

Lectie: Automatizarea cloud, managementul si orchestrarea cloud

Introducere: de ce sunt necesare automatizarea, gestionarea și orchestrarea

Calculul cloud modern se dezvoltă cu o viteză incredibilă. Organizațiile creează infrastructuri complexe, formate din sute și chiar mii de mașini virtuale, containere, rețele, sisteme de stocare și servicii. În primele etape de dezvoltare a cloud-ului, majoritatea sarcinilor erau efectuate manual de ingineri: aceștia porneau serverele, instalau aplicațiile și configurau rețelele. Însă, pe măsură ce amploarea a crescut, a devenit evident că gestionarea manuală nu mai face față acestei complexități.

Viteza afacerilor necesită automatizare. Utilizatorii se așteaptă ca resursele să fie alocate instantaneu, aplicațiile să fie implementate în câteva minute, iar infrastructura să se adapteze singură la încărcare. În același timp, companiile trebuie să păstreze controlul: cine utilizează resursele, cât costă acest lucru, dacă este în conformitate cu politica de securitate. Aici intră în joc trei concepte interconectate: **automatizarea cloud**, **managementul cloud** și **orchestrația cloud**.

Aceste concepte formează baza unei infrastructuri cloud mature. Automatizarea face posibilă executarea acțiunilor fără intervenția umană. Managementul asigură controlul, politica și transparența. Orchestrația combină toate acestea în procese coordonate, astfel încât sistemul să funcționeze ca un organism unic.

I. Automatizarea cloud: transformarea infrastructurii în cod

Automatizarea este inima cloud-ului modern. Ideea sa principală este că infrastructura nu trebuie gestionată manual, ci prin programe. Noi descriem cum ar trebui să fie starea sistemului, iar programele îl aduc singure în această stare.

Când vorbim despre „automatizarea cloud”, ne referim la crearea de mecanisme care îndeplinesc sarcini tipice fără intervenția operatorului: implementarea mașinilor virtuale, configurarea rețelelor, instalarea actualizărilor, copierea de rezervă, monitorizarea, scalarea și multe altele.

În trecut, astfel de operațiuni durau ore, uneori zile. Astăzi, ele sunt efectuate în câteva minute. Automatizarea permite accelerarea semnificativă a proceselor, creșterea fiabilității și previzibilității infrastructurii. Dar cel mai important lucru este că eliberează specialiștii de operațiunile de rutină, permițându-le să se concentreze pe arhitectură, securitate și optimizare.

Principiile automatizării

La baza automatizării stă ideea de **Infrastructure as Code**, sau infrastructura ca cod. Aceasta este o abordare în care toate elementele infrastructurii — rețele, servere, baze de date, servicii de stocare — sunt descrise sub formă de cod. Acest cod este stocat într-un sistem de control al versiunilor și poate fi testat, analizat, reprodus și modificat la fel ca codul programului.

Această abordare asigură repetabilitatea. Același scenariu poate fi aplicat în medii diferite, iar rezultatul va fi identic. Acest lucru este deosebit de important pentru organizațiile mari, unde configurații identice trebuie utilizate în medii de testare, pre-lansare și productive.

Un alt principiu important este ideopotențialitatea. Aceasta înseamnă că scenariul poate fi rulat de câte ori se dorește, iar dacă sistemul se află deja în starea dorită, nimic nu se va schimba. Acest lucru face automatizarea fiabilă și sigură.

Automatizarea se bazează adesea pe modele declarative. Nu descriem pașii de execuție, ci starea dorită: de exemplu, „sunt necesare trei mașini virtuale în această rețea, cu astfel de parametri”, iar sistemul determină singur cum să realizeze acest lucru.

Rolul automatizării în operațiunile cloud

Automatizarea devine o verigă de legătură între infrastructură și servicii. Ea asigură coerența între medii, accelerează implementarea, minimizează erorile și îmbunătățește scalabilitatea.

Ea permite infrastructurii să fie dinamică: să reacționeze automat la creșterea încărcării, să se recupereze după defecțiuni, să activeze și să dezactiveze resursele în funcție de ora din zi sau de trafic.

Furnizorii moderni de servicii cloud își construiesc platformele în jurul automatizării. Amazon Web Services oferă CloudFormation, Microsoft — Azure Resource Manager, Google — Deployment Manager. Soluții independente precum Terraform sau Ansible permit gestionarea simultană a mai multor clouduri dintr-un singur punct. Toate acestea fac din automatizare un instrument universal de gestionare a infrastructurii.

Probleme și provocări ale automatizării

Dar automatizarea nu înseamnă doar scrierea de scripturi. Ea necesită cultură, disciplină și arhitectură. Adesea, organizațiile se confruntă cu problema devierii configurației — așa-numita *configuration drift*, când starea reală a resurselor începe treptat să difere de cea descrisă în cod.

De asemenea, apare problema securității. Scripturile de automatizare conțin adesea parole, tokenuri și chei, de aceea este important să păstrați secretele în mod corespunzător, să criptați datele și să limitați accesul.

În plus, automatizarea necesită testare. Trebuie să fii sigur că scenariul nu va duce la o defecțiune în mediul productiv. Acest lucru se realizează prin utilizarea mediilor de testare preliminară, a conductelor CI/CD și a controlului versiunilor.

În ciuda tuturor complexităților, automatizarea este baza. Fără ea, este imposibil să construiești un management sau o orchestrare eficientă.

Exemple de automatizare (automatizare cloud)

Exemplul 1: Scalarea automată a aplicației web

Să presupunem că aveți o aplicație web, la care în timpul orelor de vârf încărcarea crește, iar noaptea scade. Cu ajutorul automatizării (de exemplu, prin Terraform + autoscaling de grup + scripturi de monitorizare), configurați ca, la atingerea pragului CPU sau a numărului de solicitări, să se adauge automat o nouă mașină virtuală sau un container, iar la scăderea încărcării, să se elimine exemplarul în plus. Acest lucru permite economisirea resurselor și menținerea performanței.

Exemplul 2: Aplicarea automată a actualizărilor și patch-urilor

Organizația utilizează Ansible sau Puppet pentru a verifica și actualiza automat sistemele de operare și aplicațiile din mașinile virtuale sau containere. După lansarea unei noi versiuni sau a unui patch de securitate, sistemul implementează singur actualizările pe toate nodurile, verifică succesul aplicării și, în caz de eșec, revine la modificările anterioare.

Exemplul 3: Provisioning (implementarea) infrastructurii ca cod

Descrii infrastructura dorită (rețele, mașini virtuale, stocare, firewall) în fișierele Terraform sau CloudFormation. La lansarea unui astfel de șablon, sistemul creează automat resursele necesare, configurează rețelele, conectează stocarea etc. Acest lucru permite reproducerea rapidă a întregilor medii (de testare, pre-lansare, productive) cu aceeași configurație.

Exemplul 4: Dezactivarea automată a resurselor „moarte”

În cloud rămân adesea resurse nefolosite (mașini virtuale, discuri, exemplare de baze de date) care continuă să funcționeze și să genereze costuri. Se poate configura o automatizare care verifică în fiecare noapte dacă resursele sunt utilizate și, dacă nu există activitate, le oprește sau le șterge.

II. Managementul cloud: gestionare, control și politici

Dacă automatizarea este un mecanism de execuție, atunci managementul cloud este un sistem de control al întregului proces. Acesta răspunde la întrebări precum: cine utilizează resursele, cum sunt utilizate, cât costă și dacă totul respectă regulile stabilite.

În organizațiile mari, cloudul nu este doar o tehnologie, ci un întreg ecosistem. Diferite departamente comandă resurse, dezvoltatorii creează aplicații, analiștii prelucrează date. Fără un management centralizat, apare haosul: resursele sunt consumate fără a fi luate în considerare, securitatea este încălcată, bugetele depășesc limitele.

Managementul cloud face ca acest sistem să fie gestionabil. Acesta combină monitorizarea, contabilitatea, securitatea, facturarea, politicile de acces și automatizarea într-o singură structură.

Ce face managementul cloud

Managementul asigură transparența și controlul. Acesta creează cataloage de servicii din care utilizatorii pot alege configurații gata pregătite — mașini virtuale, baze de date, spații de stocare, rețele. Acesta stabilește cote și limite, astfel încât nimeni să nu consume mai multe resurse decât este permis.

În plus, managementul este responsabil pentru contabilitatea costurilor și raportare. Într-un mediu multi-utilizator, este important să se înțeleagă cine consumă resurse, cât costă acestea și unde se pot face economii. Acest domeniu a primit denumirea de *FinOps* — operațiuni financiare în cloud.

Nu mai puțin importantă este și securitatea. Managementul implementează politici de control al accesului, separarea rolurilor și a competențelor, auditul acțiunilor utilizatorilor. Acesta monitorizează conformitatea cu standardele și cerințele corporative.

Arhitectura de management

Din punct de vedere arhitectural, managementul este împărțit în mai multe niveluri. La nivelul inferior funcționează agenți și servicii care interacționează cu infrastructura cloud și colectează date. Mai sus se află stratul de management — așa-numitul *management plane*, care analizează metricile, aplică politici și inițiază acțiuni.

La nivelul superior se află interfețele de interacțiune — portaluri de autoservire, API, panouri de monitorizare. Prin intermediul acestora, utilizatorii și administratorii gestionează resursele. Managementul este, de asemenea, strâns legat de sistemele ITSM, care asigură suportul ciclului de viață al serviciilor IT.

Toate acestea formează împreună *Cloud Management Platform* — CMP. Astfel de platforme există în diferite forme. De exemplu, ManageIQ deschis, soluțiile VMware vRealize Suite sau Red Hat CloudForms. Sarcina lor este de a oferi un instrument unic pentru gestionarea mediilor hibride și multi-cloud.

Principalele dificultăți

Managementul se confruntă cu o serie de provocări. Principala dintre ele este diversitatea cloud-urilor. Fiecare furnizor utilizează propriile API-uri și propriile formate de date, de aceea nu este ușor să le combinați într-un singur sistem.

A doua problemă este securitatea și echilibrul între controlul centralizat și libertatea utilizatorilor. Dezvoltatorii vor să acționeze rapid, dar administratorii trebuie să asigure respectarea politicilor. Managementul trebuie să fie capabil să combine ambele abordări.

A treia este gestionarea cheltuielilor. Fără management, companiile se confruntă adesea cu situații în care resursele rămân activate după teste sau sunt rezervate fără a fi necesar. Managementul permite identificarea acestor cazuri și economisirea de fonduri.

Astfel, managementul este nivelul de maturitate la care automatizarea devine gestionabilă, controlabilă și sigură.

Exemple de management al cloudului (Cloud Management)

Exemplul 1: Calcularea costurilor și alocarea bugetului (FinOps)

Sistemul de management colectează metrice de utilizare a resurselor diferitelor departamente, calculează cât a „consumat” fiecare proiect în termeni monetari, afișează rapoarte analitice, propune optimizări (de exemplu, reducerea resurselor neutilizate). Acest lucru ajută la controlul cheltuielilor și la facturarea utilizatorilor cloud.

Exemplul 2: Catalog de servicii și self-service

Organizația pune la dispoziția echipelor sale un catalog de imagini preconfigurate ale mașinilor virtuale sau containerelor. Utilizatorul selectează configurația dorită prin portal, iar sistemul o implementează automat. Conducerea se asigură că sunt respectate cotele, politica de securitate și restricțiile bugetare.

Exemplul 3: Politici de securitate și conformitate (guvernanță)

Nivelul de management stabilește politicile: numai discuri criptate, trafic prin VPN, interzicerea accesului public la anumite porturi. La încercarea de a crea o resursă care nu corespunde politicii, sistemul blochează sau corectează configurația. În plus, se efectuează un audit și o verificare a conformității cu standardele (GDPR, HIPAA etc.).

Exemplul 4: Monitorizare, alerte și audit

Sistemul de management agregă metrice și jurnale de la toate componentele cloud, construiește un panou de monitorizare, generează alerte la depășirea pragurilor, ține un jurnal al acțiunilor utilizatorilor (cine a creat, modificat, șters). Acest lucru oferă transparență și control asupra stării întregii infrastructuri cloud.

Exemplul 5: Gestionarea cotelor și a resurselor

Fiecărei divizii (sau proiect) i se alocă o cotă de resurse (numărul de mașini virtuale, volumul de stocare etc.). Conducerea se asigură că acestea nu depășesc cotele și nu permite alocarea de resurse peste limită. Dacă un utilizator încearcă să solicite mai mult, fie primește un refuz, fie se lansează o cerere de aprobare.

III. Orchestrarea cloudului: coordonarea și integritatea proceselor

Următorul nivel după automatizare și management este orchestrația. Dacă automatizarea înseamnă executarea unor acțiuni separate, atunci orchestrația înseamnă gestionarea întregii secvențe a acestor acțiuni, transformarea lor într-un proces integral.

Orchestrația este necesară acolo unde automatizarea simplă nu mai este suficientă. De exemplu, când trebuie să se implementeze o aplicație multistrat. Aici este necesar să se creeze o rețea, să se aloce subrețele, să se lanseze baze de date, să se implementeze un server de aplicații, să se conecteze un echilibrator, să se lanseze teste și abia apoi să se transfere controlul utilizatorului. Toate acestea trebuie efectuate într-o anumită secvență și cu respectarea dependențelor.

Orchestrația îndeplinește rolul unui dirijor care conduce un întreg orchestră de acțiuni automatizate.

Cum funcționează orchestrația

În centrul sistemului de orchestrare se află un motor care interpretează scenarii sau șabloane. Aceste scenarii descriu secvența operațiunilor, condițiile, dependențele și acțiunile în caz de erori.

Orchestratorul interacționează cu o mulțime de sisteme prin API, apelează scripturi de automatizare, urmărește starea de execuție și, dacă este necesar, revine la modificările anterioare. Toate acestea se întâmplă automat, fără intervenția umană.

De obicei, orchestrația se bazează pe limbaje declarative sau standarde. Unul dintre acestea este TOSCA, care descrie topologia aplicației și modul de implementare a acesteia. Alte sisteme utilizează BPMN — un limbaj de procese de afaceri care permite modelarea logicii de executare a sarcinilor.

Abordările moderne se dezvoltă în direcția orchestrării bazate pe intenție (*intent-based orchestration*). Aici, administratorul nu descrie pașii, ci rezultatul: de exemplu, „crearea unui cluster rezistent la erori pentru procesarea datelor”, iar sistemul determină singur ce resurse și acțiuni sunt necesare pentru atingerea obiectivului.

Exemple și instrumente

Cel mai cunoscut exemplu de orchestrare este Kubernetes. Acesta gestionează containerele, decide unde trebuie să fie lansate, cum să fie scalate și cum să interacționeze între ele.

În domeniul gestionării datelor, este populară platforma Apache Airflow, care orchestrează fluxurile de date și procesele analitice.

Există și soluții specializate pentru nivelul infrastructurii — Cloudify, Morpheus, SaltStack, Ansible Tower. Toate acestea permit construirea de scenarii complexe de interacțiune între diferite servicii.

Problemele orchestrării

Orchestrația este complexă nu numai din punct de vedere tehnic, ci și logic. Este necesar să se țină seama de dependențe, erori, acțiuni compensatorii. O singură defecțiune poate provoca o reacție în cascadă. Prin urmare, sistemul de orchestrație trebuie să fie stabil, să poată reveni la operațiuni și să păstreze starea.

În plus, apare problema scalării. Orchestratorul trebuie să gestioneze mii de procese, păstrând stabilitatea și productivitatea. O altă provocare este compatibilitatea. Multe șabloane depind de un anumit furnizor, ceea ce face dificilă transferul între platforme.

Cu toate acestea, orchestrația reprezintă vârful automatizării. Ea leagă toate elementele infrastructurii, transformându-le într-un proces coordonat.

Exemple de orchestrare (Cloud Orchestration)

Exemplul 1: Scenariu complet de implementare a unei aplicații multistrat

Imaginați-vă că aplicația are trei straturi: baza de date, serverul de aplicații și front-end-ul, și interacționează cu un echilibrator extern. Orchestrația stabilește un scenariu în care mai întâi se creează rețeaua și subrețelele, apoi se implementează baza de date, apoi serverul de aplicații, apoi front-end-ul, apoi se conectează la echilibrator și se verifică integritatea funcționării. Dacă un pas nu a fost parcurs, se revine la pasul anterior sau se repetă. Orchestratorul coordonează secvența tuturor pașilor automatizați.

Exemplul 2: Migrarea între clouduri (Cross-Cloud Orchestration)

Aveți două clouduri: AWS și Azure. Trebuie să transferați o parte din încărcătură de la AWS la Azure. Orchestratorul lansează scenarii automatizate: exportul datelor din AWS, transferul, implementarea în Azure, instalarea serviciilor, testele, comutarea traficului. Toate acestea constituie un proces unic, cu procesarea erorilor și menținerea coerenței.

Exemplul 3: CI/CD-pipeline cu orchestrare infrastructurală

Orchestrația gestionează lanțul: preluarea unei noi versiuni a aplicației, efectuarea testelor automate, apoi eliminarea versiunii anterioare, implementarea noii versiuni în infrastructură, actualizarea configurațiilor, rularea scenariilor de verificare, comutarea traficului. Aici, orchestratorul interacționează cu sistemele de compilare, testare și infrastructură.

Exemplul 4: Auto-reparare

Orchestratorul monitorizează starea sistemului prin monitorizare. Dacă se constată că unul dintre componente nu mai răspunde, se lansează un scenariu: se dezactivează exemplarul cu probleme, se creează unul nou, se restabilește datele și se comută traficul, se notifică sistemul de monitorizare — totul automat și în cadrul unui singur proces.

IV. Interdependența dintre automatizare, management și orchestrare

Acum, că am analizat cele trei concepte separat, este important să înțelegem cum interacționează ele între ele.

Automatizarea asigură executarea operațiunilor specifice. Managementul gestionează politicile și controlul. Orchestrarea leagă toate acestea într-un sistem integrat.

Imaginați-vă că un utilizator solicită prin portalul de autoservire implementarea unei noi aplicații. Nivelul de management verifică cotele, drepturile și politica de securitate. Apoi se lansează orchestrația, care declanșează scenarii automatizate: creează o rețea, implementează baze de date, containere, echilibratoare și teste. După finalizare, orchestrația raportează că este gata, iar managementul înregistrează utilizarea resurselor și actualizează rapoartele.

Când aplicația nu mai este necesară, orchestrația execută o secvență inversă de acțiuni — eliberează resursele, elimină componentele temporare și finalizează procesul. Toate acestea se întâmplă în mod automat și controlat.

Această abordare poate fi comparată cu un organism viu. Automatizarea este mușchii, managementul este creierul, iar orchestrația este sistemul nervos care leagă totul împreună. Fără unul dintre aceste elemente, organismul nu poate funcționa.

V. Perspective și dezvoltare

Dezvoltarea tehnologiilor cloud se îndreaptă spre o autonomie din ce în ce mai mare. Automatizarea devine intelectuală — scenariile sunt completate cu elemente de învățare automată. Managementul se transformă într-o platformă analitică care prezice sarcinile și

optimizează utilizarea resurselor. Orchestrarea devine intenționată – în loc de procese pas cu pas, sistemul se concentrează pe rezultatul final.

În viitor, cloud-ul va semăna din ce în ce mai mult cu un sistem autonom, în care omul doar stabilește obiectivele, iar infrastructura determină singură calea optimă pentru atingerea acestora.

Dar pentru a ajunge la acest nivel, trebuie să începem cu elementele de bază: automatizarea corectă, managementul bine structurat și orchestrația bine gândită. Aceste trei componente creează fundamentul maturității cloud și determină eficiența întregului ecosistem.

Referință

https://www.researchgate.net/publication/390890431_Automation-Driven_Cloud_Migration_A_Case_Study_in_Retail_Infrastructure_Modernization

<https://enterpriseproject.com/article/2021/1/how-approach-infrastructure-automation>

<https://habr.com/ru/companies/fgts/articles/590433/>