Riassunti di ITCloud

Simone Montali
monta.li

23 gennaio 2020

1 Gestione di sistemi Linux

1.1 La bash

1.1.1 Che cos'è una command line?

Una command line è un'interfaccia testuale utilizzabile per inserire istruzioni in un computer. La command line di Linux è fornita da un programma chiamato shell. Bash è una versione migliorata di sh, sta infatti per Bourne Again Shell. Oggi, altre shell sono disponibli, come ZSH, che fornisce funzionalità aggiornate rispetto a bash. Quando una shell è usata interattivamente, mostra una stringa con utente, host, working directory.

1.1.2 Che cos'è il globbing?

Il globbing è il path name matching della shell (nato da global command); permette di utilizzare dei meta-characters (wildcards) per matchare nomi di file e directories.

2 Virtualizzazione

2.1 Introduzione a VMware vSphere

2.1.1 Che cos'è la virtualizzazione?

La virtualizzazione permette di ospitare più sistemi operativi all'interno di una stessa macchina fisica, riducendo lo spreco hardware, astraendo gli elementi hardware e rendendoli disponibili sotto forma di risorse virtuali. La differenza con l'emulazione è che in un emulatore, tutte le operazioni sono svolte da un software che gira nel suddetto, che fa da ponte. Invece, con una macchina virtuale (ossia l'insieme delle risorse virtuali), le operazioni vengono eseguite direttamente sull'hardware dell'host.

2.1.2 Che cos'è un hypervisor?

L'hypervisor è il componente che, nell'approccio alla virtualizzazione di VMware, agisce da ponte tra l'hardware fisico e le macchine virtuali, permettendo la creazione di "hardware virtuale". L'hypervisor di VMware si chiama ESXi. Le macchine fisiche su cui è installato un hypervisor sono dette host, mentre le VM ospitate sono dette guest. L'hypervisor permette la creazione di VM e la distribuzione di risorse.

2.1.3 Quali sono i vantaggi delle VM?

Primo fra tutti i vantaggi è quello della riduzione dei costi, diminuendo le risorse sprecate mantenute in idle, siano esse CPU, memoria, elettricità. Vi sono poi anche dei vantaggi di scalability (ossia la possibilità di variare le risorse di una VM dinamicamente), fault tolerance (ad esempio il trasferimento rapido di una VM in caso di errori), compatibilità (la VM vede sempre lo stesso hardware).

2.1.4 Che cos'è un'infrastruttura virtuale?

L'infrastruttura virtuale è il sistema che governa il data center, permettendo di gestire tutte le varie risorse fisiche e virtuali da uno stesso punto di accesso (che, più avanti, vedremo essere vCenter Server), e prevedendo la completa virtualizzazione di server, storage, risorse. È possibile configurare le VM con schede di rete (dotate di IP e MAC), rendendole indistinguibili da macchine fisiche.

2.1.5 Che cos'è il cloud computing?

Il cloud computing prevede più server, su data center, connessi su reti eterogenee. Si tratta in pratica di un modo per nascondere le caratteristiche fisiche di un sistema. Distinguiamo due tipologie: il cloud pubblico, di proprietà di un'organizzazione ma venduto agli utenti, e privato, a disposizione di una sola organizzazione. C'è anche la possibilità di un hybrid cloud, dove cloud pubblici e privati si mischiano in una sola entità. Qualsiasi azienda può costruirsi un'infrastruttura cloud privata senza dover pagare servizi esterni. I cloud pubblici realizzati con VMware prevedono tre tipologie di servizio: Basic (non riservato, pay per use) per operazioni poco costose, Committed vDC (risorse riservate espandibili in caso di bisogno), Dedicated vDC (risorse riservate su hardware dedicato). Distinguiamo inoltre tre categorie di lavoro: transient (lavoro poco frequente), higly elastic (con consumo molto variabile), infrastructure (consumo lineare nel tempo)

2.1.6 VMware vSphere, che cos'è?

VMware vSphere è una suite di software, funzionalità, servizi pensati per il cloud computing; è il primo sistema operativo cloud del settore, e trasforma i data center in infrastrutture di cloud computing semplificate, e consente di virtualizzare totalmente server, storage e hardware di rete, fornendo funzioni per l'alta disponibilità. Distinguiamo tra servizi per le applicazioni (ossia che forniscono controlli integrati sui livelli di servizio di tutte le applicazioni eseguite su vSphere) e servizi per l'infrastruttura (che permettono la virtualizzazione completa di server, storage, risorse di rete).

2.1.7 Elencare funzionalità e servizi di vSphere

Elenchiamo le varie funzionalità:
Elaborazione: l'hypervisor ESXi si occupa di partizionare un server fisico in più macchine virtuali, eseguibili simultaneamente. Sostituisce la precedente ESX, basata su RedHat, con una versione che astrae dal kernel linux ed ha quindi un kernel proprietario (VMKernel), molto più leggero, sicuro, ed adatto agli utilizzi prospettati.
Storage: abbiamo qui diversi servizi. Citiamo Virtual Machine File System (FS di tipo cluster), vStorage Thin Provisioning (consente l'allocazione dinamica dello spazio), Storage DRS (migliora la gestione delle risorse tramite raggruppamento, posizionamento, bilanciamento), Profile-Driven Storage (identifica lo storage appropriato in base al livello di servizio), Storage I/O Control (migliora la gestione dei Service Level Agreement tramite l'estensione dei limiti e delle condivisioni nei datastore NFS), Storage API (raccolta di API utile a rilevare le funzionalità di LUN/datastore degli storage array, grazie anche a VASA).
Networking: citiamo vSphere Standard e Distributed Switch. Questi sono switch virtuali che permettono

il virtual networking, consentendo alle VM di comunicare tra loro con gli stessi protocolli degli switch fisici. Possiamo anche agganciarli alla rete fisica associandoli alle relative interfacce.
Disponibilità: citiamo vMotion(vSphere e Storage), HA, Fault Tolerance. I primi permettono di migrare VM da un host all'altro in tempo reale, senza disservizi, addirittura host+storage contemporaneamente. HA consente di riavviare in pochi minuti VM e applicazioni bloccate, anche cambiando host. HA controlla costantemente tutti gli host e rileva i blocchi. Necessita di uno storage condiviso. Fault Tolerance è il più sicuro dei tre: consiste nel duplicare la VM su un altro host, tenendola in attesa. Abbiamo una doppia occupazione di risorse, ma downtime azzerato, in quanto la seconda VM è sempre pronta. È quindi adatto ad applicazioni mission-critical su cluster. Inoltre, possiamo utilizzare vSphere Data Protection per eseguire backup e ripristino senza agenti esterni, e Replication per il disaster recovery su un sito remoto, replicando VM accese da un host all'altro.
Sicurezza: possiamo trasferire le funzionalità di scansione antivirus ad una macchina virtuale con sicurezza rinforzata, tramite vShield Endpoint.
Automazione: tramite Host Profiles possiamo creare dei "template" utili alla creazione di altre VM, e tramite vSphere Auto Deploy possiamo centralizzare l'installazione di patch sugli host grazie agli standard PXE/gPXE. Update Manager automatizza il monitoraggio, il patching e l'aggiornamento degli host. Altri due strumenti fondamentali per rendere il data center cost-effective sono Distributed Resource Scheduler, che bilancia i carichi di lavoro, e Distributed Power Management che raggruppa gle VM consumanti energia per mandare in standby alcuni host. Citiamo infine Hot Add, Hot Plug, Hot Extend, ossia funzionalità "a caldo", cioè modifiche alle caratteristiche delle VM senza lo spegnimento delle stesse. Utile ricordare che non possiamo togliere risorse, ma solo aggiungerle.

2.2 vCenter Server

2.2.1 Come comunicano gli host con vCenter Server?

L'accesso agli host ESXi da parte di vCenter Server è permesso da un servizio chiamato vpxa che dialoga con hostd (il processo più importante degli host) e vpxd, il servizio in esecuzine su vCenter Server. È necessario lasciare alcune porte aperte sul firewall: 80 e 443 (HTTP/S), 389 (LDAP, directory services), 636 (comunicazione tra vCenter in SSL), 902 (scambio di dati tra vCenter e gli host, tra cui heartbeat), 903 (interfaccia console delle VM su vSphere Client), 8080 (connessioni Web Services HTTP), 7005-7009-7080-7444 (servizi SSO), 9090-9443 (vSphere Web Client), 10080-10109-10111-10443 (vCenter Inventory Service).

2.2.2 Disponibilità del vCenter Server

È necessario fornire alta disponibilità ai suoi componenti Active Directory e Database. Una buona soluzione è eseguirlo su VM, in modo da poterlo coprire con HA (che richiede vCenter Server solo per la configurazione iniziale). In alternativa, possiamo usare il vCenter Server Heartbeat, che esegue il failover su un server in standby con rilevamento preciso dei componenti.

2.3 vSphere ESXi

2.3.1 Che cos'è vSphere ESXi?

ESXi è l'hypervisor di VMware vSphere (disponibile anche in versione gratuita), supporta fino a 512 VM, 2048 CPU, 2Tb di RAM. È necessaria CPU fisica a 64 bit con set di istruzioni LAHF e SAHF. L'architettura di ESXi è basata sul VMKernel, un kernel realizzato ad hoc che permette di estrapolare risorse hardware per rilasciarle a più macchine virtuali. VMKernel riceve le richieste di risorse dalle VM attraverso il Virtual Machine Monitor, e le presenta all'HW. VMKernel è a 64 bit, supporta Intel Xeon o AMD Opteron. Le

VM possono essere anche 32 bit. Può essere installato su hard disk, USB, SD o direttamente su Storage Area Network. È richiesto minimo 1Gb, ma se si usa un disco locale o una LUN iSCSI sono richiesti 4 gb ulteriori di scratch partition (non creata invece su SD/USB per non sovraccaricarne l'I/O).

2.3.2 Com'è gestita la sicurezza di ESXi?

Il VMKernel di base è progettato per le VM, quindi le interazioni col mondo esterno sono ridotte. Abbiamo alcune funzionalità per la protezione del kernel, come il memory hardening: tutti gli eseguibili (kernel, applicazioni, librerie, driver...) vengono caricati su indirizzi di memoria casuali, rendendo l'esecuzione di exploit di memoria complicata. Questa funzionalità si affianca alle funzioni native NX e XD di Intel e AMD, che marcano le pagine di memoria come data-only, per evitare l'esecuzione di codice sulle stesse ma limitarle al salvataggio di dati. È possibile nascondere queste due flag alle VM. Grazie alla Module Integrity, possiamo firmare digitalmente moduli, applicazioni e driver, in modo da assicurarne integrità ed autenticità. Infine, grazie a Trusted Platform Module certifichiamo i processi di avvio, permettendo la memorizzazione sicura delle chiavi di crittografia e di protezione. È necessario abilitarlo sul BIOS.

2.3.3 Come funziona il firewall di ESXi?

L'interfaccia di gestione degli host ESXi è protetta da un firewall stateless, dedicato alla protezione dei servizi interni, con chiusura di default.

2.3.4 Che cos'è la modalità lockdown?

La modalità lockdown inibisce l'accesso diretto all'host: qualsiasi modifica dev'essere eseguita tramite vCenter Server, con l'eccezione dell'utente root che ha sempre e comunque i privilegi di utilizzo della console.

2.4 Profili Host

I profili host permettono di incapsulare la configurazione di uno specifico host all'interno di un template, per velocizzare la creazione di host simili, con configurazione, impostazioni di rete, storage, sicurezza... Il profilo host viene assegnato ad un reference host, di norma corrispondente al generatore del profilo, che agisce da modello. Possiamo usare questo modello per verificare la conformità dell'host col profilo associato.

2.5 Autenticazione e controllo accessi

Il sistema di controllo accessi prevede la possibilità di definire permessi per utenti/gruppi con grande precisione. Bisogna introdurre alcuni concetti: un privilege è un permesso per un utente di eseguire determinate azioni, un role è un insieme di privilegi. Di default, abbiamo tre ruoli: administrator, read-only, no access. Attenzione: i ruoli creati sul vCenter non sono visibili dagli host direttamente. Un object è un'entità dell'ambiente su cui è possibile eseguire azioni, ed ha quindi dei privilegi. User/group definiscono utenti o gruppi di utenti, definibili su vCenter o direttamente sugli host.

2.5.1 Come funzionano i ruoli default di vSphere?

Abbiamo tre ruoli predefiniti non modificabili: no-access, read-only, administrator. Per apportare modifiche è necessario clonarli. No access è il più stringente: l'utente non ha alcun tipo di visibilità, ed i tab appariranno vuoti. Si utilizza per revocare permessi su un oggetto figlio. Read only permette la visibilità dello stato ed i dettagli dell'oggetto. Administrator ha tutti i privilegi possibili. Di default, due utenti

sono administrator: root e vpxuser. Per creare ruoli è consigliabile assegnare il minor numero possibile di privilegi e dare nomi esplicativi.

2.6 Virtual networking

2.6.1 Che cos'è il virtual networking?

Il networking virtuale permette il collegamento in rete, tramite switch virtuali, delle VM ed addirittura il loro interfacciamento alla rete fisica. Gli switch virtuali consentono di comunicare con gli stessi protocolli degli switch fisici, senza HW aggiuntivo: le VM hanno infatti schede ethernet virtuali con IP e MAC.

2.6.2 Componenti principali del networking

Innanzitutto, c'è bisogno di interfacce di rete virtuali (Virtual NIC), utilizzate dalle VM. Poi abbiamo bisogno di switch virtuali standard (vSphere Standard Switch), che collegano le VM tra loro e con la rete fisica esterna (sfruttando le risorse/porte uplink dell'host), aventi dei gruppi di porte (Port Groups), ossia insiemi di porte accomunate dalle stesse caratteristiche. Infine citiamo gli switch virtuali distribuiti (vSphere Distributed Switch) che permettono di operare come se su avesse un singolo switch centralizzato. Uno switch virtuale funziona come uno switch Ethernet fisico, al livello 2 ISO-OSI, mantenendo una tabella di MAC address. A differenza degli switch fisici, non c'è bisogno di fasi di apprendimento dei MAC, conosciuti direttamente dagli host. Non possiamo assegnare alla stessa interfaccia fisica più switch virtuali, e non possiamo interconnetterli tra loro (Virtual Switch Isolation): evitiamo così loop di rete.

2.6.3 Che cos'è una porta di uplink?

Una porta di uplink è la porta di uno switch virtuale associata ad una o più interfacce fisiche di rete dell'host, e permette il collegamento della rete virtuale a quella fisica. Ognuna delle interfacce fisiche dell'host è detta semplicemente uplink. Possono anche esistere switch virtuali senza uplink: formiamo così una virtual intranet.

2.6.4 Che cos'è un port group?

I port group sono una funzionalità virtuale non disponibile nelle reti fisiche; essi sono insiemi di porte con caratteristiche comuni, che permettono quindi di collegare VM a porte senza specificarne il numero preciso ma solo il port group desiderato.

2.6.5 Tipologie di connessione di un vSwitch

Un vSwitch permette due tipologie di connessione: VMkernel e Virtual Machine. La prima connette ai servizi tramite le cosiddette VMKernel ports, gestendo servizi come l'accesso allo storage IP, le migrazioni vMotion, le funzioni di Fault Tolerance, l'accesso alla rete di management. La seconda è una connessione per le VM tramite port group. L'installazione di ESXi comporta la creazione di un vSwitch predefinito, con un port group VM Network ed una porta VMKernel detta MAnagement Network, utilizzata per la festione dell'host.

2.6.6 Cosa sono le VLAN?

Le VLAN sono un metodo per segmentare un dominio di broadcast (ossia la parte di rete raggiunta dai pacchetti di broadcast) in più domini di dimensione ridotta. A livello 2, ogni VLAN contiene solo il traffico dei dispositivi appartenenti a quella VLAN. Per definirne, uno switch associa ogni porta ad un

identificativo, detto VLAN ID. È necessario il protocollo IEEE 802.1q, supportato da ESXi. Distinguiamo tra porte di accesso (usate per collegare gli host) e trunk (per gli uplink tra switch/router). Le VLAN si configurano a livello di port group.

2.6.7 Come viene bilanciato il carico di rete?

Uno switch virtuale può essere connesso a più interfacce fisiche, e sfruttare più uplink in un processo detto NIC Teaming. Grazie al suddetto possiamo fare load balancing, ossia distribuzione del traffico attraverso più uplink (ottenendo un throughput più alto), e failover (instradamento del traffico su un'altra interfaccia quando la prima non è più operativa).

2.6.8 Elenca le tipologie di Load Balancing nel NIC Teaming

Abbiamo 4 differenti approcci:
Route in base al port ID: il traffico in uscita dalla rete virtuale è mappato ad un'interfaccia fisica. Ogni VM ha una porta, il traffico in uscita è inviato sempre allo stesso uplink (a meno di failover) e quello in entrata anche. Non pesa molto sulla CPU ma non è un vero e proprio load balancing.
Route in base all'hash del MAC sorgente: funzionamento molto simile al precedente. Viene hashato il MAC della sorgente; una VM non può avere più uplink a meno di non avere più MAC address. A differenza del primo, non garantisce l'uso di uplink differenti quando una VM ha più interfacce virtuali.
Route sull'IP Hash: qui, l'uplink è deciso in base all'hash di source/destination IP. Ogni sessione ha quindi uplink differente. Lo switch fisico deve supportare il protocollo EtherChannel (802.3ad). Qui il load balancing è ottimale: infatti le VM possono utilizzare più uplink.
Use explicit failover order: non c'è bilanciamento del carico ma solo failover, che ricade sull'uplink in linea da più tempo.

2.6.9 Tipologie di Failover nel NIC Teaming

Distinguiamo tra Link Status Only e Beacon Probing. Il primo si basa solo sullo stato di collegamento, quindi non rileva errori di configurazione o applicativi. Il secondo è basato sull'invio di beacon packets, che hanno un funzionamento simile ai ping ma in broadcast. Se un uplink non riceve pacchetti per tre volte consecutive, viene considerato failed. Inoltre è utile citare l'opzione Notify Switch, tramite cui possiamo notificare agli switch fisici esterni i cambiamenti di failover. Ricordiamo che di default è attivo il failback: se una scheda failata torna online, riprende servizio immediatamente. Distinguiamo tre gruppi di interfacce: active (utilizzate), standby (in attesa per failover), unused (non utilizzate).

2.6.10 Cosa si intende con Jumbo Frame?

Di default, la MTU (Maximum Transmission Unit) del livello Network ISO-OSI è 1500 byte. Per migliorare il rendimento della rete (e ridurre i carichi sulla CPU) possiamo aumentare la dimensione con i cosiddetti Jumbo Frames, frame thernet con dimensioni maggiori di 1500 bytes. Se un dispositivo della rete li utilizza, anche gli altri devono, con lo stesso valore MTU. ESXi supporta jumbo frames fino a 9000 byte.

2.7 Virtual Networking Distribuito

Il networking virtuale distribuito si basa sull'impiego di switch virtuali distribuiti (vSphere Distributed Switch), oggetti festiti da vCenter Server, creati per avere una configurazione del networking uguale su tutti gli host. Uno switch distribuito ha porte e gruppi distribuiti. Come negli switch standard abbiamo delle Uplink Ports (dvUplink), a cui possiamo associare una o più interfacce fisiche degli host. Tutte le

dvUplink sono organizzate in un unico gruppo detto Uplink Port Group, gestito da vCenter Server, che modifica tutte le politiche di rete e regole insieme.

2.8 Storage virtuale

2.8.1 Quali sono le tecnologie disponibili per lo storage?

vSphere supporta diverse tecnologie di storage: il primo che citiamo è ovviamente lo storage locale, che non può essere condiviso ed i cui dati sono accessibili solo dall'host. Poi, parliamo di Fibre Channel, tecnologia usata nelle Storage Area Network che consente di convogliare i segnali su fibre ottiche col protocollo Fibre Channel Protocol, impiegato per il trasporto dei comandi SCSI. Questo protocollo è incapsulabile su ethernet, e viene detto FcoE (Fibre Channel over Ethernet). iSCSI è un protocollo che permette l'impacchettamento di comandi SCSI su TCP/IP, inviati da un client detto initiator. Infine, parliamo di NAS, dispositivi di storage condiviso che rendono disponibile il loro spazio in rete tramite protocollo NFS, che però non supporta dischi RDM.

2.8.2 Che cos'è un datastore?

Un datastore è un'unità di memorizzazione, contenitore logico che si mostra agli host come storage generico, astraendo dalla reale implementazione/tecnologia di storage utilizzata. Possiamo utilizzare due tipi di file system: VMFS o NFS. Il primo è un FS di tipo cluster, permette quindi I/O da più host (fino a 64) contemporaneamente. Supporta unità fino a 64Tb, grazie a GUID Partition Table (non più MBR). Utilizza blocchi da 1Mb (adatti ai file grandi), e utilizza un indirizzamento per sotto-blocchi con file di piccole dimensioni, ottimizzando l'uso dello spazio. Per la concurrency, si sfrutta un meccanismo di blocco distribuito. Invece, NFS è un FS che permette di utilizzare i dischi remoti come se fossero locali. È reso disponibile da dispositivi NAS (solitamente basati su Unix), ai quali accediamo tramite porte VMKernel, che permettono di sfruttare NFS. Dobbiamo ovviamente essere sulla stessa rete, possibilmente separata e dedicata ad NFS. In ESXi, i privilegi di accesso NFS sono solitamente assegnati a root. Se questo crea dei problemi, dobbiamo modificare l'opzione root-squash del NAS.

2.8.3 Cos'è uno storage iSCSI?

Un sistema di storage basato su iSCSI lavora come una SAN, in modo che lo spazio sia disponibile a qualsiasi server della LAN tramite comandi SCSI su TCP/IP. Un host ESXi deve utilizzare un client iSCSI, detto initiator (software o hardware con HBA), che consente di inviare al target i comandi. Initiator e target sono detti nodi iSCSI identificabili da un IQN (iSCSI Qualified Name), lungo fino a 255 caratteri e con prefisso iqn.

2.8.4 Quali tipi di adattatori iSCSI possiamo usare?

Distinguiamo tre tipi di adattatori iSCSI. Quello software è implementato direttamente nel VMKernel, permettendo di usare iSCSI senza hw aggiuntivo. Poi, abbiamo i Dependent Hardware iSCSI Adapters, ossia interfacce di rete in cui parte dello stack iSCSI è implementata in hardware, portando a un risparmio in CPU/memoria. Hanno comunque configurazione da ESXi. Negli ultimi, gli Independent Hardware iSCSI Adapters (iSCSI HBA), tutto lo stack è implementato al livello hardware. L'interfaccia di gestione è qui implementata direttamente nel firmware. Gli ultimi due adattatori liberano molte risorse del server, rendendole disponibili alle operazioni I/O in caso di necessità. Ovviamente aumentano i costi, ma permettono anche Jumbo Frame e multipathing.

2.8.5 Com'è gestita la sicurezza in iSCSI?

I dati iSCSI non sono criptati nella rete. È quindi necessario implementare, almeno, un metodo di autenticazione come CHAP. CHAP verifica periodicamente l'identità dei nodi tramite un handshake a tre vie basato su un segreto condiviso. ESXi supporta CHAP a livello di adapter, sia one-way (il target autentica l'initiator) che mutual (entrambi si autenticano). Per la seconda sono necessari adattatori software o dependent, che inoltre supportano la per-target CHAP (credenziali diverse in base al target). Possiamo forzare CHAP con 4 livelli: nessun utilizzo, non utilizzo se non richiesto dal target, utilizzo se non proibito dal target, utilizzo obbligato. L'autenticazione avviene solo nel momento della connessione, non dopo.

2.8.6 Che cos'è Storage Fibre Channel?

Gli host ESXi supportano Fibre Channel e FCoE, composto da vari elementi: una SAN composta da hard disk e controller, delle LUN (Logical Unit Numbers) unità logiche identificate da indirizzi numerici (dischi singoli o RAID), uno storage processor che s'interpone tra host e dischi (con cache), HBA (Host Bus Adapter), scheda d'espansione che consente la connessione dell'host alla SAN tramite un WWN (di nodo o porta). Infine, è necessaria una fibre channel fabric, ossia l'insieme di uno o più switch fibre channel che permettano la connessione degli HBA. Per garantire la disponibilità, ESXi supporta il multipathing, per cui in caso di blocco di un HBA o switch è possibile accedere alla memoria da altri path. È però necessario avere due o più HBA, se possibile collegati da due o più switch e due storage processor.

2.8.7 Quali sono i metodi di controllo degli accessi degli host alle LUN?

Distinguiamo 3 metodi per il controllo degli accessi alle LUN: nel soft zoning ci basiamo sugli identificativi WWN, nell'hard zoning ci basiamo sulle porte degli switch, nel LUN masking ci basiamo sulla visibilità di una LUN ad un host, implementabile sia a livello ESXi che a livello di storage processor. Gli switch fibre channel implementano lo zoning per impedire il traffico non voluto, permettendo di impedire ai server non-ESXi di accedere allo storage, di ridurre il numero di LUN, di isolare determinati path.

2.8.8 Che cos'è Fiber Channel over Ethernet?

Possiamo accedere a una SAN Fibre Channel attraverso ethernet grazie al protocollo FCoE, che incapsula il traffico in un frame ethernet. Servono adattatori hardware o software. Nel primo caso, sono detti CNA (Converged Network Adapter), e contengono nella stessa interfaccia sia la componente Ethernet sia quella Fibre Channel, entrambe visibili da ESXi. Gli adattatori software sono invece interfacce di rete con supporto hardware ai processi FCoE.

2.8.9 Cosa intendiamo con NPIV?

N-Port ID Virtualization è uno standard che descrive come un HBA può registrarsi ad una Fibre channel fabric con diversi nomi (WWPN). Un utilizzo tipico è associato all'uso di dischi RDM.

2.8.10 Che cos'è il thin provisioning?

VMware vStorage Thin Provisioning consente un utilizzo dinamico dello storage da parte delle VM, tramite allocazione intelligente dello spazio: lo spazio occupato da una VM non sarà quello riservato, ma solo quello effettivamente utilizzato. Questo riduce di molto i costi in memoria dell'infrastruttura.

2.8.11 Come integriamo lo storage con vCenter Server?

VMware ha sviluppato delle API specifiche per gestire i dispositivi di storage da vCenter Server: vSphere Storage APIs for Storage Awareness (VASA). Con queste possiamo verificare e gestire caratteristiche e funzionalità dello storage da vCenter Server tramite funzionalità definite in base al sistema, grazie a plug-in creati dai produttori dello storage stesso.

2.8.12 Percorsi multipli per lo storage

Per garantire affidabilità, generalmente raddoppiamo gli storage processor, in modalità active-active o active-passive. Nella prima, l'accesso alle LUN è consentito da tutto gli storage processor disponibili, mentre nel secondo, solo uno storage processor rimane attivo, mentre l'altro attende. VMware offre bilanciamento dei carichi e meccanismi di failover nativi, con 3 diverse politiche di gestione dei percorsi: fixed, in cui si usa sempre il preferito (se è disponibile), Most Recently Used, in cui si usa l'ultimo finché è disponibile, Round Robin che utilizza tutti i percorsi disponibili a rotazione.

2.9 Le macchine virtuali

2.9.1 Qual è l'hardware di una VM?

La virtualizzazione permette di ospitare più sistemi operativi, distribuendo le risorse disponibili astraendole e virtualizzandole. L'insieme di queste risorse virtuali si dice macchina virtuale (VM). Tutte le macchine virtuali hanno una certa uniformità per quanto riguarda il tipo di hardware, rendendole molto portbaili. Una VM può avere una o più CPU virtuali, con il limite massimo di quelle fisiche (logiche, ossia socket*core). ESXi supporta fino a 64 CPU. È usuale avere almeno un disco, avente in automatico un controlle SCSI per la connessione. L'adattatore SCSI può essere di diversi tipi, scelto automaticamente in base al SO installato. Una VM ha poi interfacce di rete che possono essere flessibili (interfaccia vlance/vmxnet se sono installati i VMware tools) oppure emulare schede reali, come la E1000. La differenza tra vlance e vmxnet è che la seconda è paravirtualizzata, progettata specificatamente per ambienti virtuali (quindi i driver vanno installati). Citiamo inoltre VMXNET2, che aggiunge jumbo frame e hardware off-load, e VMXNET3, che aggiunge multiqueue, offload IPv6, MSI. Quest'ultima è consigliata, ma supportata solo da pochi SO e VM con hardware versione 7 o superiore. Una VM può poi avere fino a 1Tb di RAM ed altri dispositivi come DVD/CD, seriali, PCI, mousem tastiera, USB...

2.9.2 Che cos'è Raw Device Mapping?

RDM è una modalità di accesso per macchine virtuali, che consente l'accesso diretto a una LUN presente sullo storage fisico tramite un file puntatore .vmdk contenente informazioni di puntamento. Questa mappatura consente alle LUN di apparire come facenti parte di un volume VMFS: la VM vede il dispositivo RDM come un disco SCSI virtuale. Gli host ESXi supportano 2 modalità RDM: physical compatibility (il guest accede direttamente allo storage, con passaggio diretto SCSI e caratteristiche LUN visibili), e virtual compatibility (VMKernel invia solo i comandi di lettura/scrittura al dispositivo mappato).

2.9.3 Cos'è un template?

Un template è la versione master di una macchina virtuale, da cui possiamo creare nuove VM con caratteristiche predeterminate (SO guest, applicazioni installate, configurazioni hardware). Possiamo creare il template con una clonazione a template (si mantiene la macchina originale in esecuzione) o una conversione a template. Le operazioni sui template sono possibili solo dal vCenter. La clonazione su template permette di scegliere il formato dei dischi generati, mentre la conversione non prevede possibilità di scelta.

2.9.4 Clonazione di una macchina virtuale

Clonare una VM significa crearne una copia esatta da vCenter; possiamo eseguirla sia a caldo che a freddo. È necessario personalizzare le macchine clonate per evitare conflitti in rete: i parametri di rete rimangono gli stessi! Possiamo utilizzare per questo scopo la Guest customization dei VMware Tools.

2.9.5 Cos'è lo snapshot di una macchina virtuale?

Lo snapshot si una VM corrisponde ad una sua istantanea registrata su disco. L'istantanea può essere utilizzata in qualsiasi momento per riportare la macchina virtuale allo stato di snapshot. Possiamo sfruttare questa funzione prima di modifiche sui software, come aggiornamenti o installazioni. È possibile creare più snapshot, in una struttura padre/figlio. Solo l'ultimo figlio è modificabile. È fondamentale notare che gli snapshot non sono un valido metodo di backup. È inoltre consigliabile evitare di lasciare attive le snapshot create, perché generano overhead: l'operazione di eliminazione è detta consolidation, ed avviene manualmente o durante il cloning della VM. Possiamo escludere i dischi dal processo di snapshot impostandoli come dischi indipendenti. Questi ultimi possono essere persistent (senza delta files) o nonpersistent (con redo log che salva le modifiche).

2.9.6 Da quali file è composto uno snapshot?

Una snapshot è composta da diversi file. Il file più importante è il Memory State file, che rappresenta lo stato della memoria. Ne viene creato uno per snapshot, e può includere il contenuto della memoria. Abbiamo poi lo Snapshot description file, che contiene informazioni sulla snapshot. Lo snapshot delta file contiene le modifiche effettuate sul disco della VM successive alla creazione di una snapshot. Infine, lo snapshot list file mantiene le informazioni riguardanti tutte le snapshot della VM.

2.10 Profili storage per VM

2.10.1 Cosa sono i profili storage?

vSphere permette di mettere in relazione le diverse tipologie di datastore con le VM attraverso il profile-driven storage, una funzionalità che fornisce posizionamento rapido e intelligente delle macchine virtuali. Le diverse storage capabilities possono essere collegate ad un profilo storage, e quest'ultimo ad una VM. I profili sono impiegati durante il provisioning, la duplicazione e l'uso di Storage vMotion.

2.10.2 Quali sono le funzionalità di Profile-Driven Storage?

Profile-Driven Storage ha integrazione completa con VASA, un set di API per la comunicazione con vCenter Server, che le inserisce nelle storage capabilities. Ha supporto per NFS, iSCSI, FC e per tutti gli array di storage. Ha possibilità di creare regole di posizionamento per le VM sotto forma di profili storage, e possibilità di verifica delle suddette.

2.11 Migrazione con vSphere vMotion

2.11.1 Quali sono i tipi di migrazione possibili?

Distinguiamo tra le diverse possibilità di migrazione tra VM: la meno difficile è quella cold, ossia a VM spenta. Poi, parliamo di migrazione Suspended quando la VM è in pausa. Per le VM accese, invece, è necessario vMotion: semplice per spostamento tra host diversi, e Storage per migrazione su un altro datastore. Da vSphere 5.1 è addirittura possibile migrare senza uno storage condiviso: in pratica possiamo effettuare simultaneamente il cambio di storage e host. Con cold e suspended possiamo spostare però VM

ad altri datacenter. vMotion non modifica i dischi RDM ma ne consente lo spostamento. Con vMotion, l'intero stato della VM viene traslato da un host all'altro: memoria, hardware, rete... Prima di tutto viene però svolta una preverifica dei requisiti, che restituisce warning ed errori. I requisiti sono vari: VMkernel, interfacce fisiche gigabit (anche in teaming), copresenza della stessa rete virtuale, mancanza di virtual intranets (switch privi di uplinks).

2.11.2 Funzionamento di vMotion

Avviata l'operazione, la memoria della VM viene copiata sull'host di destinazione, mentre gli utenti continuano a usare la source. Viene tenuta traccia delle modifiche avvenute in questo lasso di tempo grazie ad una memory bitmap. Prima di essere trasferita, la VM viene messa in uno stato di quiescenza (impossibilità di modifica), durante il quale viene trasferito sulla destinazione lo stato dei dispositivi e la memory bitmap. Appena attivato lo stato di quiescenza, la VM viene avviata sulla destinazione. Si informa la rete del cambio di switch (tramite RARP) e si termina la sincronizzazione.

2.11.3 Cosa sono le flag NX e XD?

Queste due flag indicano tecnologie proprietarie di AMD (No eXecute) e Intel (eXecute Disable), che marcano le pagine di memoria come data-only per evitare l'esecuzione di codice dannoso/buffer overflow. Vengono isolate le aree di memoria da dedicare alle istruzioni della CPU oppure ai dati. Se un'area di memoria ha questo flag non vi possono risiedere istruzioni. Se la tecnologia è abilitata sulla source, vMotion la richiede anche sulla destinazione. Queste flag possono essere esposte alla VM (default) oppure no, per aumentare la compatibilità a scapito della sicurezza.

2.11.4 Che cos'è vSphere Storage vMotion?

Storage vMotion impiega la stessa tecnologia di vMotion, applicata però ai filem spostati da un datastore all'altro mantenendo le VM accese. La tecnologia è indipendente dal tipo di storage: è compatibile con NFS, VMFS/iSCSI, Fibre Channel... Una migrazione di questo tipo prevede un'architettura di tipo mirroring, con copia dei blocchi disco tra source e destinazione. Si effettua una prima copia, durante la quale le modifiche sono tracciate dal Mirror Driver, che poi le copia su entrambi i dischi e attende il consenso per l'invio della modifica al SO. Questo aumenta l'efficienza e la prevedibilità temporale delle migrazioni. I dischi non possono essere in modalità non-persistent.

2.12 Cluster DRS e bilanciamento tra host

2.12.1 Che cos'è un cluster?

Un cluster è un insieme di host ESXi e relative VM, dove le risorse sono condivise e l'interfaccia di gestione è comune; un cluster DRS è un cluster con il servizio vSphere Distributed Resource Shoulder abilitato, che permette di distribuire e bilanciare le risorse fisiche. Distinguiamo tre modalità: initial placement, load balancing, power management. La prima è la più semplice: DRS posiziona la VM quando viene avviata. Con il load balancing, DRS monitora in modo continuo la distribuzione delle risorse memoria/CPU tra host e VM, con un uso ideale definito dal Migration Threshild. L'ultimo, power management, permette di avere risparmi nei consumi: vengono migrate le VM accorpandole il più possibile, allo scopo di poter mettere in standby alcuni host. Possiamo anche decidere dove posizionare i file di swap.

2.12.2 Modelli di automazione DRS

Distinguiamo tre modelli di automazione: manuale (con suggerimenti), parzialmente automatizzato (automatico all'avvio host, poi manuale), completamente automatizzato (DRS interviene per spostare). Definiamo un migration threshold, su 5 livelli, dall'1, il più conservativo, a 5, molto aggressivo.

2.12.3 Cosa sono i gruppi DRS e regole?

Possiamo imporre al sistema l'osservazione di determinati criteri per quanto riguarda il posizionamento delle VM, detti affinity rules: possiamo decidere di mantenere determinate VM in un host (Keep virtual machines together), tenerle separate (Separate virtual machines), e legare un gruppo di VM a un gruppo di hosts (Virtual machine to hosts).

2.12.4 Cos'è EVC?

VMware Enhanced vMotion Compatibility (EVC) permette di migliorare la compatibilità del vMotion fra gli host del cluster, facendo si che le CPU fisiche offrano alle VM lo stesso set di istruzioni. Grazie a EVC è possibile aggiungere nuove CPU nel cluster, configurate in modalità compatibile con le esistenti. Tuttavia, è prevista compatibilità solo tra CPU dello stesso produttore. La compatibilità è misurata a livelli detti baseline.

2.12.5 Cos'è DPM?

VMware Distributed Power Management ottimizza l'efficienza energetica nei cluster DRS, grazie alla costante ottimizzazione dei consumi. DPM sfrutta DRS per migrare le VM dagli host ESXi che possono essere spenti. Dimensioniamo infatti le capacità del cluster in base alle necessità dei carichi di lavoro. In caso di necessità si riaccendono gli host spenti. Se Fault Tolerance è attivo, gli host non vengono spenti.

2.12.6 Cos'è un datastore cluster?

Il datastore cluster è un insieme di datastore in cui le risorse storage sono condivise e gestite insieme. Un cluster può contenere datastore con dimensioni e capacità I/O diverse, basta che il datastore sia in un solo datacenter, abbia ESXi¿5, non venga scollegato dall'host quando rimosso. Sono garantite due funzionalità: initial placement (alla creazione di una VM, Storage DRS seleziona il datastore su cui posizionarne i dischi in base ai vincoli di spazio e bilanciamento I/O, tentando di ridurre le attività intense I/O e il sovrautilizzo) e ongoing balancing (monitoraggio ogni 8 ore dei datastore per controllare il superamento dei limiti, anche sulle macchine spente e gli snapshot).

2.13 High Availability e Fault Tolerance

2.13.1 Cos'è vSphere HA?

Il servizio vSphere HA può intervenire in caso di blocco degli host, dei SO, delle applicazioni, riavviando le VM su un altro host. Non è detto che il riavvio risolva il problema. Ci sono alcuni requisiti: il cluster non deve avere più di 32 host, 4000 VM, 512 host per VM. Il servizio si abilita all'interno di un cluster, attivando il servizio Fault Domain Manager, in modo che l'host sappia di essere parte di un fault domain. Deve rispettare tre condizioni: essere connesso a vCenter, non essere in maintainance, non essere in standby. Il fault domain è gestito da un host master, gli altri sono detti slave; il master è eletto in base alla quantità di datastore connessi.

2.13.2 Come si rilevano i disservizi?

Il master invia periodicamente degli heartbeat sulle reti di management per informare gli slave della sua presenza. Gli slave utilizzano una sola rete di management. In caso di fail si utilizzano i datastore, che fungono da canale di comunicazione alternativo; se lo slave non risponde sulla rete, HA effettua una verifica degli heartbeat tramite datastore per capire se l'host è funzionante. Il datastore utilizzato è quello con il più alto numero di host ESXi connessi. Una volta che il master denota il fail, lo slave è etichettato come agent unreachable e le VM vengono avviate su altri nodi. La rete di management è ridondante grazie a due interfacce di rete in teaming.

3 Docker

3.0.1 Che cos'è Docker?

Docker è uno strumento creato da Solomon Hykes nel 2013, che forza le persone a pensare in termini di microservices. È un software Linux (ora non più esclusivamente), non è un linguaggio di programmazione o un framework, ma risolve problemi di distribuzione e deployment del software, semplificando la vita dei system admins. Docker non ha creato il concetto di container, ma piuttosto ne ha semplificato l'utilizzo.

3.0.2 Che cos'è un container?

Il container è un concetto che fu inventato ben prima di Docker. Prima del 2005 si parlava anche di "jail", equivalente a dei runtime environments dove un programma aveva il permesso di accedere a risorse protette. Quello del container è un concetto simile alle macchine virtuali, ma con una grandissima differenza: le VM utilizzano la virtualizzazione dell'hardware, mentre i container si interfacciano direttamente col kernel Linux. Questo li rende molto più leggeri, performanti, portabili. I programmi che girano su un container possono accedere solo a memoria e risorse del container. I container sono isolati su 8 aspetti:
namespace PID: id processi
namespace UTS: host e domain names
namespace MNT: accesso e struttura del file system
namespace IPC: comunicazione dei processi su memoria condivisa
namespace NET: accesso in rete e struttura
namespace USR: username e identificatori
chroot(): root del FS
Cgroups: protezione delle risorse Un container è riempito con un'immagine (che è uno snapshot), disponibile su registri/index (pubblici o anche privati).

3.0.3 Quali vantaggi porta l'utilizzo di Docker?

Docker risolve molti problemi del deployment di software. Prima di tutto, migliora la portabilità e l'abstraction: oltre a "shippare" il codice, "shippiamo" anche l'ambiente su cui il codice gira, rendendone l'esecuzione perfetta su ogni tipologia di sistema. Altrimenti potremmo avere problemi di compatibilità con le API native del sistema operativo, problemi di memoria, di configurazione di rete... L'environment del software è quindi molto più assicurato e stabile, permettendoci di concentrarci sui dettagli più importanti. Questa astrazione ci dà anche vantaggi di sicurezza: i contenuti del container accedono solo a un set limitato di risorse, quindi in caso di fault non c'è il rischio di diffusione del problema.

3.0.4 Quali sono gli stati possibili per un container?

Un container può avere diversi stati:
created: il container è stato creato (docker create) ma non avviato
restarting: il container sta venendo riavviato
running: il container sta girando
paused: i processi del container sono stati pausati
exited: un container che ha terminato la sua esecuzione
dead: un container che il daemon ha cercato di stoppare, senza successo

3.0.5 Come otteniamo un vero Environment-Agnostic System?

Con environment agnostic system intendiamo la creazione di un sistema "generico" per ottenere portabilità ed astrazione. Docker ottiene questo obiettivo con 3 features: read-only filesystems, environment variable injection, volumes. Grazie al primo, il container non può venire specializzato da modifiche ai file, apportando anche sicurezza verso i malintenzionati. Le envars sono utilizzate per comunicare informazioni tra container, e vengono injectate all'esecuzione del container. I volumi, infine, sono utilizzati per la persistenza dei dati: essi creano un mount point sul container, in modo da salvare i dati direttamente nell'host. Ne distinguiamo due tipi: bind-mount e docker-managed. Il primo è una directory nel computer, per condividere i dati con l'ambiente esterno. Il secondo viene creato in una directory di Docker, ed è utilizzato quando la posizione dei dati non è importante. In entrambi i casi stiamo attenti a concorrenza e ad orfani generati con l'eliminazione dei container (utilizzare -v). Possiamo anche condividere volumi tra containers in due modi: dando la stessa host folder, oppure in modo generico con l'opzione –volumes-from. Infine, c'è anche la possibilità di condividere la stessa memoria tra due container (più sicuro).

3.0.6 Come funziona il network su docker?

Docker utilizza l'OS sottostante per creare una rete virtuale. Le reti virtuali possono interagire a 4 livelli diversi: closed, bridged, joined, open. Nel primo abbiamo solo l'interfaccia di loopback. Il bridged, oltre a questa, ha anche un'interfaccia privata, che può interagire col Docker Bridge. Docker fornisce differenti opzioni anche per la configurazione del DNS. Il comportamento predefinito per i bridged è quello di non essere accessibili dalla rete: per permetterlo utilizziamo l'opzione –publish. I container bridged possono comunicare tra loro normalmente, possiamo disabilitare questo comportamento con il parametro -icc falso. Due container joined condividono lo stack di rete, quindi non c'è isolamento tra loro. Utilizziamo questo livello quando ci serve una sola interfaccia di loopback, o abbiamo bisogno di monitorare il traffico di rete per un programma in un altro container. Gli open container, infine, sono totalmente aperti. Non andrebbero usati. A creation time un container può essere linkato ad un altro con l'opzione –link.

3.0.7 Come si possono limitare i container?

Possiamo applicare restrictions ai container con le opzioni -m, –cpu-shares, rispettivamente i mb di RAM e la percentuale di CPU utilizzabile. Per limitare la CPU usiamo anche –cpuset-cpus per limitare i core.

3.0.8 Come funzionano gli utenti in Docker?

L'utente start di default per un container è root, ma si può modificare. Questo porta ad un dubbio: se scaricassimo un'immagine da internet, che utente verrà usato? Non c'è un comando preciso per farlo. Possiamo utilizzare docker inspect per verificare se è stato settato un utente diverso da root, ma c'è anche la possibilità di cambiare l'utente in uno script init: è sicuro azzerare l'entrypoint.

3.0.9 Come si builda un'immagine?

Il building di un'immagine è composto di tre fasi: creazione di un container, modifica del filesystem, commit delle modifiche. Per controllare le modifiche possiamo eseguire un docker diff. È utile settare un entrypoint, ossia un programma da eseguire all'avvio del container. Ogni layer dell'immagine eredita da quello precedente. Nell'immagine è salvato anche tutto il metadata context: env variables, working dir, porte esposte, entrypoint, comandi, argomenti.

3.0.10 Come funziona Union File System?

Union File System è un sistema a strati: quando un file viene letto, si parte dal layer più alto. Un'eliminazione in UFS non è una vera eliminazione: semplicemente il file viene mascherato nel layer più alto. Infatti, i layer sottostanti al più alto sono read-only. Quando un layer viene committato, riceve un identifier, contiene il precedente, e l'execution context. Questi identifier sono hash, quindi per migliorare la memorabilità, docker fornisce le repository con nomi e versioni.

3.0.11 Che cos'è un Dockerfile?

Un Dockerfile è un file contenente istruzioni per il building di un'immagine, rendendolo tracciabile e riproducibile. Ogni istruzione costituisce un nuovo layer: è meglio accorparle e non superare le 42 righe. È utile citare anche il .dockerignore, che contiene una lista di file esclusi dalla copia.

3.0.12 Parla del design delle immagini

La maggior parte delle immagini ha bisogno di uno script iniziale come entrypoint, scritto solitamente in bash/sh, ed è meglio verificare le precondizioni, fallire velocemente in caso non siano rispettate, e droppare i privilegi al più presto. Docker fornisce anche delle funzionalità di distribuzione di immagini, come i registri, ossia servizi che rendono repository disponibili, come Docker Hub. Per scegliere la soluzione adatta alla distribuzione delle nostre immagini vanno tenuti in conto più fattori: costi, visibilità, velocità, longevità, integrità, confidenzialità, facilità.

3.0.13 Che cos'è un webhook?

Un webhook è un modo, per una repo di Git, di notificare modifiche all'image repository.

3.0.14 Che cos'è Docker Compose?

Docker Compose è un tool per definire ed eseguire applicazioni multi-container, definendole tramite semplici file YAML. Col comando docker-compose possiamo buildare le immagini, lanciare i container, lanciare sistemi di servizi, gestirne lo stato, lavorare sulla scalability e visualizzare i log.

3.0.15 Che cos'è Docker Machine?

Docker Machine è un tool che permette di installare Docker Engine su host virtuali, gestendoli con comandi docker-machine. Prima della 1.12, era l'unico metodo di eseguire Docker su Windows e Linux. Permette di creare host dockerizzati.

4 Kubernetes

4.0.1 Che cos'è kubernetes?

Kubernetes è un orchestratore di container: un cluster di nodi con gestione dell'infrastruttura ed automazione. Consente infatti di eliminare molti dei processi legati al deployment di container, permettendo la gestione semplice di cluster. È una valida alternativa a Docker Swarm, in quanto fornisce strumenti per la scalabilità di ambienti di produzione di cluster di container. È un progetto Google open source nato dalla CNCF, leader nel programma degli orchestrator.

4.0.2 Quali sono i benefici di Kubernetes?

Kubernetes apporta diversi benefici:
Paradigma dichiarativo: lo stato desiderato è descritto in file YAML in modo non imperativo, ma dichiarativo, definendo l'infrastruttura necessaria all'applicazione in produzione.
Distribuzione automatica: il collocamento dei runtime sui nodi è automatico, rispetto a requirements/constraint con politiche di massima disponibilità e load balancing
Scaling orizzontale/verticale: col primo intendiamo la modifica della quantità di container, col secondo la modifica delle risorse cpu/ram. Può essere manuale o automatico.
Rollout e rollback: intendiamo con questi termini l'applicazione progressiva di modifiche ed aggiornamenti, monitorando costantemente lo stato e svolgendo modifiche automatiche.
Gestione dello storage: sono supportati vari storage provider locali, in rete, in cloud; manuali o auto-provisioned.
Self healing: grazie ad health check definiti dall'utente, abbiamo riavvio e ricollocamento dei container/nodi in fault.
Service Discovery: abbiamo meccanismi built-in per la pubblicazione di servizi ed endpoint, un DNS interno per gli IP dei container, networking virtuale sul cluster e load balancing.
Gestione secret e configurazione: le informazioni sensibili sono gestite esplicitamente, le configurazioni disaccoppiate.

4.0.3 Cos'è un pod Kubernetes?

Un pod di Kubernetes è un gruppo di container deployati insieme sullo stesso host, con shared network, namespaces, risorse. Possiamo interagire col cluster tramite kubectl.