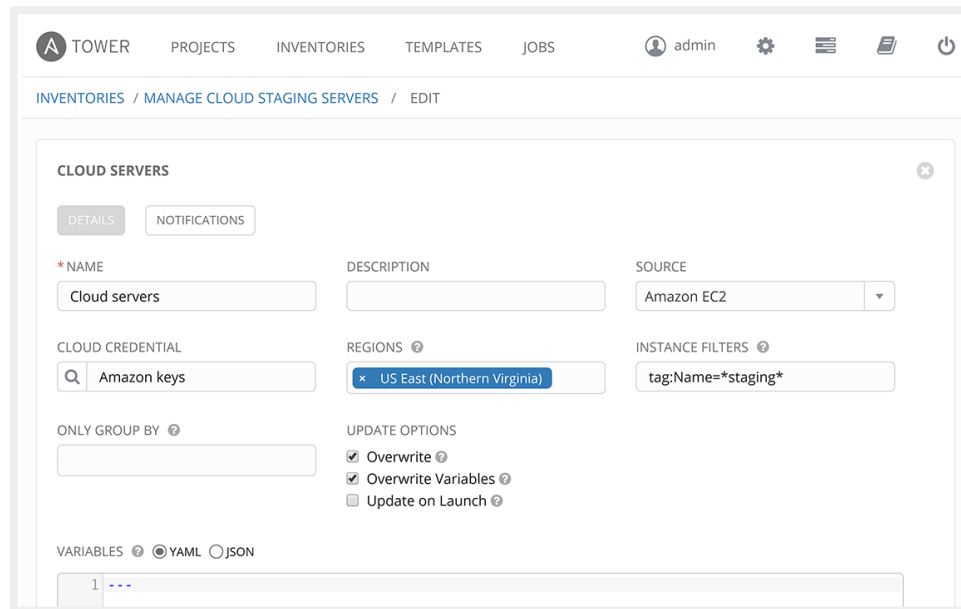
- every day
- OCCURRENCES (Limited to first 10) DATE FORMAT LOCAL TIME UTC
- 10/03/2016 01:23:45 EDT

SCHEDULE JOBS

允許現在、未來或重複地執行任何工作

MANAGE AND TRACK YOUR INVENTORY

Tower 的 inventory syncing 與 provisioning callbacks 機制，允許主機即時提交設定需求，以完成自動擴充



LAUNCH JOB | DEPLOY SOFTWARE ✕

INVENTORY CREDENTIAL SURVEY

* ENTER NUMBER OF SERVICE INSTANCES.

* PLEASE SELECT THE SERVICE OWNER.

* ENTER PASSWORD FOR DEPLOYED CERTIFICATE.

SHOW

INVENTORY CREDENTIAL
Cloud staging servers Staging ssh key

CANCEL LAUNCH

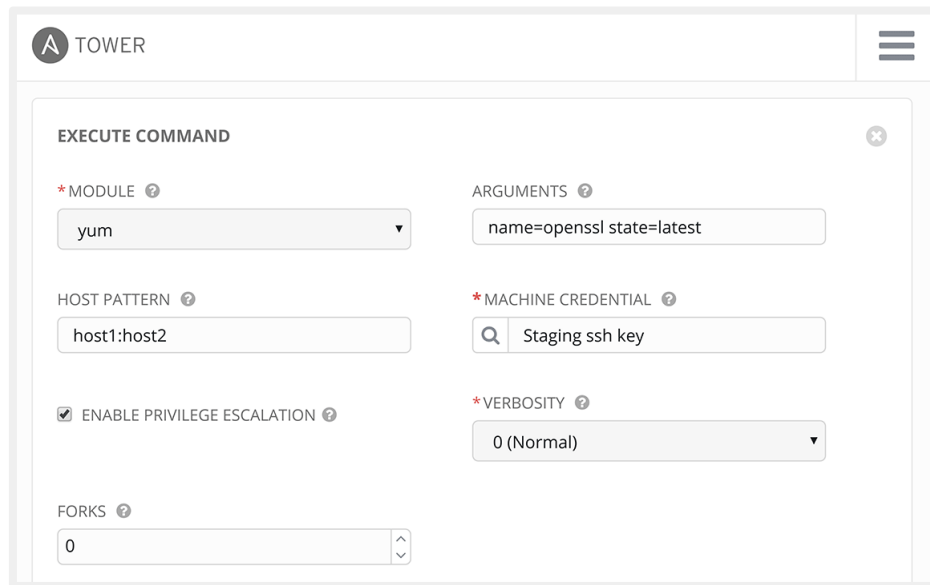
SELF-SERVICE IT

Tower 讓你可以用簡單地點擊，就能執行 Playbooks 當執行時，還能夠彈出選項視窗，以供進行改變或選擇，例如變數、帳密等

REMOTE COMMAND EXECUTION

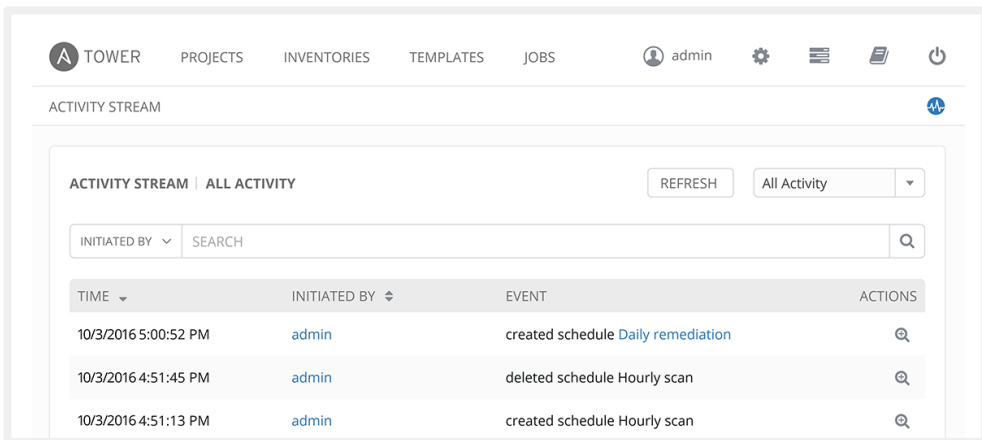
透過 Tower 的 remote command execution，在任何主機上執行簡單工作，例如：

新增用戶或群組，重設密碼，重啟異常服務或馬上修補嚴重的資安套件等



The screenshot shows the 'EXECUTE COMMAND' configuration page in Ansible Tower. The interface includes the following fields and options:

- *MODULE**: A dropdown menu with 'yum' selected.
- ARGUMENTS**: A text input field containing 'name=openssl state=latest'.
- HOST PATTERN**: A text input field containing 'host1:host2'.
- *MACHINE CREDENTIAL**: A search field with 'Staging ssh key' selected.
- ENABLE PRIVILEGE ESCALATION**: A checked checkbox.
- *VERBOSITY**: A dropdown menu with '0 (Normal)' selected.
- FORKS**: A text input field with '0' and up/down arrow controls.



The screenshot displays the Ansible Tower web interface. At the top, there is a navigation bar with the 'TOWER' logo and menu items for 'PROJECTS', 'INVENTORIES', 'TEMPLATES', and 'JOBS'. The user 'admin' is logged in. Below the navigation bar, the 'ACTIVITY STREAM' section is visible, showing a list of events under the heading 'ACTIVITY STREAM | ALL ACTIVITY'. The interface includes a 'REFRESH' button and a dropdown menu currently set to 'All Activity'. A search bar is present with the text 'INITIATED BY' and a search icon. The main content is a table with the following data:

TIME	INITIATED BY	EVENT	ACTIONS
10/3/2016 5:00:52 PM	admin	created schedule Daily remediation	
10/3/2016 4:51:45 PM	admin	deleted schedule Hourly scan	
10/3/2016 4:51:13 PM	admin	created schedule Hourly scan	

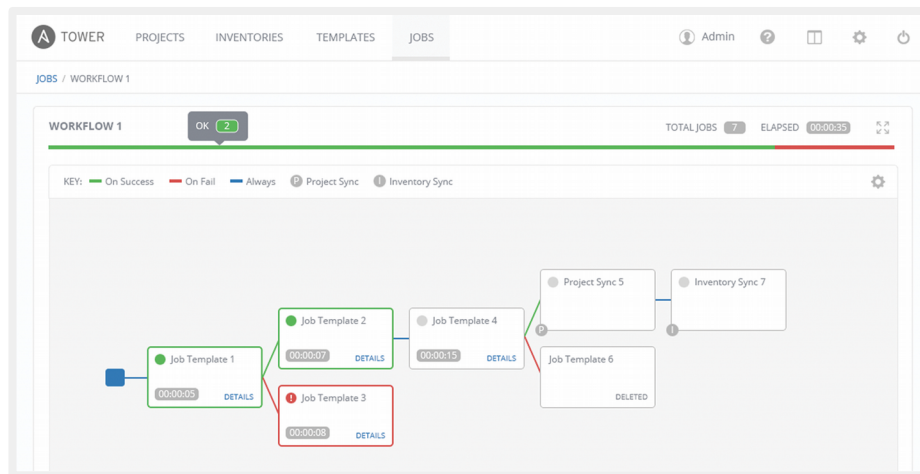
EXTERNAL LOGGING

連接 Tower 到外部日誌與分析服務，以便進行環境自動化與相關事件的進一步分析

MULTI-PLAYBOOK WORKFLOWS

Tower 的多 Playbook 工作流程可以串接任意數量的 Playbook，以建立一個全新的工作流程

依據上一個 Playbook 執行的成功或失敗，可以執行不同的工作



全球 **ANSIBLE TOWER** 的客戶（部份選擇）



自動化 = 加速



“If you do things in **complex ways**, they become very **difficult to maintain**, and you end up **paying a lot more for operations** later.”

J.CREW

“With Ansible Tower, **we just click a button and deploy to production in 5 minutes**. It used to take us 5 hours with 6 people sitting in a room, making sure we didn't do anything wrong (and we usually still had errors). We now deploy to production every other day instead of every 2 weeks, and nobody has to be up at 4am making sure it was done right.”



“**Everyone in this room** needs to **around Ansible automation**. **All Starbucks network changes** will be scheduled **using Ansible Tower** by the end of 2017.”



存放超過 **10,000 ROLES**

重複使用的 Roles 與應用容器可以讓你作的更快、更多
為 Ansible CLI 與 Tower 所設計

galaxy.ansible.com

如何開始?

如果你已經有用 Ansible

你可以試用 *Tower*: ansible.com/tower-trial

如果你想學 Ansible

你可以由 *Ansible get started* 開始: ansible.com/get-started

還想要學更多?

到右方連結尋找影片, webinars, 客戶案例, 白皮書: ansible.com/resources

或是

AUTOMATION WITH ANSIBLE I

(DO407)

Designed for system administrators who are intending to use Ansible for automation, configuration, and management. Learn how to install and configure Ansible, create and run playbooks to configure systems, and learn to manage inventories.

AUTOMATION WITH ANSIBLE II: RED HAT ANSIBLE TOWER

(DO409)

Designed for IT professionals who use Ansible and need to centrally manage their Ansible projects in a way that scales to large teams and complex enterprise installations using Red Hat® Ansible® Tower.

謝謝！