

## 1. Introduzione

### 1.1 Architettura multi-tier, [1]

Nell'ingegneria del software, il termine architettura multi-tier o architettura multi-strato (spesso definita con l'espressione inglese n-tier architecture) indica un'architettura software di tipo client-server per sistemi distribuiti, in cui le varie funzionalità del software sono logicamente separate ovvero suddivise su più strati o livelli software differenti in comunicazione tra loro (nel caso di applicazioni web questi strati sono la logica di presentazione, l'elaborazione dei processi e la gestione della persistenza dei dati).

Ciascuno strato è in comunicazione diretta con quelli adiacenti ovvero richiede ed offre servizi allo strato adiacente in maniera concettualmente simile a quanto accade con le architetture di rete a strati.

L'architettura delle applicazioni N-tier fornisce un modello per gli sviluppatori per creare vantaggiosamente un'applicazione flessibile e riutilizzabile ovvero scalabile. Con la separazione di un'applicazione in livelli, per modificare o aggiungere funzionalità, gli sviluppatori possono infatti modificare solo uno specifico livello, piuttosto che dover riscrivere l'intera applicazione, garantendo dunque una maggiore semplicità di progettazione/implementazione secondo la filosofia del divide et impera ed una maggiore manutenibilità.

### Architettura three-tier

Three-tier è un'architettura client-server in cui l'interfaccia utente, i processi logico funzionali ("regole aziendali"), l'archiviazione informatica dei dati e l'accesso ai dati sono sviluppate e mantenute come moduli indipendenti, la maggior parte delle volte su piattaforme separate.

Il three-tier è un modello di architettura software e allo stesso tempo uno schema di progettazione software. Oltre ai vantaggi abituali di software modulare con interfacce ben definite, l'architettura three-tier è destinata a consentire a qualsiasi dei tre livelli di essere aggiornati o sostituiti indipendentemente dal cambiamento di requisiti o tecnologia. Ad esempio, un cambiamento di sistema operativo nel livello di presentazione interesserebbe solo il codice di interfaccia utente.

Three-tier architecture ha i seguenti tre livelli:

- Livello di presentazione. Questo è il livello più alto dell'applicazione. Il livello di presentazione mostra le informazioni relative a servizi come merce online, acquisti, e i contenuti del carrello della spesa. Comunica con altri livelli attraverso i risultati di output al livello browser/client e tutti gli altri livelli della rete;
- Livello applicazione (business logic, la logica di primo livello, l'accesso ai dati di secondo livello, o di livello intermedio). La logica di primo livello viene tirata fuori dal livello di presentazione e, come suo proprio livello, controlla la funzionalità di un'applicazione eseguendo elaborazioni dettagliate;
- Livello dati. Questo livello è costituito da server database. Qui le informazioni vengono memorizzate e recuperate. Questo livello mantiene i dati neutrali e indipendenti da applicazioni server o da logica di business. Fornendo informazioni del proprio livello inoltre migliora la scalabilità e le prestazioni.

## 1.2 Definizione di Cloud Computing, [1]

In parole semplici, il cloud computing è la distribuzione di servizi di calcolo, come server, risorse di archiviazione, database, rete, software, analisi e intelligence, tramite Internet ("il cloud"), per offrire innovazione rapida, risorse flessibili ed economie di scala. Paghi solo per i servizi cloud che usi e risparmi sui costi operativi, esegui l'infrastruttura in modo più efficiente e ridimensioni le risorse in base all'evoluzione delle esigenze aziendali.

**Principali vantaggi del cloud computing:** Il cloud computing rappresenta un grande cambiamento rispetto alla visione tradizionale delle aziende in materia di risorse IT. Ecco sette motivi comuni per cui le organizzazioni ricorrono ai servizi di cloud computing:

1. **Costo:** Il cloud computing elimina le spese di capitale associate all'acquisto di hardware e software e alla configurazione e alla gestione di data center locali, che richiedono rack di server, elettricità 24 ore su 24 per alimentazione e raffreddamento ed esperti IT per la gestione dell'infrastruttura. I conti tornano in fretta;
2. **Scalabilità globale:** I vantaggi dei servizi di cloud computing includono la possibilità di ridimensionare le risorse in modo elastico. In materia di cloud questo significa fornire la giusta quantità di risorse IT, ad esempio una quantità maggiore o minore di potenza di calcolo, risorse di archiviazione e larghezza di banda, proprio quando è necessario e dalla località geografica appropriata;
3. **Prestazioni:** I più grandi servizi di cloud computing vengono eseguiti su una rete mondiale di data center sicuri, aggiornati regolarmente all'ultima generazione di hardware, veloce ed efficiente. Questo offre diversi vantaggi rispetto a un singolo data center aziendale, tra cui latenza di rete ridotta per le applicazioni e maggiori economie di scala;
4. **Sicurezza:** Molti provider di servizi cloud offrono un'ampia gamma di criteri, tecnologie e controlli che rafforzano il comportamento di sicurezza complessivo, grazie alla protezione di dati, app e infrastruttura dalle minacce potenziali;
5. **Velocità:** La maggior parte dei servizi di cloud computing viene fornita in modalità self-service e su richiesta, quindi è possibile effettuare il provisioning anche di grandi quantità di risorse di calcolo in pochi minuti, in genere con pochi clic del mouse. Le aziende possono quindi usufruire di una grande flessibilità senza la pressione legata alla pianificazione della capacità;
6. **Produttività:** I data center locali richiedono in genere un impegno notevole nell'organizzazione e nell'assemblaggio dei rack, che include la configurazione dell'hardware, l'applicazione di patch software e altre attività di gestione IT dispendiose in termini di tempo. Il cloud computing elimina la necessità di molte di queste attività, consentendo ai team IT di dedicare il proprio tempo al raggiungimento di obiettivi aziendali più importanti;
7. **Affidabilità:** Il cloud computing aumenta la semplicità e riduce i costi di backup dei dati, ripristino di emergenza e continuità aziendale, grazie alla possibilità di eseguire il mirroring dei dati in più siti ridondanti nella rete del provider di servizi cloud.

**Tipi di cloud computing:** Non tutti i cloud sono uguali e non sempre lo stesso tipo di cloud computing è adatto a tutte le esigenze. Sono disponibili numerosi modelli, tipi e servizi diversi per offrire la soluzione più adatta in base alle tue esigenze.

Per prima cosa, devi determinare il tipo di distribuzione cloud, ovvero l'architettura di cloud computing, in cui verranno implementati i servizi cloud. Ci sono tre modalità diverse di distribuzione dei servizi cloud: in un cloud pubblico, in un cloud privato e in un cloud ibrido.

1. **Cloud pubblico:** I cloud pubblici sono di proprietà di un provider di servizi cloud di terze parti, che fornisce le risorse di calcolo, come server e risorse di archiviazione, tramite Internet. Microsoft Azure è un esempio di cloud pubblico. In un cloud pubblico, l'hardware, il software e l'infrastruttura di supporto appartengono al provider di servizi cloud, che li gestisce. Puoi accedere a questi servizi e gestire il tuo account usando un Web browser;
2. **Cloud privato:** Un cloud privato si riferisce alle risorse di cloud computing usate esclusivamente da una singola azienda o organizzazione. Un cloud privato può trovarsi fisicamente nel data center locale della società. Alcune società, inoltre, pagano provider di servizi di terze parti per ospitare il proprio cloud privato. Un cloud privato è un cloud in cui servizi e infrastruttura sono gestiti in una rete privata;
3. **Cloud ibrido:** I cloud ibridi combinano cloud privato e pubblico, grazie a una tecnologia che consente la condivisione di dati e applicazioni tra i due tipi di cloud. Grazie alla possibilità di spostare dati e applicazioni tra cloud pubblici e privati, un cloud ibrido offre all'azienda maggiore flessibilità e più opzioni di distribuzione e aiuta a ottimizzare l'infrastruttura esistente, la sicurezza e la conformità.

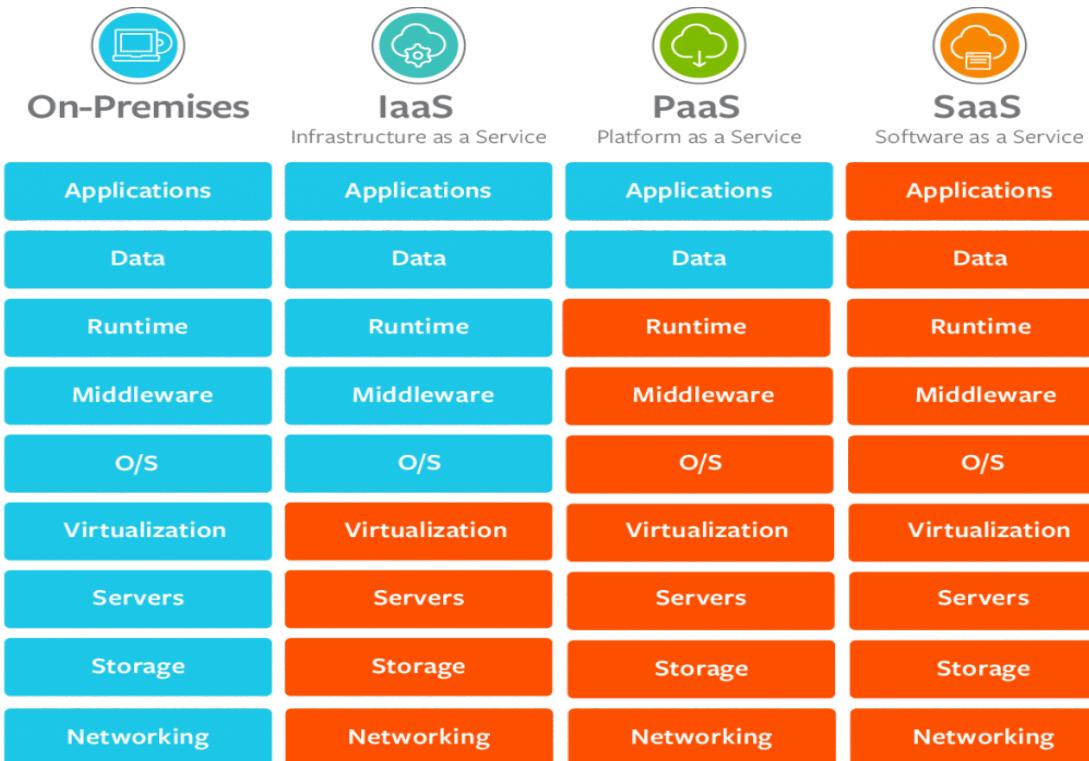
**Tipi di servizi cloud: IaaS, PaaS, serverless e SaaS:** La maggior parte dei servizi di cloud computing rientra in quattro ampie categorie: infrastruttura distribuita come servizio (IaaS), piattaforma distribuita come servizio (PaaS), elaborazione serverless e software come un servizio (SaaS). Talvolta si parla di "stack" di cloud computing, in quanto queste categorie sono basate una sull'altra. La conoscenza di queste soluzioni e delle loro differenze semplifica il raggiungimento degli obiettivi aziendali.

1. **Infrastruttura distribuita come servizio (Infrastructure as a service, IaaS):** Si tratta della categoria di base dei servizi di cloud computing. Con una soluzione IaaS, affitti l'infrastruttura IT, ovvero server e macchine virtuali (VM), risorse di archiviazione, reti e sistemi operativi, da un provider di servizi cloud con pagamento in base al consumo;
2. **Piattaforma distribuita come servizio (PaaS, Platform as a Service):** PaaS (piattaforma distribuita come servizio, Platform as a Service) si riferisce a servizi di cloud computing che forniscono un ambiente su richiesta per lo sviluppo, il test, la distribuzione e la gestione di applicazioni software. Una soluzione PaaS è progettata per consentire agli sviluppatori di creare in modo più semplice e rapido app Web o per dispositivi mobili, senza doversi preoccupare della configurazione o della gestione dell'infrastruttura di server sottostante, della rete di archiviazione e dei database necessari per lo sviluppo;
3. **Elaborazione serverless:** In sovrapposizione con il modello PaaS, l'elaborazione serverless si concentra sulla funzionalità di creazione delle app senza che sia necessario spendere tempo per la gestione dei server e dell'infrastruttura necessari. Il provider di servizi cloud gestisce automaticamente la configurazione, la pianificazione della capacità e la gestione dei server. Le architetture serverless sono basate su eventi, offrono scalabilità elevata e usano le risorse solo quando si verifica una funzione o un trigger specifico;
4. **Software come un servizio (SaaS, Software as a Service):** SaaS (Software as a Service, software come un servizio) è un metodo per la distribuzione di applicazioni software tramite Internet, su richiesta e in genere in base a una sottoscrizione. Con una soluzione SaaS, i provider di servizi cloud ospitano e gestiscono l'applicazione software e l'infrastruttura sottostante e si occupano delle attività di manutenzione, come gli

aggiornamenti software e l'applicazione di patch di protezione. Gli utenti si connettono all'applicazione tramite Internet, in genere con un Web browser nel telefono, tablet o PC.

**Usi del cloud computing:** Di seguito alcuni esempi di ciò che è possibile fare oggi con i servizi cloud di un provider di servizi cloud:

- **Crea applicazioni native del cloud:** Crea, distribuisci e ridimensiona rapidamente le applicazioni per il Web, i dispositivi mobili e le API. Sfrutta i vantaggi delle tecnologie e degli approcci nativi del cloud, tra cui contenitori, Kubernetes, architettura basata su microservizi, comunicazioni basate su API e DevOps;
- **Archiviare i dati ed eseguirne il backup e il ripristino:** Proteggi i dati razionalizzando i costi e su vasta scala, grazie alla possibilità di trasferire i dati tramite Internet su un sistema di archiviazione cloud esterno accessibile da qualsiasi posizione e da qualunque dispositivo;
- **Trasmettere in streaming audio e video:** Rimani in contatto con i tuoi destinatari ovunque, in qualsiasi momento e su qualunque dispositivo, grazie alle funzionalità audio e video ad alta definizione con distribuzione globale;
- **Fornire software on demand:** Anche noto come Software as a Service, il software su richiesta ti permette di offrire le versioni e gli aggiornamenti più recenti del software ai tuoi clienti, sempre e ovunque si trovino;
- **Testare e compilare le applicazioni:** Riduci i costi e i tempi di sviluppo delle applicazioni usando infrastrutture cloud che consentono di aumentare o ridurre facilmente le prestazioni in base alle esigenze;
- **Analizzare i dati:** Unifica i dati tra team, divisioni e sedi nel cloud. Usa quindi i servizi cloud, come Machine Learning e intelligenza artificiale, per acquisire informazioni dettagliate e prendere decisioni più informate;
- **Incorporare l'intelligence:** Usa modelli intelligenti per coinvolgere i clienti e raccogliere informazioni dettagliate preziose dai dati acquisiti.



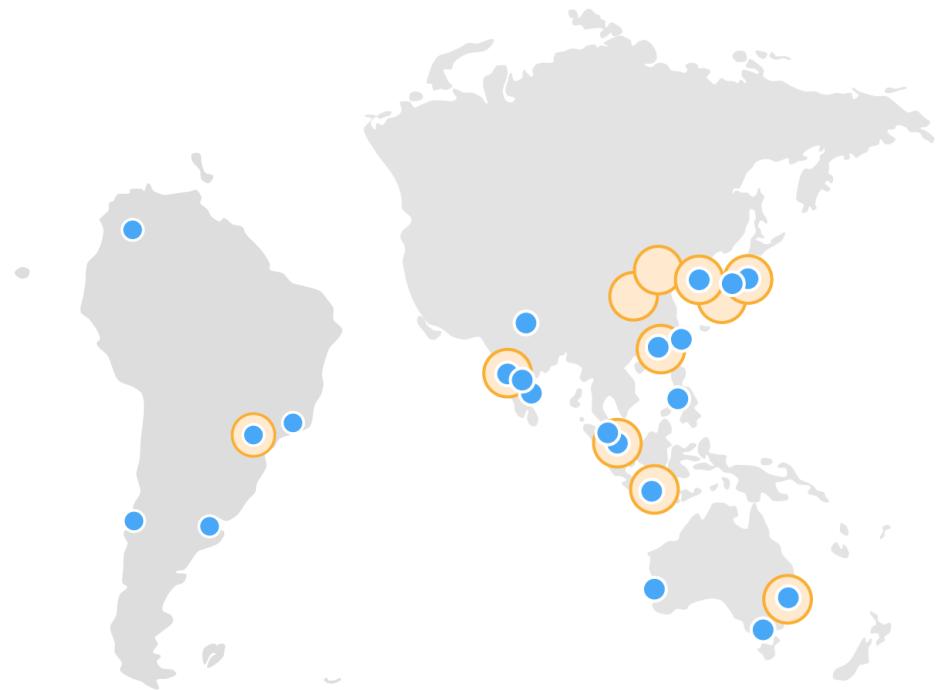
### 1.3 Regioni e zone di disponibilità, [1]

**Regioni:** In AWS è nato il concetto di Regione, intesa come luogo fisico nel mondo in cui si clusterizzano i data center. Ogni gruppo di data center logici viene chiamato **zona di disponibilità**. Ogni Regione AWS consiste in una serie di zone di disponibilità isolate e fisicamente separate all'interno di un'area geografica. A differenza di altri provider cloud che spesso definiscono una regione come un singolo data center, la struttura composta da varie zone di disponibilità di ogni Regione AWS offre molti vantaggi ai clienti. Ogni zona dispone di capacità di alimentazione, raffreddamento e sicurezza fisica proprie ed è connessa grazie a reti ridondanti e a latenza bassissima. I clienti AWS che cercano un'alta disponibilità possono progettare le loro applicazioni in modo da eseguirle in diverse zone di disponibilità per raggiungere una tolleranza ai guasti maggiore. Le Regioni dell'infrastruttura AWS hanno i più alti livelli di sicurezza, conformità e protezione dei dati.

AWS fornisce un'impronta globale più ampia rispetto agli altri provider cloud e per supportare questa impronta globale e assicurarsi di essere al servizio dei clienti di tutto il mondo, AWS apre costantemente nuove regioni. AWS gestisce varie regioni geografiche, comprese le regioni dell'America settentrionale, il Sud America, l'Europa, la Cina, l'Asia Pacifico, il Sud Africa e il Medio Oriente.

Di seguito la mappa delle regioni:

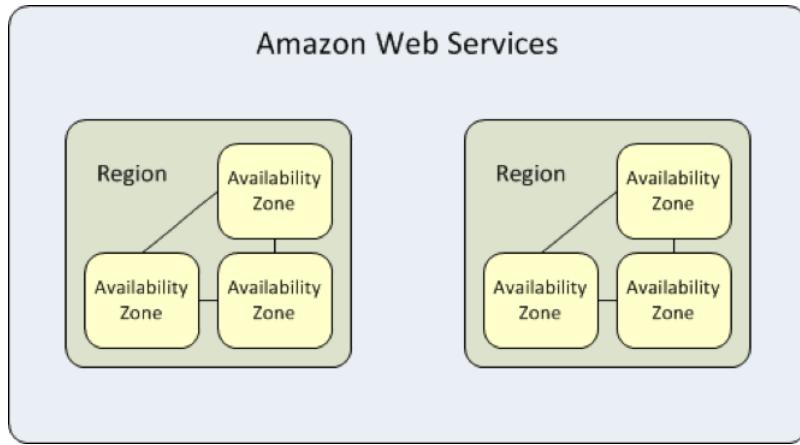




**Zone di disponibilità:** Una zona di disponibilità consiste in uno o più data center provvisti di alimentazione, rete e connettività ridondanti in una regione AWS. Le zone di disponibilità (AZ) consentono ai clienti di eseguire applicazioni e database in ambienti di produzione con elevata disponibilità, tolleranza ai guasti e scalabilità, altrimenti impossibili da ottenere all'interno di un singolo data center. Tutte le zone di disponibilità in una regione AWS sono interconnesse tramite una rete a elevata larghezza di banda e a bassa latenza, su una fibra metropolitana dedicata completamente ridondante che distribuisce reti a alto throughput e bassa latenza tra esse. Todo el tráfico entre las zonas de disponibilidad es cifrado. La prestación de red es suficiente para obtener una replicación sincrónica entre las zonas de disponibilidad. Las zonas de disponibilidad facilitan la partición de las aplicaciones para una disponibilidad elevada muy sencilla. La partición de una aplicación en varias zonas de disponibilidad permite la aislamiento de las empresas y las protege de problemas como el blackout, los truenos, tornados, terremotos y mucho más. Las zonas de disponibilidad están físicamente separadas entre sí por una distancia significativa de varios kilómetros, quedando dentro del radio de 100 km una de la otra.

**Zone locali AWS:** Le zone locali AWS avvicinano il calcolo, lo storage, il database e altri servizi AWS selezionati ai clienti finali. Grazie alle zone locali AWS puoi eseguire facilmente le parti sensibili delle applicazioni che richiedono latenze di pochi millisecondi agli utenti finali, come la creazione di contenuti per media e intrattenimento, i giochi in tempo reale, le simulazioni delle riserve, l'automazione del design elettronico e il machine learning.

Ciascun luogo che include una zona locale AWS rappresenta un'estensione della regione AWS in cui puoi eseguire le applicazioni sensibili alla latenza tramite servizi AWS quali Amazon Elastic Compute Cloud, Amazon Virtual Private Cloud, Amazon Elastic Block Store, Amazon File Storage e Amazon Elastic Load Balancing in aree geografiche vicine agli utenti finali. Le zone locali AWS offrono una connessione protetta e a elevata larghezza di banda fra i carichi di lavoro locali e quelli in esecuzione nella regione AWS, consentendo di ricollegarsi senza problemi agli altri servizi disponibili nella regione tramite le stesse API e gli stessi strumenti.



**Edge Location:** Le edge location sono data center AWS progettati per fornire servizi con la latenza più bassa possibile. Amazon ha dozzine di questi data center sparsi in tutto il mondo. Sono più vicini agli utenti rispetto alle regioni o alle zone di disponibilità, spesso nelle principali città, quindi le risposte possono essere rapide. Un sottoinsieme di servizi per i quali la latenza conta davvero utilizza le edge location, tra cui:

1. CloudFront (una [content delivery network](#)), che utilizza le edge location per memorizzare nella cache le copie del contenuto che serve, in modo che il contenuto sia più vicino agli utenti e possa essere consegnato loro più velocemente.
2. Route 53, che serve le risposte DNS dalle edge location, in modo che le query DNS che hanno origine nelle vicinanze possano essere risolte più velocemente (e, contrariamente a quanto si potrebbe pensare, è anche il principale database di Amazon).
3. Web Application Firewall e AWS Shield, che filtrano il traffico nelle edge location per fermare il traffico indesiderato il prima possibile.

Quali sono i vantaggi delle edge location AWS? Le EL riducono la latenza in un paio di modi.

- Alcuni servizi di edge location restituiscono una risposta rapida direttamente all'utente. Ad esempio, CloudFront memorizza nella cache il contenuto nelle edge location e tale contenuto può essere servito direttamente dalla cache. Poiché la edge location è fisicamente molto più vicina all'utente rispetto al server di origine, ha una latenza inferiore.
- Altri servizi di edge location instradano il traffico sulla rete AWS. AWS dispone di una spina dorsale di rete globale di collegamenti in fibra ridondanti a larghezza di banda elevata. Il traffico inviato su questa rete è spesso più veloce e affidabile della rete Internet pubblica, soprattutto su lunghe distanze. Ad esempio, se scarichi un oggetto utilizzando S3 Transfer Acceleration, quell'oggetto viaggia da S3 attraverso la rete globale AWS alla edge location più vicina e utilizza solo l'Internet pubblica per l'hop finale.
- Ci sono molte più edge location rispetto alle Regioni. Ciò significa che è più probabile che gli utenti si trovino vicino a una edge location e ottengano quelle risposte a bassa latenza. Amazon aggiunge regolarmente nuove edge location e gli utenti che vivono nelle vicinanze vedranno un miglioramento automatico delle prestazioni. Ad esempio, Amazon ha recentemente aggiunto la sua prima edge location in Thailandia. Se la tua applicazione utilizzava AWS Global Accelerator, sarebbe diventata più veloce per gli utenti tailandesi, senza alcuno sforzo da parte dell'utente.

## **2. EC2 (Amazon Elastic Compute Cloud), [1]**

Amazon Elastic Compute Cloud (Amazon EC2) fornisce capacità di calcolo scalabile in Amazon Web Services (AWS) Cloud. L'utilizzo Amazon EC2 elimina la necessità di investimenti anticipati in hardware e ti permette di sviluppare e distribuire più rapidamente le applicazioni. Puoi utilizzare Amazon EC2 per avviare il numero di server virtuali necessari, configurare la sicurezza e i servizi di rete, nonché gestire l'archiviazione. Amazon EC2 consente di dimensionarsi verso l'alto o verso il basso per gestire le variazioni a livello di requisiti o i picchi di popolarità, riducendo la necessità di elaborare previsioni relative al traffico.

Amazon EC2 offre le seguenti caratteristiche:

- Ambienti di elaborazione virtuale, noti come istanze;
- Modelli preconfigurati per le istanze, noti come Amazon Machine Image (AMI), contenenti i pacchetti di bit necessari per il server (compresi il sistema operativo e il software aggiuntivo);
- Varie configurazioni di CPU, memoria, archiviazione e capacità di rete per le istanze, note come tipi di istanza;
- Informazioni di login sicure per le istanze mediante le coppie di chiavi (AWS archivia la chiave pubblica, mentre l'utente archivia la chiave privata in un luogo sicuro);
- Volumi di archiviazione per i dati temporanei che verranno eliminati quando l'istanza viene arrestata o interrotta, noti come volumi di archivio istanza;
- Volumi di archiviazione persistente per i dati tramite Amazon Elastic Block Store (Amazon EBS), noti come volumi Amazon EBS;
- Più posizioni fisiche per le risorse, ad esempio istanze e volumi Amazon EBS, note come regioni e zone di disponibilità;
- Un firewall che ti permette di specificare i protocolli, le porte e gli intervalli di indirizzi IP di origine che possono raggiungere le istanze tramite i gruppi di sicurezza;
- Gli indirizzi IPv4 statici per il cloud computing dinamico, noti come indirizzi IP elastici;
- I metadati, noti come tag, che puoi creare e assegnare alle risorse Amazon EC2;
- Reti virtuali che puoi creare, che sono isolate logicamente dal resto del cloud AWS e che facoltativamente puoi collegare alla tua rete, note come cloud privati virtuali (VPC).

### **2.1 Opzioni di acquisto per le istanze, [1]**

Amazon mette a disposizione diversi modelli di pagamento per le istanze EC2 che permettono di gestire al meglio i costi AWS in funzione delle più disparate necessità aziendali. Attraverso un mix di soluzioni a breve e lungo termine possiamo gestire ogni tipo di workload nel modo più flessibile e dinamico possibile. La risposta alla nostra domanda precedente risulta quindi semplice: per gestire i costi cloud bisogna prima di tutto conoscerli. Partiamo dalle basi allora. Esistono 5 diversi modelli di pagamento per le istanze EC2:

- On-demand;
- Istanze riservate;
- Istanze spot;
- Host dedicati;
- Saving Plans.

Ma quali sono le differenze?

Nel resto dell'articolo li analizzeremo singolarmente per capire come funzionano e se possono aiutarci a contenere i costi AWS. Prima di tutto però, è bene precisare che nonostante i nomi

sembrino suggerire il contrario, quelli che seguono sono concetti di fatturazione, e non tipi di istanza.

### 2.1.1 Istanze on-demand

- Adatte a workload imprevedibili;
- Paghi solo quello che usi;
- Nessun impegno a lungo termine;
- Zero costi iniziali.



È il modello di pagamento che comporta zero costi iniziali e nessun impegno a lungo termine. Con l'on-demand AWS ti addebita la capacità di calcolo su base oraria, e puoi sempre aumentare o diminuire la potenza in base alle necessità della tua applicazione.

Questo modello di pagamento, oltre ad essere un ottimo modo di testare le istanze EC2 per la prima volta, è particolarmente indicato per quelle applicazioni che non richiedono carichi di lavoro fissi. In questi casi infatti l'on-demand rappresenta la soluzione ideale per usufruire di un'infrastruttura resiliente mantenendo contemporaneamente costi bassi e zero impegni a lungo termine.

Ma come abbiamo detto poco sopra, possiamo contenere i costi AWS rinunciando ad un pizzico di questa flessibilità.

### 2.1.2 Istanze riservate

- Adatte a workload fissi;
- Risparmi fino al 75%;
- Ti impegni per 1 o 3 anni;
- Puoi anticipare il pagamento



A differenza delle istanze on-demand, le istanze riservate permettono di risparmiare fino al 75% sui costi AWS con l'impegno di utilizzare una determinata configurazione hardware per periodi di uno o tre anni. Queste istanze, o meglio questo modello di pagamento, offre diverse opzioni adatte a diversi casi d'uso.

Più precisamente, con le istanze riservate possiamo scegliere:

- Pagamento – senza anticipo, con anticipo parziale, con anticipo totale;
- Termine – un anno, tre anni;
- Attributi – piattaforma, tenancy, famiglia di istanza, tipo di istanza;
- Portata – regionale, zonale;
- Classe – standard, convertibile.

#### Istanze riservate Standard:

Relativamente alla classe, le istanze riservate standard offrono il maggior livello di sconto al minor livello di flessibilità. Una volta acquistata un'istanza riservata standard infatti non potrai apportare grandi modifiche, ma sarai comunque flessibile sulla sua dimensione e sulla sua portata. Potrai quindi modificarla scegliendone una più grande o più piccola all'interno della stessa famiglia, e cambiare la portata da regionale a zonale. Inoltre, nel caso in cui non avessi più bisogno dell'istanza, potrai sempre rivenderla all'interno di un apposito Marketplace.

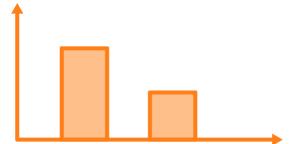
#### Istanze riservate Convertibili:

Rispetto alle Standard invece, le istanze riservate Convertibili permettono una maggiore flessibilità: risparmiando qualcosa in meno, hai la possibilità di cambiare l'istanza con una diversa a livello di dimensione, famiglia, piattaforma e tenancy. Se non ti senti a tuo agio con il

lock in delle Standard e vuoi comunque contenere i costi aws, questa è la scelta che fa per te. A differenza delle Standard però, le Convertibili non possono essere rivendute nel Marketplace, quindi valuta attentamente prima di scegliere questo modello di pagamento.

### 2.1.3 Istanze Spot

- Adatte a workload non critici;
- Risparmi fino al 90%;
- AWS può spegnerle senza preavviso;
- Decidi un prezzo massimo.



Le istanze Spot sono un caso particolare: si tratta di istanze EC2 inutilizzate che AWS può spegnere in qualsiasi momento e senza preavviso, ma che ti permettono di risparmiare fino al 90% rispetto alle istanze on-demand.

Sono indicate per casi d'uso specifici, ad esempio per esigenze di elaborazione urgenti o per workload non critici. In questi casi, le istanze spot possono essere un ottimo modo di tagliare i costi AWS pagando soltanto una frazione del prezzo on-demand.

Il prezzo delle istanze spot viene definito “prezzo spot”, ed è stabilito da AWS sulla base di diversi criteri, tra cui la disponibilità di istanze in una specifica Availability Zone. Ma sei tu a stabilire quanto sei disposto a pagare per una di queste istanze, impostando un’offerta di prezzo massimo. Solo se la tua offerta supera il prezzo spot -e se c’è disponibilità- puoi aggiudicarti queste istanze.

Come puoi immaginare, il prezzo spot è costantemente variabile. Ad esempio, nel momento in cui scrivo, il prezzo spot per un’istanza t3a.large è di \$0.0245 all’ora, con uno sconto del 69,9% rispetto al prezzo on-demand di \$0,0816.

Al contrario della volatilità delle Spot troviamo gli Host dedicati.

### 2.1.4 Host dedicati

Gli host dedicati si differenziano dagli altri modelli di pagamento perché sono basati su server EC2 fisici in cui puoi utilizzare le tue licenze software.

Puoi acquistarli pagandoli al prezzo on-demand, su prenotazione o come Saving Plans, di cui parleremo tra poco. Quando si acquista in modalità di prenotazione, viene fatturata ogni ora dell’intero periodo selezionato -uno o tre anni- indipendentemente dall’uso effettivo delle istanze all’interno dell’host.

### 2.1.5 Saving Plans

- Adatte a workload fissi;
- Risparmi fino al 72%;
- Semplificano le istanze riservate;
- Si applicano anche a Fargate e Lambda.



Saving Plans è un modello di prezzi flessibile che si pone l’obiettivo di semplificare l’acquisto di istanze riservate. È stato introdotto di recente, e sembra che AWS spinga gli utenti verso l’uso di questo nuovo modello di pagamento nonostante le istanze riservate restino ancora disponibili.

Con i Saving Plans, anziché impegnarti per preciso un consumo di risorse ti impegni per una precisa spesa oraria, sempre in termini di uno o tre anni. Le risorse che utilizzi e che ricadono entro i parametri del tuo Saving Plan vengono automaticamente fatturate al prezzo scontato, mentre quelle che superano la soglia vengono fatturate al normale prezzo on-demand.

Esistono due tipi di Saving Plans:

- EC2 Saving Plans;
- Compute Saving Plans.

Vediamo quali sono le differenze.

### **EC2 Saving Plans:**

Offrono sconti fino al 72% rispetto al prezzo on-demand e possono essere paragonati alle istanze riservate Standard. Rispetto a queste però, con gli EC2 Saving Plans puoi modificare le istanze a livello di sistema operativo e tenancy, oltre che di grandezza dell'istanza.

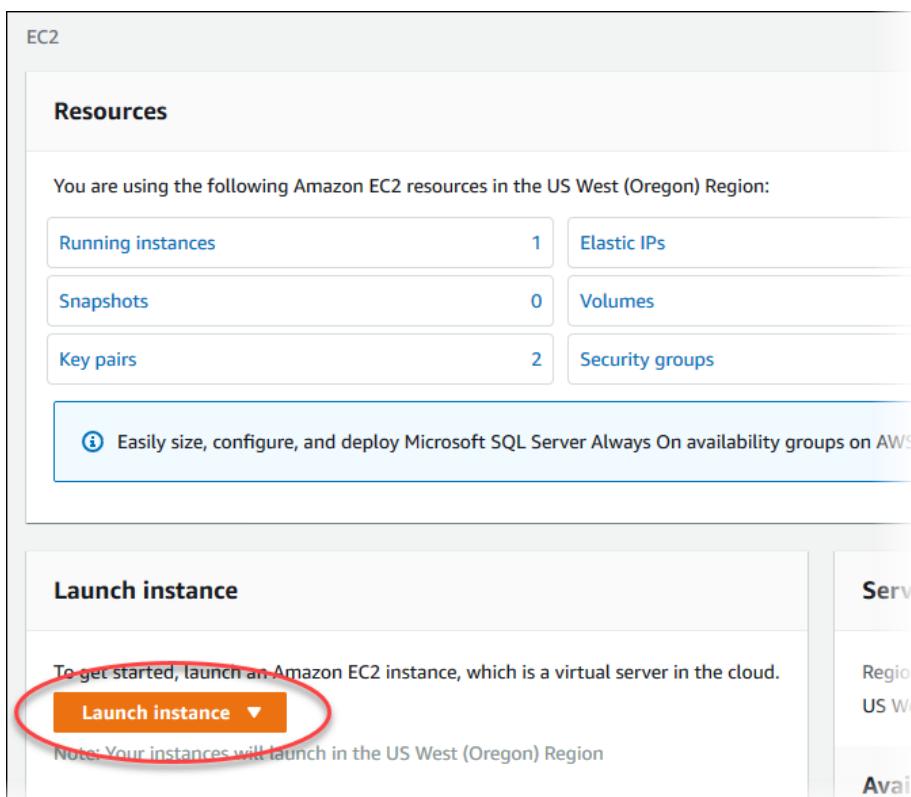
### **Compute Saving Plans:**

Offrono sconti fino al 66% e possono essere paragonati alle istanze riservate Convertibili, trovando qui importanti differenze. Con i Compute Saving Plans possiamo infatti modificare le nostre istanze a livello di grandezza, famiglia, sistema operativo, tenancy e soprattutto Region. Ad esempio, puoi cambiare la tua istanza da una famiglia ad un'altra e spostarla da una Region ad un'altra continuando ad usufruire dello stesso sconto. Oltre tutto, i Compute Saving Plans si applicano anche ad AWS Fargate ed AWS Lambda.

## **2.2 Creazione di un'istanza EC2 Linux, [1]**

Per avviare un'istanza EC2

1. Accedere a AWS Management Console e aprire la console Amazon EC2 all'indirizzo <https://console.aws.amazon.com/ec2/>
2. Selezionare Pannello di controllo EC2, quindi **Avvia istanza**, come visualizzato di seguito.



3. Scegli il Amazon Linux 2 AMI.

1. Choose AMI   2. Choose Instance Type   3. Configure Instance   4. Add Storage   5. Add Tags   6. Configure Security Group   7. Review   [Cancel and Exit](#)

### Step 1: Choose an Amazon Machine Image (AMI)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. You can select an AMI provided by AWS, our user community, or the AWS Marketplace; or you can select one of your own AMIs.

Search for an AMI by entering a search term e.g. "Windows"

Quick Start

My AMIs	Amazon Linux 2 AMI (HVM), SSD Volume Type - ami-07a0da1997b55b23e (64-bit x86) / ami-0787fda5708b00aa3 (64-bit Arm)	Select
AWS Marketplace	Amazon Linux 2 comes with five years support. It provides Linux kernel 4.14 tuned for optimal performance on Amazon EC2, systemd 219, GCC 7.3, Glibc 2.26, Binutils 2.29.1, and the latest software packages through extras. This AMI is the successor of the Amazon Linux AMI that is approaching end of life on December 31, 2020 and has been removed from this wizard.	<input checked="" type="radio"/> 64-bit (x86) <input type="radio"/> 64-bit (Arm)
Community AMIs	Root device type: ebs   Virtualization type: hvm   ENA Enabled: Yes	
<input type="checkbox"/> Free tier only <small>(i)</small>		
Red Hat	Red Hat Enterprise Linux 8 (HVM), SSD Volume Type - ami-01e78c5619c5e68b4 (64-bit x86) / ami-0a1158e1a81f6e09a (64-bit Arm)	Select
SUSE Linux	Red Hat Enterprise Linux version 8 (HVM), EBS General Purpose (SSD) Volume Type	<input checked="" type="radio"/> 64-bit (x86) <input type="radio"/> 64-bit (Arm)
	Root device type: ebs   Virtualization type: hvm   ENA Enabled: Yes	
	SUSE Linux Enterprise Server 15 SP2 (HVM), SSD Volume Type - ami-063c2d222d223d0e9 (64-bit x86) / ami-0fc92b18fd79372c (64-bit Arm)	Select
	<input checked="" type="radio"/> 64-bit (x86)	

4. Scegliere il tipo di istanza **t2.small**, come visualizzato di seguito, quindi **Next: Configure Instance Details (Successivo: configura i dettagli dell'istanza)**.

### Step 2: Choose an Instance Type

Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instances are virtual servers that can run applications. They have varying combinations of CPU, memory, storage, and networking capacity, and give you the flexibility to choose the appropriate mix of resources for your applications. [Learn more](#) about instance types and how they can meet your computing needs.

Filter by: All instance families (i) Current generation (i) Show/Hide Columns

Currently selected: t2.micro (- ECUs, 1 vCPUs, 2.5 GHz, -, 1 GiB memory, EBS only)

	Family	Type	vCPUs <small>(i)</small>	Memory (GiB)	Instance Storage (GB) <small>(i)</small>	EBS-Optimized Available <small>(i)</small>	Network Performance <small>(i)</small>	IPv6 Support <small>(i)</small>
<input type="checkbox"/>	t2	t2.nano	1	0.5	EBS only	-	Low to Moderate	Yes
<input checked="" type="checkbox"/>	t2	<b>t2.micro</b> <small>Free tier eligible</small>	1	1	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	t2	t2.small	1	2	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	t2	t2.medium	2	4	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	t2	t2.large	2	8	EBS only	-	Low to Moderate	Yes

[Cancel](#) [Previous](#) [Review and Launch](#) [Next: Configure Instance Details](#)

5. Nella pagina Configurazione dettagli istanza, visualizzata di seguito, impostare questi valori e lasciare gli altri valori come predefiniti:
  - Rete:** scegliere il VPC con sottoreti pubbliche e private selezionate per l'istanza database, come il **vpc-*identifier*** | tutorial-vpc creato in [Creazione di un VPC con sottoreti pubbliche e private](#).
  - Subnet (Sottorete):** selezionare una sottorete pubblica esistente, come la **subnet-*identifier*** | Tutorial public | us-west-2a creata in Creazione di un gruppo di sicurezza VPC per un server Web pubblico.
  - In Auto-assign Public IP: (Assegna automaticamente IP pubblico:),** scegliere **Enable (Abilita)**.

### Step 3: Configure Instance Details

Configure the instance to suit your requirements. You can launch multiple instances from the same AMI, request Spot instances to take advantage of the lower pricing, assign an access management role to the instance, and more.

Number of instances: 1 [Launch into Auto Scaling Group](#)

Purchasing option: Request Spot Instances

**Network:** vpc- | tutorial-vpc | Create new VPC  
**Subnet:** subnet- | Tutorial public | us-eas | Create new subnet  
**Auto-assign Public IP:** Enable

Placement group: Add instance to placement group

Capacity Reservation: Open

Domain join directory: No directory | Create new directory

IAM role: None | Create new IAM role

CPU options: Specify CPU options

Shutdown behavior: Stop

Stop - Hibernate behavior: Enable hibernation as an additional stop behavior

Enable termination protection: Protect against accidental termination

Monitoring: Enable CloudWatch detailed monitoring  
Additional charges apply.

Tenancy: Shared - Run a shared hardware instance  
Additional charges will apply for dedicated tenancy.

Elastic Inference: Add an Elastic Inference accelerator

[Cancel](#) [Previous](#) [Review and Launch](#) [Next: Add Storage](#)

6. Scegliere **Next: Add Storage (Successivo: Aggiungi storage)**.
7. Nella pagina **Add Storage (Aggiunta storage)**, mantenere i valori predefiniti e selezionare **Next: Add Tags (Fase successiva: aggiunta di tag)**.
8. Nella pagina **Add Tags (Aggiungi tag)**, visualizzata di seguito, scegliere **Add Tag (Aggiungi tag)** e quindi inserire **Name** per **Key (Chiave)** e inserire **tutorial-web-server** per **Value (Valore)**.

### Step 5: Add Tags

A tag consists of a case-sensitive key-value pair. For example, you could define a tag with key = Name and value = Webserver.

A copy of a tag can be applied to volumes, instances or both.

Tags will be applied to all instances and volumes. [Learn more](#) about tagging your Amazon EC2 resources.

Key	(127 characters maximum)	Value	(255 characters maximum)	Instances	Volumes
Name		tutorial-web-server		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
<a href="#">Add another tag</a> (Up to 50 tags maximum)					

[Cancel](#) [Previous](#) [Review and Launch](#) [Next: Configure Security Group](#)

9. Scegliere **Next: Configure Security Group (Fase successiva: configurazione del gruppo di sicurezza)**.
10. Nella pagina **Configura gruppo di sicurezza**, visualizzata di seguito, scegliere **Selezione un gruppo di sicurezza esistente**. Quindi scegliere un gruppo di sicurezza esistente, ad esempio tutorial-securitygroup creato in [Creazione di un gruppo di sicurezza VPC per un server Web pubblico](#). Assicurarsi che il gruppo di sicurezza scelto includa regole in ingresso per Secure Shell (SSH) e l'accesso HTTP.

## Step 6: Configure Security Group

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group or select from an existing one below. [Learn more](#) about Amazon EC2 security groups.

Assign a security group:  Create a new security group  
 Select an existing security group

Security Group ID	Name	Description	Actions
sg-[REDACTED]	default	default VPC security group	<a href="#">Copy to new</a>
sg-[REDACTED]	tutorial-db-securitygroup	Tutorial DB Instance Security Group	<a href="#">Copy to new</a>
<input checked="" type="checkbox"/> sg-[REDACTED]	tutorial-securitygroup	Tutorial Security Group	<a href="#">Copy to new</a>

Inbound rules for sg-0ef508f81f84a5764 (Selected security groups: sg-0ef508f81f84a5764)

Type	Protocol	Port Range	Source	Description
HTTP	TCP	80	[REDACTED]	
SSH	TCP	22	[REDACTED]	

[Cancel](#) [Previous](#) [Review and Launch](#)

11. Scegliere **Review and Launch (Analizza e avvia)**.

12. Nella pagina **Review Instance Launch (Verifica avvio istanza)**, visualizzata di seguito, verificare le impostazioni e poi selezionare **Launch (Avvia)**.

## Step 7: Review Instance Launch

 **Amazon Linux 2 AMI (HVM), SSD Volume Type - ami-0cf6f5c8a62fa5da6**

**Free tier eligible** Amazon Linux 2 comes with five years support. It provides Linux kernel 4.14 tuned for optimal performance on Amazon EC2, systemd 219, GCC 7.3, Glibc 2.26, Binutils 2.29.1, and the latest software packages through extras. This AMI is the successor of the Amazon Linux AMI that is a...

Root Device Type: ebs Virtualization type: hvm

**Instance Type** [Edit instance type](#)

Instance Type	ECUs	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance
t2.micro	-	1	1	EBS only	-	Low to Moderate

**Security Groups** [Edit security groups](#)

Security Group ID	Name	Description
sg-[REDACTED]	tutorial-securitygroup	Tutorial Security Group

All selected security groups inbound rules

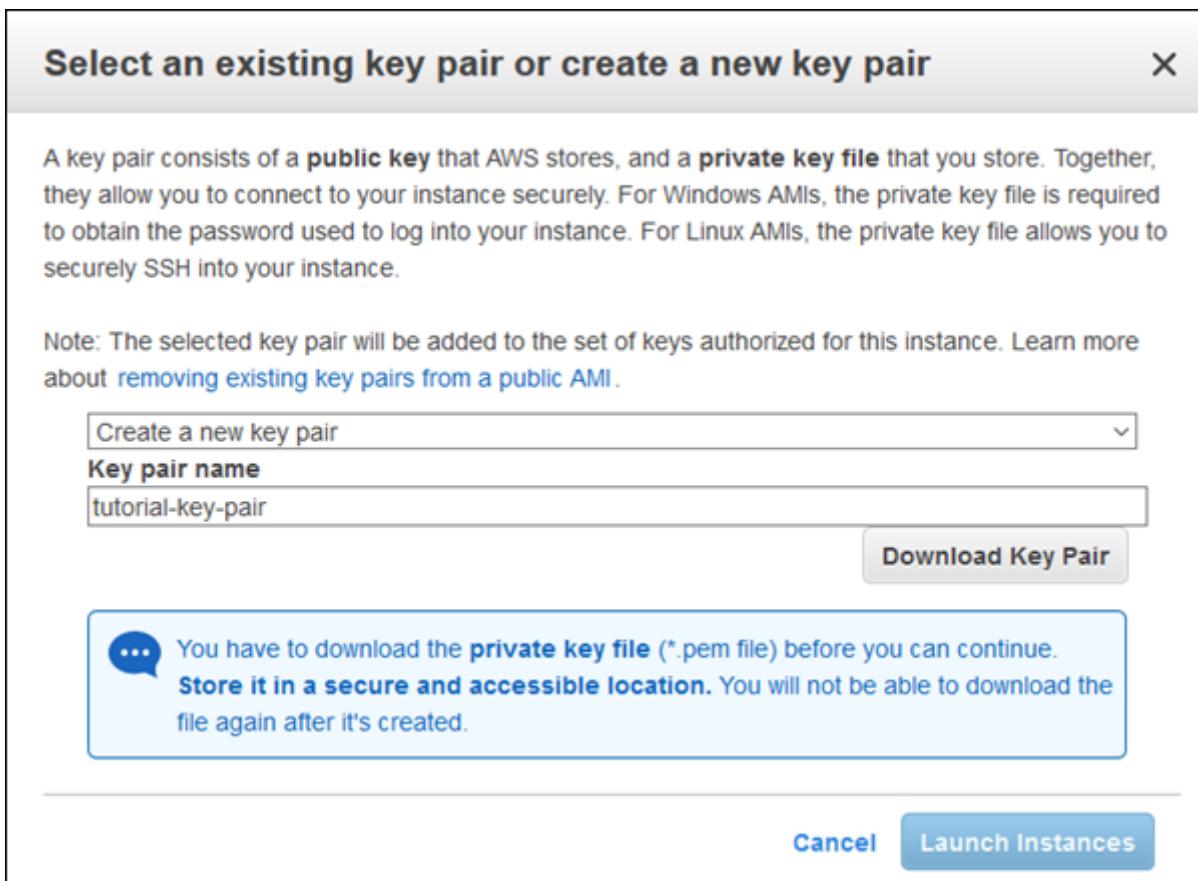
Type	Protocol	Port Range	Source	Description
HTTP	TCP	80	[REDACTED]	
SSH	TCP	22	[REDACTED]	

**Instance Details** [Edit instance details](#)

[Cancel](#) [Previous](#) [Launch](#)

13. Nella pagina **Select an existing key pair or create a new key pair (Seleziona una coppia di chiavi esistenti o crea una nuova coppia di chiavi)**, visualizzata di seguito, scegliere **Create a new key pair (Crea nuova coppia di chiavi)** e impostare **Key pair name (Nome coppia di chiavi)** su tutorial-key-pair. Selezionare **Download Key Pair**

(Scarica coppia di chiavi) e poi salvare il file della coppia di chiavi sulla macchina locale. Utilizzare questo file della coppia di chiavi per effettuare la connessione all'istanza EC2.



14. Per avviare l'istanza EC2, selezionare **Launch Instances (Avvia istanze)**. Nella pagina **Launch Status (Stato avvio)**, visualizzata di seguito, prendere nota dell'identificatore per la nuova istanza EC2, ad esempio: i-0288d65fd4470b6a9.

## Launch Status

✓ Your instances are now launching

The following instance launches have been initiated: [i-0288d65fd4470b6a9](#) [View launch log](#)

ⓘ Get notified of estimated charges

Create [billing alerts](#) to get an email notification when estimated charges on your AWS bill exceed an amount you define (for example, if you exceed the free usage tier).

### How to connect to your instances

Your instances are launching, and it may take a few minutes until they are in the **running** state, when they will be ready for you to use. Usage hours on your new instances will start immediately and continue to accrue until you stop or terminate your instances.

Click [View Instances](#) to monitor your instances' status. Once your instances are in the **running** state, you can **connect** to them from the Instances screen. [Find out](#) how to connect to your instances.

▼ Here are some helpful resources to get you started

- [How to connect to your Linux instance](#)
- [Learn about AWS Free Usage Tier](#)
- [Amazon EC2: User Guide](#)
- [Amazon EC2: Discussion Forum](#)

While your instances are launching you can also

- [Create status check alarms](#) to be notified when these instances fail status checks. (Additional charges may apply)
- [Create and attach additional EBS volumes](#) (Additional charges may apply)
- [Manage security groups](#)

[View Instances](#)

15. Per trovare l'istanza, selezionare **Vedi istanze**.

16. Attendere che lo **Stato istanza** dell'istanza sia in **esecuzione** prima di continuare.

### 2.3 Connettersi ad un'istanza EC2 Linux, [1]

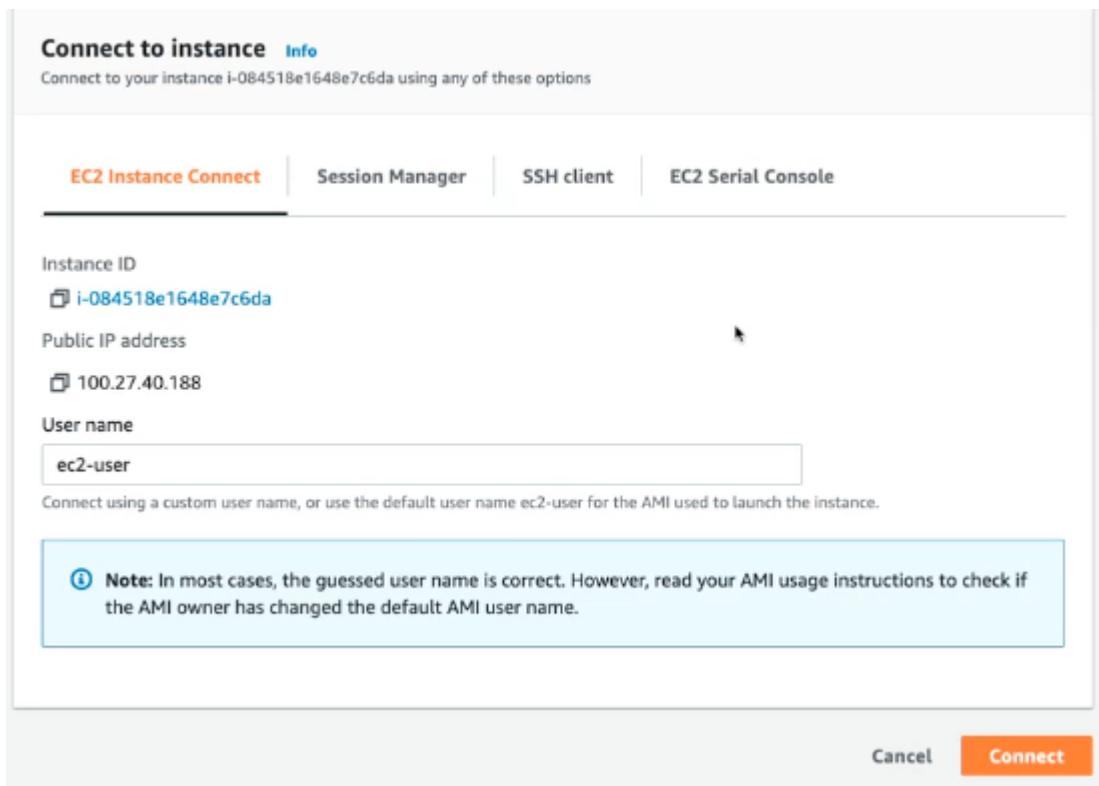
Creata un'istanza EC2, sarà possibile collegarsi ad essa da remoto.

1. Selezionare l'istanza e premere **Connetti**.

Istanze (1/2) <a href="#">Informazioni</a>						
<input type="text"/> Cerca		<a href="#">C</a> <a href="#">Connetti</a>				
Stato dell'istanza = running		Annulla filtri				
Name	ID istanza	Stato dell'ista...	Tipo di istanza	Verifica dello stato	Stato dell'all...	Avail
<input checked="" type="checkbox"/> 1	i-0dbb781fc86eb11c2	<input checked="" type="checkbox"/> In esecuzione	t2.micro	<input checked="" type="checkbox"/> 2/2 controlli superati	Nessun al...	+ us-ea
<input checked="" type="checkbox"/>	i-04072b73e28a77b93	<input checked="" type="checkbox"/> In esecuzione	t2.micro	<input checked="" type="checkbox"/> 2/2 controlli superati	Nessun al...	+ us-ea

2. Si aprira' un menu' di selezione della modalita' di accesso. Ne vedremo 2: **EC2 Instance Connect** e **SSH Client**.

- a. Il primo, EC2 Instance Connect, apre una finestra nel browser che simula il terminale dell'istanza;



```
Amazon Linux 2 AMI
https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-172-31-9-11 ~]$ ls
[ec2-user@ip-172-31-9-11 ~]$ ls -a
.. .bash_logout .bash_profile .bashrc .ssh
[ec2-user@ip-172-31-9-11 ~]$ ls .ssh/
authorized_keys
[ec2-user@ip-172-31-9-11 ~]$ cat .ssh/authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQCNBLWctyeYERLxhAvsaJVe0QcALmqucuw/a94gwDlnAYrfaKsHuUK+B2YoD78XEEQJut+7h/LITkL+dKdn4WMJjd2Qh1BzMxDXWUJMILfqL1ax
XSNSYaqT/0ZZ3Xr75Fs7yc17+PT0ghguLgU2LHUtltznEcEaDexbMrDsKzea9grHi rm3WcYLpfsjgaqSgfni0Vh94SgL2BWhFRqHoS8fnIV9exd77/H01uUyXkPsyauWsx72bLb+2GxEKmYfL
g0IA6+x7Ha/72XvqsFW3nBlN8e1l+6/Vs6h6VhP6EjNNwTvLteFmK3TDk1H7sU19UemU3k/0Yrv8G9qUw03I3j 16-marzo-2022
[ec2-user@ip-172-31-9-11 ~]$
```

*Immagine che mostra la chiave pubblica memorizzata nell'istanza EC2.*

- b. Il secondo, SSH Client, prevede l'inserimento del file contenente la chiave privata precedentemente creata (in rosso), nome utente (in blu) e il nome della macchina (in verde). Per il nome della macchina è possibile usare il nome dns oppure l'indirizzo ip pubblico.

## Connettiti all'istanza Informazioni

Connettiti all'istanza i-04072b73e28a77b93 utilizzando una qualsiasi di queste opzioni

[EC2 Instance Connect](#)

[Gestore di sessioni](#)

[Client SSH](#)

[Console seriale EC2](#)

ID istanza

 [i-04072b73e28a77b93](#)

1. Aprire un client SSH.
2. Individua il file della chiave privata. La chiave utilizzata per avviare questa istanza è prova.pem
3. Esegui questo comando, se necessario, per far sì che la chiave non sia visualizzabile pubblicamente.  
 `chmod 400 prova.pem`
4. Connottiti alla tua istanza usando il suo DNS pubblico:  
 `ec2-18-212-164-156.compute-1.amazonaws.com`

Esempio:

 `ssh -i "prova.pem" ubuntu@ec2-18-212-164-156.compute-1.amazonaws.com`

 **Nota:** nella maggior parte dei casi, il nome utente ipotizzato è corretto. Tuttavia, leggi le istruzioni di utilizzo dell'AMI per verificare che il relativo proprietario non abbia modificato il nome utente predefinito.

3. A questo punto sembra chiaro che dobbiamo svolgere un'azione preliminare prima di poterci connettere all'istanza. Andremo quindi a modificare i permessi alla nostra chiave privata.

Ricordiamo che `Chmod 400 (chmod a+rwx, u-wx, g-rwx, o-rwx)` è il permesso sul file del proprietario in sola lettura:

### Permessi file

**chmod octal file** – cambia i permessi di *file* a *octal*, numero di 3 cifre, rispettivamente per l'utente, il gruppo e tutti gli altri, somme di:

- 4 – lettura (r)
- 2 – scrittura (w)
- 1 – esecuzione o visita (x)

Esempi:

**chmod 777** – lettura, scrittura ed esecuzione per tutti

**chmod 755** – rwx per il proprietario, rx per il gruppo e tutti gli altri

Per ulteriori dettagli si esegua `man chmod`.

Una volta scaricato ed installato, portiamoci nella cartella dove abbiamo salvato il file `.pem`

Digitare il seguente comando:

`chmod 400 prova.pem`

Prima di lanciare il comando è possibile controllare i permessi attuali del file tramite il seguente comando:

`ls -la`

Vediamo la seguente immagine per verificare i permessi come erano impostati prima dell'operazione e poi lo ripetiamo per verificare che l'operazione è stata eseguita correttamente:

```

$ ls -la
total 28
drwxr-xr-x 1 Buje 197121    0 Dec 22 21:59 .
drwxr-xr-x 1 Buje 197121    0 Dec 22 21:58 ..
-rw-r--r-- 1 Buje 197121 1700 Dec 22 20:04 prova.pem

          MINGW64 ~/Desktop/prova
$ chmod 400 prova.pem

          MINGW64 ~/Desktop/prova
$ ls -la
total 28
drwxr-xr-x 1 Buje 197121    0 Dec 22 21:59 .
drwxr-xr-x 1 Buje 197121    0 Dec 22 21:58 ..
-rw-r--r-- 1 Buje 197121 1700 Dec 22 20:04 prova.pem

```

Come è possibile controllare l'operazione è stata svolta correttamente.

4. A questo punto siamo pronti per lanciare il comando successivo:

```
ssh -i "prova.pem" ubuntu@ec2-18-212-164-156.compute-1.amazonaws.com
```

### 2.3.1 Creazione manuale della coppia di chiavi

È inoltre possibile creare in locale la chiave privata e pubblica e passare SOLO la chiave pubblica ad aws dall'apposito menu. Ecco i passaggi:

1. Le chiavi si creano sfruttando il comando:

```
ssh-keygen -t rsa
```

Successivamente si inserirà il *key pair name*, ed eventualmente una *passphrase* per proteggere il file (nell'esempio lasciato vuoto).

```
pierpaolo@Pierpaolos-MacBook ~ % ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/Users/pierpaolo/.ssh/id_rsa): aws-18-mar
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in aws-18-mar.
Your public key has been saved in aws-18-mar.pub.
The key fingerprint is:
SHA256:8ktZuEdW0lIfT4ip3AV0o6BMkNhp5tuXyIQU6KstSH0 pierpaolo@Pierpaolos-MacBook.local
The key's randomart image is:
+---[RSA 3072]---+
| +0=. . =+... |
| o Bo . . *ooo+ |
| . = .o .+o+... . |
| . o . .o+. |
| .. =..S.+ |
| ....E+oo* |
| oo . .= . |
| + . . o |
| . . |
+---[SHA256]---+
```

2. Procediamo a caricare su aws la chiave pubblica. Dal pannello di controllo EC2 laterale selezionare la voce **copie di chiavi**:

## Pannello di controllo EC2

Visualizzazione di EC2

Global

Eventi

Tag

Limiti

► Istanze

► Immagini

► Elastic Block Store

▼ Rete e sicurezza

Gruppi di sicurezza

IP elastici

Gruppi di collocamento

Copie di chiavi

Interfacce di rete

► Bilanciamento del carico

3. Selezionare quindi il menù a tendina **Operazioni** e successivamente **Importa una coppia di chiavi**:

The screenshot shows the AWS EC2 Key Pairs page. At the top, there's a header with 'Coppie di chiavi (24)' and a 'Informazioni' button. Below is a search bar labeled 'Filtro coppie di chiavi'. A table lists two key pairs: 'Nome' (16-marzo-2022), 'Tipo' (rsa), 'Impronta digitale' (3f:ec:68:5e:10:b0:a8:22:3c:a2:83:7c:46...), and 'ID' (key-0a7275e89f8d9c01d). To the right of the table is an 'Operazioni' dropdown menu with the following options: 'Importa una coppia di chiavi' (highlighted with a red box and circled with red number 1), 'Elimina', and 'Gestisci tag'. Red arrows point from the circled number 1 to the 'Importa una coppia di chiavi' option and from the circled number 2 to the 'Operazioni' button.

4. Dalla finestra Importa una coppia di chiavi è possibile:

- Inserire il nome della coppia di chiavi;
- Caricare il file contenente SOLO la chiave pubblica;
- Oppure copiare e incollare (e modificare all'occorrenza) la chiave pubblica direttamente dall'apposito campo;
- Aggiungere un Tag (facoltativo).

Fatto ciò, premere su **Importa una coppia di chiavi**.

# Importa una coppia di chiavi

## Impostazioni di importazione

Nome

Inserisci il nome della coppia di chiavi

1

Il nome può includere fino a 255 caratteri ASCII. Non può includere spazi iniziali o finali.

File della coppia di chiavi

Sfoglia

2

Scegli Sfoglia e vai alla chiave pubblica. Se necessario, puoi cambiare il nome della tua chiave. Altrimenti, puoi incollare i contenuti della tua chiave pubblica nella casella di testo Contenuti chiave pubblica.

3

Tag (facoltativo)

Nessun tag associato alla risorsa

Aggiungi tag

4

Puoi aggiungere altri 50 tag.

5

Annulla

Importa una coppia di chiavi

**Nota:** Le chiavi pubbliche delle istanze EC2 sono salvate nel file **authorized\_keys** raggiungibile tramite:

```
cat .ssh/authorized_keys
```

5. Per accedere all'istanza è possibile seguire gli stessi passaggi descritti nel paragrafo precedente.

## 2.4 Tagging delle risorse Amazon EC2, [1]

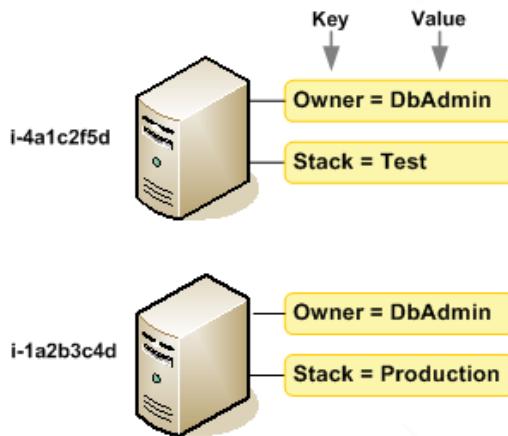
Per semplificare la gestione di istanze, immagini e altre risorse Amazon EC2 puoi decidere di assegnare metadati personalizzati a ogni risorsa sotto forma di **tag**. I tag consentono di categorizzare le tue risorse AWS in modi diversi, ad esempio, per scopo, proprietario o ambiente. Questa caratteristica è molto utile quando hai tante risorse dello stesso tipo in quanto puoi rapidamente individuare una risorsa specifica in base ai tag assegnati. Questo argomento descrive i tag e mostra come crearli.

### Nozioni di base sui tag

Un tag è un'etichetta che si assegna a una risorsa AWS. Ogni tag è composto da una chiave e da un valore opzionale, entrambi personalizzabili.

I tag consentono di categorizzare le risorse AWS in modi diversi, ad esempio, per scopo, proprietario o ambiente. Ad esempio, è possibile definire un set di tag per le istanze Amazon EC2 dell'account e monitorare così ogni proprietario dell'istanza e il livello dello stack.

Lo schema seguente illustra il funzionamento del tagging. In questo esempio si è assegnato due tag a ciascuna istanza, un tag con la chiave *Owner* e un altro tag con la chiave *Stack*. A ogni tag è inoltre associato un valore.



I tag non hanno alcun significato semantico per Amazon EC2 e vengono interpretati rigorosamente come una stringa di caratteri. Inoltre, i tag non vengono assegnati automaticamente alle risorse. È possibile modificare chiavi e valori di tag e rimuovere tag da una risorsa in qualsiasi momento. È possibile impostare il valore di un tag su una stringa vuota, ma non su *null*. Se aggiungi un tag con la stessa chiave di un tag esistente a una risorsa specifica, il nuovo valore sovrascrive quello precedente. Se elimini una risorsa, verranno eliminati anche tutti i tag associati alla risorsa.

È possibile assegnare tag alla maggior parte delle risorse Amazon EC2 già esistenti nell'account. In questa [tabella](#) sono elencate le risorse che supportano il tagging.

**Nota:** È possibile associare tag a istanze e volumi in fase di creazione utilizzando la procedura guidata per l'avvio di istanze Amazon EC2 nella console Amazon EC2.

#### 2.4.1 Creazione di un Tag

È possibile creare un tag selezionando un'istanza e spostarsi nel menu **Tag**. Premere su **Gestisci tag** per creare uno o più tag:

Name	ID Istanza	Stato dell'istanza	Tipo di istanza	Verifica dello stato	Stato dell'all.	Aval
<input checked="" type="checkbox"/>	i-0384e5a5838eac073	In esecuzione	t2.micro	Initiazlizzazione in	Nessun al...	+

Istanza: i-0384e5a5838eac073

Selezione un'istanza qui sopra

Dettagli Sicurezza Reti Storage Verifiche di stato Monitoraggio in corso **Tag**

**Tag**

Gestisci tag

Nessun tag trovato

Gestisci tag

Si aprirà quindi la seguente schermata:

**Gestisci tag** [Informazioni](#)

Un tag è un'etichetta personalizzata che assegna a una risorsa AWS. Puoi utilizzare i tag per organizzare e identificare le istanze.

Chiave	Valore - facoltativo
<input type="text" value="Inserisci chiave"/>	<input type="text" value="Inserisci valore"/>
Chiave di tag personalizzata	
<b>Aggiungi tag</b>	
Puoi aggiungere altri 49 tag.	
<b>Annulla</b> <b>Salva</b>	

Una volta creati i tag è possibile visualizzarli e gestirli dall'apposito pannello laterale:

New EC2 Experience [Tell us what you think](#)

Pannello di controllo EC2

Visualizzazione di EC2 Global

Eventi

**Tag**

Limiti

Istanze

Immagini

Elastic Block Store

Rete e sicurezza

Bilanciamento del carico

Auto Scaling

**Gestisci tag**

Filtra:	Cerca chiavi	Cerca valori	da 1 a 9 di 9					
	Chiave tag	Valore tag	Totali	Istanze	AMI	Volumi	Snapshot	
Gestisci tag	aws:cloud9:environment	54865ad49d54476b9ec4e22c74a0697a	1	0	0	0	0	C
Gestisci tag	aws:cloud9:owner	AIDAT7AMEIY06NJYLVZRX	1	0	0	0	0	C
Gestisci tag	aws:cloudformation:logical-id	InstanceSecurityGroup	1	0	0	0	0	C
Gestisci tag	aws:cloudformation:stack-id	arn:aws:cloudformation:us-east-1:27...	1	0	0	0	0	C
Gestisci tag	aws:cloudformation:stack-na...	aws-cloud9-IoTOnAWS-54865ad49d...	1	0	0	0	0	C
Gestisci tag	for-use-with-amazon-emr-m...	true	2	0	0	0	0	C
Gestisci tag	ManagedByAmazonSageMa...	arn:aws:sagemaker:us-east-1:27275...	2	0	0	0	0	C
Gestisci tag	ManagedByAmazonSageMa...	arn:aws:sagemaker:us-east-1:27275...	2	0	0	0	0	C
Gestisci tag	Name	aws-cloud9-IoTOnAWS-54865ad49d...	1	0	0	0	0	C

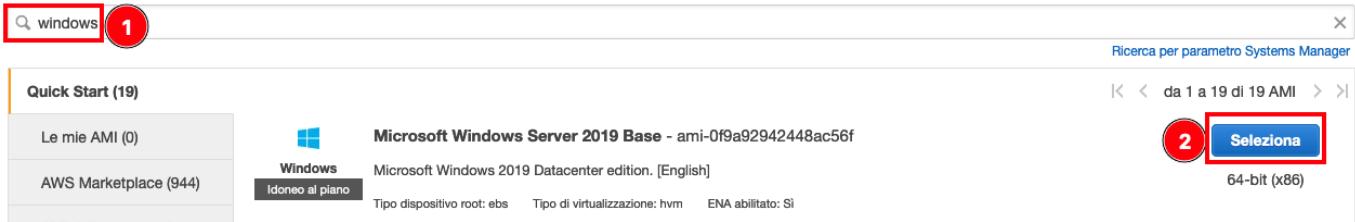
## 2.5 Istanze Windows, [1]

È possibile connettersi alle istanze Amazon EC2 create dalla maggior parte delle Amazon Machine Image (AMI) per Windows tramite Desktop remoto. Desktop remoto utilizza il protocollo [RDP \(Remote Desktop Protocol\)](#) per connettersi e utilizzare l'istanza con le stesse procedure usate per un computer vero e proprio. È disponibile per la maggior parte delle versioni di Windows e anche per Mac OS.

La licenza per il sistema operativo di Windows Server consente due connessioni remote simultanee per attività amministrative. Il costo della licenza per Windows Server è incluso nel costo della tua istanza Windows. Se servono più di due connessioni remote simultanee, devi acquistare una licenza di Remote Desktop Services (RDS). Se tenti di stabilire una terza connessione, si verifica un errore.

### 2.5.1 Creazione di un'istanza Windows

1. Dal menu di avvio di una nuova istanza, spostarsi nel campo di ricerca e digitare **windows**. Verranno mostrate tutte le AMI disponibili, premere quindi su **Seleziona**.



2. Selezionare una macchina in base alle risorse di CPU, memoria, storage e capacità di rete che si necessitano. Premere quindi su **Analizza e avvia**.

Filtra per: Tutte le famiglie di istanze Generazione attuale Mostra/nascondi colonne

Attualmente selezionato: t2.micro (- ECU, 1 vCPU, 2.5 GHz, -, 1 GiB memoria, Solo EBS)

	Famiglia	Tipo	vCPU	Memoria (GiB)	Storage istanza (GB)	Disponibile ottimizzato EBS	Prestazioni di rete	Supporto per il protocollo IPv6
<input type="checkbox"/>	t2	t2.nano	1	0.5	Solo EBS	-	Da basso a moderato	Si
<input checked="" type="checkbox"/>	t2	t2.micro Idoneo al piano gratuito	1	1	Solo EBS	-	Da basso a moderato	Si
<input type="checkbox"/>	t2	t2.small	1	2	Solo EBS	-	Da basso a moderato	Si
<input type="checkbox"/>	t2	t2.medium	2	4	Solo EBS	-	Da basso a moderato	Si
<input type="checkbox"/>	t2	t2.large	2	8	Solo EBS	-	Da basso a moderato	Si

Annulla Precedente Analizza e avvia Successivo: Configura i dettagli dell'istanza

3. Verificare che il gruppo di sicurezza associato all'istanza consenta il traffico RDP (port 3389) in entrata dall'indirizzo IP. Per impostazione predefinita, il gruppo di sicurezza predefinito non consente il traffico RDP in entrata. Premere quindi su **Lancio**.

▶ Dettagli AMI [Modifica AMI](#)

▶ Tipo di istanza [Modifica tipo di istanza](#)

▼ Gruppi di sicurezza [Modifica i gruppi di sicurezza](#)

Nome del gruppo di sicurezza: launch-wizard-46  
Descrizione: launch-wizard-46 created 2022-04-10T17:33:32.777+02:00

Tipo	Protocollo	Intervallo porte	Origine	Descrizione
RDP	TCP	3389	0.0.0.0/0	

▶ Dettagli istanza [Modifica dettagli istanza](#)

▶ Storage [Modifica storage](#)

▶ Tag [Modifica tag](#)

Annulla Precedente **Lancio**

4. Selezionare una coppia di chiavi esistente oppure creare una nuova coppia di chiavi. Premere quindi su **Avvia le istanze**.

## Seleziona una coppia di chiavi esistente oppure crea una nuova coppia di chiavi

Una coppia di chiavi è costituita da una **chiave pubblica** che AWS archivia, e da un **file di una chiave privata** che tu archivi. Insieme ti consentono di connetterti all'istanza in modo sicuro. Per le AMI di Windows, il file della chiave privata è necessario per ottenere la password di accesso alla tua istanza. Per le AMI di Linux, il file della chiave privata consente un SSH in sicurezza alla tua istanza. Amazon EC2 supporta i tipi di coppie di chiavi ED25519 e RSA.

Nota: la coppia di chiavi selezionata sarà aggiunta al set di chiavi autorizzate per questa istanza.

Ulteriori informazioni in [Eliminare coppie di chiavi esistenti da un'AMI pubblica](#).

1 Scegli una coppia di chiavi esistente

2 16-marzo-2022 | RSA

3  Sono consapevole di avere accesso al file della chiave privata corrispondente e che senza questo file non potrei accedere alla mia istanza.

4

**Avvia le istanze**

Nota: La coppia di chiavi appena creata serve a decriptare la password dell'istanza Windows. Si usa quindi la chiave per accedere alla password di Windows (vedere successivamente).

### 2.5.2 Connessione all'istanza Windows con il protocollo RDP

Per connettersi ad un'istanza di Windows, è necessario recuperare la password iniziale dell'amministratore e immetterla quando ci si connette all'istanza tramite Desktop remoto. Dopo l'avvio dell'istanza, bisogna attendere alcuni minuti prima che la password sia disponibile. Il nome dell'account amministratore dipende dalla lingua del sistema operativo. Per esempio, per l'inglese è *Administrator*, per il francese è *Administrateur* e per il portoghese è *Administrador*.

Per connetterti all'istanza Windows utilizzando un client RDP:

1. Aprire la console Amazon EC2. Nel pannello di navigazione, scegliere **Instances (Istanze)**. Selezionare l'istanza quindi scegliere **Connect (Connetti)**.

Istanze (1/2) <a href="#">Informazioni</a>			
<input type="text"/> Cerca		<a href="#">Annulla filtri</a>	
Stato dell'istanza = running	X		
Name	ID istanza	Stato dell'istanza	Tipo di istanza
<input type="checkbox"/>	i-0dbb781fc86eb11c2	In esecuzione	t2.micro
<input checked="" type="checkbox"/> 1	i-04072b73e28a77b93	In esecuzione	t2.micro

2. Nella pagina **Connect to instance (Connettiti all'istanza)**, scegliere la scheda **RDP client (Client RDP)** e quindi scegliere **Get password (Ottieni password)**.

**Connect to instance** [Info](#)

Connect to your instance i-XXXXXXXXXX) using any of these options

Session Manager [RDP client](#) [EC2 Serial Console](#)

You can connect to your Windows instance using a remote desktop client of your choice, and by downloading and running the RDP shortcut file below:

[Download remote desktop file](#)

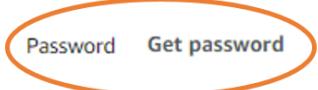
When prompted, connect to your instance using the following details:

Public DNS	User name
<input type="text"/> ec2-100-25-146-1.compute-1.amazonaws.com	<input type="text"/> Administrator

**>Password** [Get password](#)

If you've joined your instance to a directory, you can use your directory credentials to connect to your instance.

[Cancel](#)



3. Scegliere **Browse (Sfoglia)** e selezionare il file della chiave privata (.pem) creato al momento dell'avvio dell'istanza. Selezionare il file e scegliere **Open (Apri)** per copiare l'intero contenuto del file in questa finestra. Selezionare quindi **Decrypt Password (Decifra password)**.

**Get Windows password** [Info](#)

Retrieve and decrypt the initial Windows administrator password for this instance.

To decrypt the password, you will need your key pair for this instance.

**Key pair associated with this instance**  
16-marzo-2022

**1** Browse to your key pair:  
 16-marzo-2022.pem  
1.7KB

Or copy and paste the contents of the key pair below:

```
-----BEGIN RSA PRIVATE KEY-----
MIIEowIBAAKCAQEApwZVnLcnmBES8YQL7GiVXjkHAC5qmLsP2veiMA5ZwGK32Ir
B7ICvgdmKA+/Fx8ECbrfu4fyE5C/nSnZ+FjCY3dkizQczA11lCTCC36i9WsVOJu
mGtE/9GW5V6++RbO2HNe/J09lYL14FNlx1L57c5xj/BGg5sWzKKw7Cs3mvYKx4q5
t1nGT37l4Gqkoh59afR/eEc9gVmhxUah6EvH5yFxsXe+/xztbIMSD7MnlFrM
e9m5W/thsRCpmHSYNCAOvsex2v+9l76r8Vt5wZTffHz2fuv1bOoeLYT+hizTcE7y7
XhZl0w5NR+TFNfVHpIn5Pzmk7/BvaIMDty4wIDAQABaoIBAF/QHZ5+FswCDebk
A98DA65AfEabWfFm1aVnIIWTH5q6ZMr5BUKyANp57h9PISR+3TJjrZKrYxeKS2I
```

**2** [Decrypt password](#)

[Cancel](#)

4. In **Password**, la console visualizzerà la password di amministratore predefinita per l'istanza, che sostituisce il collegamento **Get Password (Ottieni password)** mostrato in precedenza. **Conservare la password in un luogo sicuro**. Questa password servirà per

connettersi all'istanza. Seleziona **Download remote desktop file** (Scarica file per desktop remoto).

**Connect to instance** [Info](#)

Connect to your instance i-██████████ using any of these options

Session Manager | **RDP client** | EC2 Serial Console

**⚠** You may not be able to connect to this instance as ports 3389 may need to be open in order to be accessible. The current associated security groups don't have ports 3389 open. **X**

You can connect to your Windows instance using a remote desktop client of your choice, and by downloading and running the RDP shortcut file below:

[Download remote desktop file](#)

When prompted, connect to your instance using the following details:

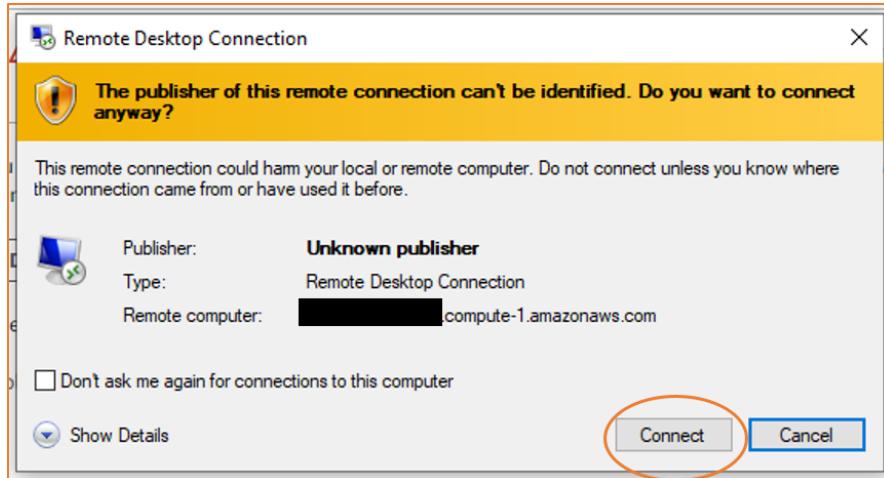
Public DNS	User name
<input type="text"/> i-██████████.compute-1.amazonaws.com	<input type="text"/> Administrator
<b>Password</b>	
<input type="password"/> ██████████	

If you've joined your instance to a directory, you can use your directory credentials to connect to your instance.

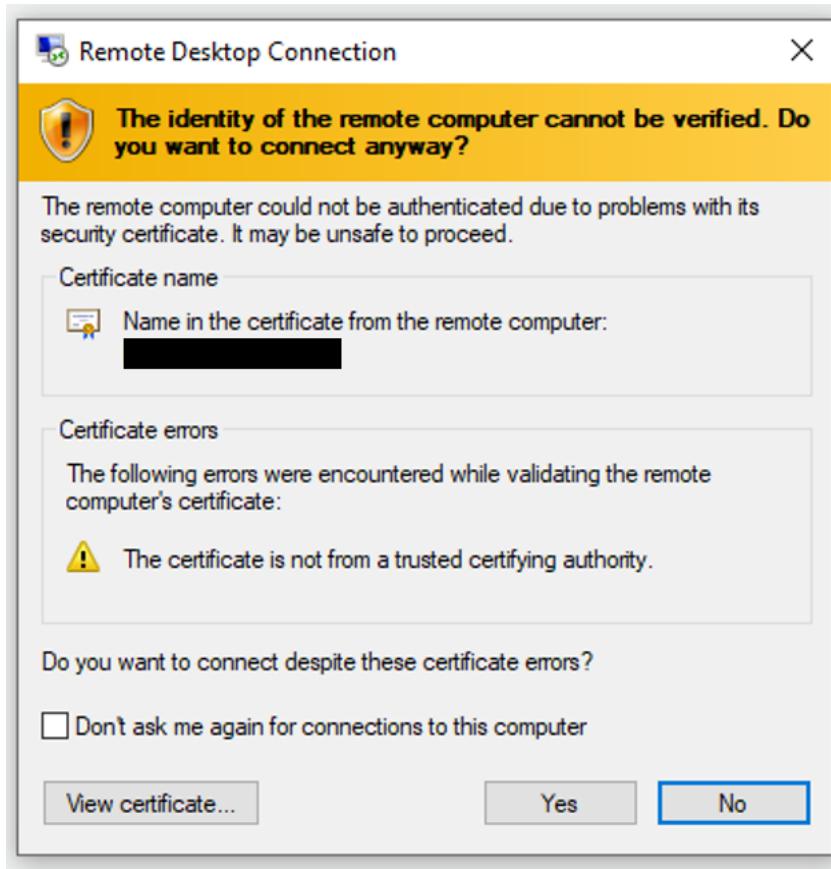
**Cancel**

The screenshot shows a 'Connect to instance' dialog box. It has tabs for 'Session Manager', 'RDP client' (which is selected), and 'EC2 Serial Console'. Below the tabs, a warning message says: '⚠ You may not be able to connect to this instance as ports 3389 may need to be open in order to be accessible. The current associated security groups don't have ports 3389 open.' A close button 'X' is next to the message. The main area says: 'You can connect to your Windows instance using a remote desktop client of your choice, and by downloading and running the RDP shortcut file below:' followed by a 'Download remote desktop file' button. Below that, it says: 'When prompted, connect to your instance using the following details:' and shows fields for 'Public DNS' (with value 'i-██████████.compute-1.amazonaws.com'), 'User name' (with value 'Administrator'), and 'Password' (with value redacted). An orange oval highlights the 'Password' field and its redaction. At the bottom, it says: 'If you've joined your instance to a directory, you can use your directory credentials to connect to your instance.' and has a 'Cancel' button.

5. Il browser visualizza un messaggio che chiede se si desidera aprire o salvare il file di collegamento RDP. Al termine del download del file, scegliere **Cancel (Annulla)** per tornare alla pagina **Instances (Istanze)**.
  - Se hai aperto il file RDP, visualizzerai la finestra di dialogo **connessione Desktop remoto**.
  - Se hai salvato il file RDP, puoi navigare nella directory dei download e aprire il file RDP per visualizzare la finestra di dialogo.
6. È possibile visualizzare un avviso che informa che non è nota l'identità di chi ha pubblicato la connessione remota. Scegliere **Connect (Connetti)** per collegarti all'istanza.



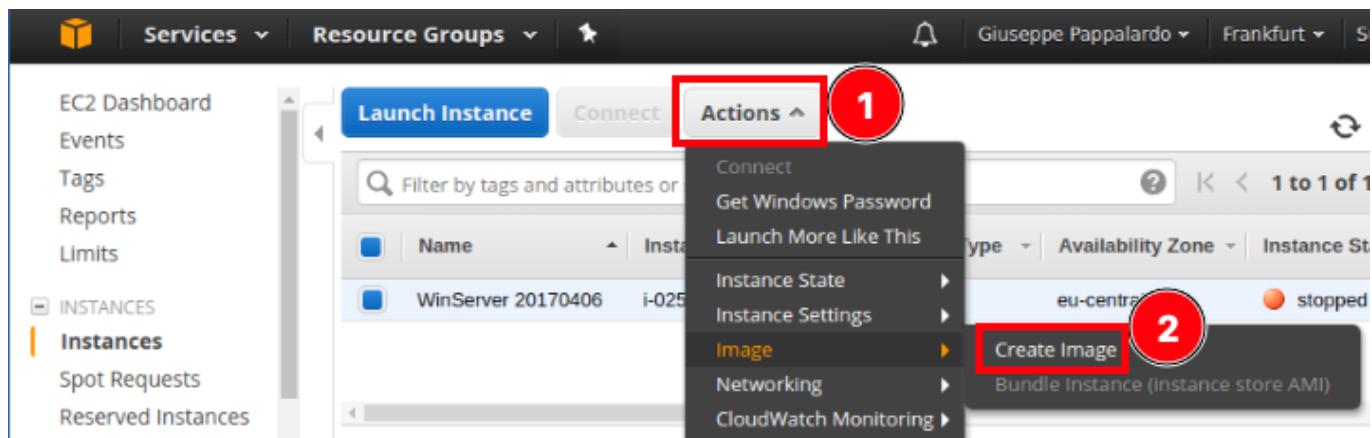
7. Per impostazione predefinita è selezionato l'account amministratore. Copiare e incollare la password salvata in precedenza.  
**Nota:** Se viene visualizzato un errore di password non corretta, provare a immettere la password manualmente. A volte copiare e incollare i contenuti può danneggiare i dati.
8. Data la natura dei certificati autofirmati, è possibile che venga visualizzato un avviso relativo all'impossibilità di autenticare il certificato di sicurezza. Procedere come descritto di seguito per verificare l'identità del computer remoto oppure scegliere **Yes (Sì)** (Windows) o **Continue (Continua)** (Mac OS X) per continuare se si ritiene attendibile il certificato.



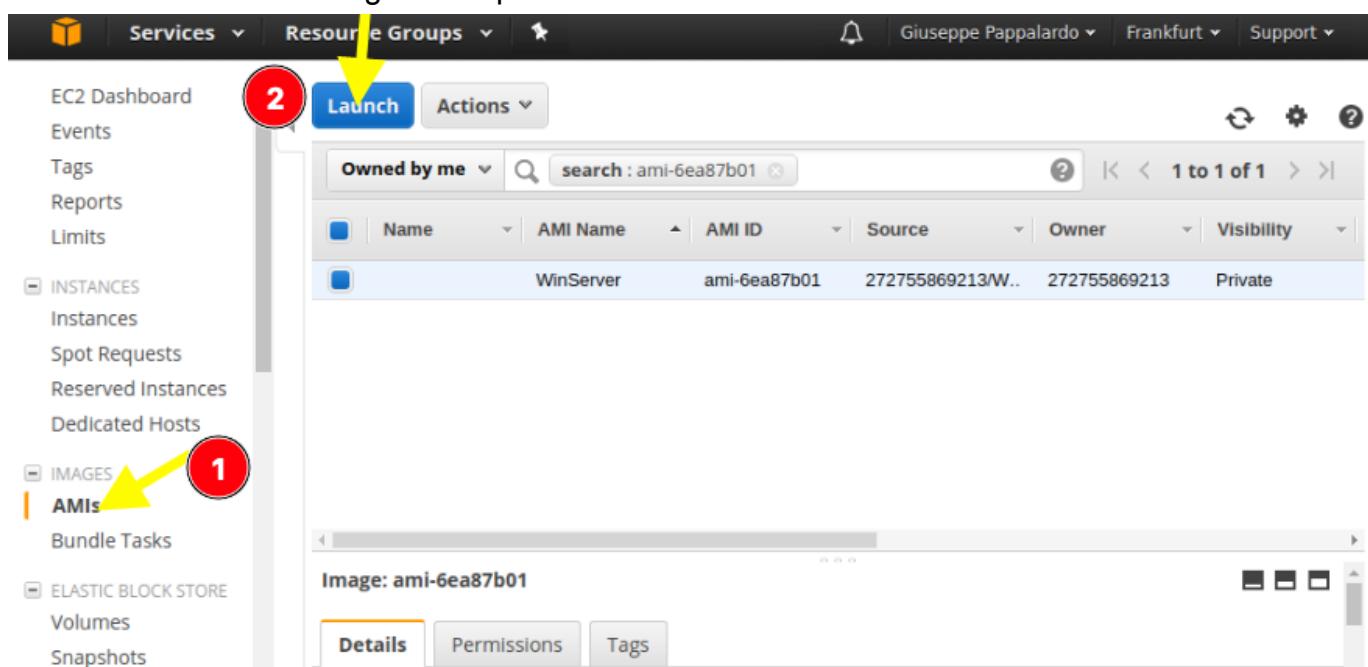
## 2.6 Perdita delle chiavi dell'istanza

La soluzione è creare l'immagine AMI dell'istanza (mantenerla memorizzata ha dei costi), successivamente terminare l'istanza, infine ripristinare l'istanza dell'immagine AMI (verrà chiesto di indicare una nuova coppia di chiavi e verrà rigenerata la password).

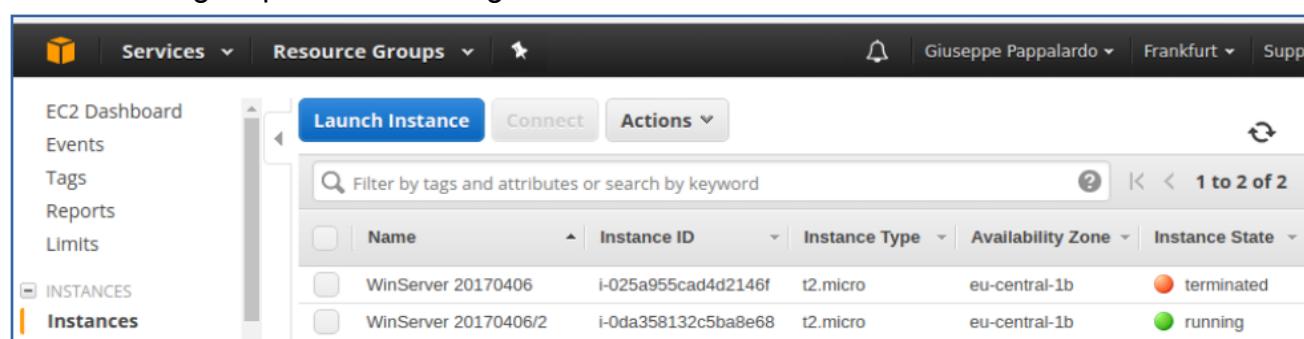
- Si crea l'immagine AMI dell'istanza:



- Una volta stoppata, è possibile recarsi nel pannello laterale e selezionare AMIs e selezionare l'immagine da ripristinare:



- Durante il ripristino dell'istanza dall'immagine, verrà riproposta la creazione di una nuova coppia di chiavi (rimediando così all'eventuale perdita della coppia originaria).
- Immagine ripristinata! Ora, conviene cancellare l'immagine AMI (dal servizio AMI, Actions/Deregister e attendere un po'). Conviene cancellare il volume che ospita l'immagine per evitare billing!



### 3. S3 (Amazon Simple Storage Service), [1]

Amazon Simple Storage Service (Amazon S3) è un servizio di archiviazione di oggetti che offre scalabilità, disponibilità dei dati, sicurezza e prestazioni tra le migliori del settore. I clienti di tutte le dimensioni e settori possono utilizzare Amazon S3 per archiviare e proteggere qualsiasi quantità di dati in un'ampia gamma di casi d'uso, come data lake, siti Web, applicazioni mobili, backup e ripristino, archivi, applicazioni per aziende, dispositivi IoT e analisi dei Big Data. Amazon S3 offre caratteristiche di gestione che consentono di ottimizzare, organizzare e configurare l'accesso ai dati per soddisfare specifici requisiti aziendali, organizzativi e di conformità.

The screenshot shows the AWS Management Console interface. On the left, there's a navigation sidebar with various categories like 'Gestione e governance', 'Integrazione di applicazioni', etc., and a 'Storage' category highlighted with a red box and a red circle containing the number '2'. Below 'Storage', another 'Storage' category is also highlighted with a red box and a red circle containing the number '3'. The main content area is titled 'Storage' and lists several services: 'AWS Backup', 'EFS', 'FSx', 'Ripristino di emergenza elastico AWS', 'S3' (which is also highlighted with a red box and a red circle containing the number '3'), 'S3 Glacier', and 'Storage Gateway'. The 'S3' service is described as 'Storage scalabile nel cloud'.

## 3.1 Caratteristiche di Amazon S3

### 3.1.1 Classi di archiviazione

Amazon S3 offre una gamma di classi di storage concepite per i diversi casi d'uso. Ad esempio, è possibile archiviare dati di produzione essenziali su S3 Standard per accedervi più spesso, mentre è possibile risparmiare sui costi archiviando i dati a cui si accede raramente in S3 Standard-IA o S3 One Zone-IA e archiviare i dati al costo più basso in S3 Glacier Flexible Retrieval e S3 Glacier Deep Archive.

### 3.1.2 Gestione dello storage

Amazon S3 dispone di caratteristiche di gestione dell'archiviazione utilizzabili per gestire i costi, rispettare i requisiti normativi, ridurre la latenza e salvare più copie distinte dei dati per soddisfare i requisiti di conformità.

- **Ciclo di vita S3:** consente di configurare policy relative al ciclo di vita per gestire gli oggetti e archiviarli all'insegna dell'efficienza in termini di costi durante l'intero ciclo di

vita. Puoi spostare gli oggetti in altre classi di archiviazione S3 o far scadere oggetti che raggiungono la fine del loro ciclo.

- **Blocco degli oggetti S3:** impedisce che un oggetto di Amazon S3 venga eliminato o sovrascritto per un determinato periodo di tempo o in modo indefinito. Il blocco degli oggetti consente di soddisfare i requisiti normativi che richiedono l'archiviazione write-once-read-many (WORM) o semplicemente di aggiungere un ulteriore livello di protezione contro la modifica e l'eliminazione degli oggetti.
- **Replica S3:** consente di replicare gli oggetti e i rispettivi metadati e tag oggetto in uno o più bucket di destinazione nella stessa Regioni AWS o in una diversa per ridurre la latenza, garantire conformità e sicurezza e per altri casi d'uso.
- **Operazioni in batch S3:** consente di gestire qualsiasi numero di oggetti su larga scala con una singola richiesta API S3 o pochi clic nella console Amazon S3. Puoi utilizzare le operazioni in batch per eseguire operazioni quali Copy (Copia), Invoke AWS Lambda function (Richiama funzione AWS Lambda) e Restore (Ripristina) su milioni o miliardi di oggetti.

### 3.1.3 Gestione degli accessi

Amazon S3 offre caratteristiche per la verifica e la gestione degli accessi ai tuoi bucket e oggetti. Per impostazione predefinita, i bucket S3 e gli oggetti al loro interno sono privati. Puoi accedere solo alle risorse S3 che hai creato. Per concedere autorizzazioni granulari delle risorse che supportano il tuo caso d'uso specifico o per verificare le autorizzazioni delle tue risorse Amazon S3, puoi utilizzare le seguenti caratteristiche.

- **Blocco dell'accesso pubblico di S3:** blocca l'accesso pubblico a bucket S3 e oggetti. Per impostazione predefinita, le impostazioni di blocco dell'accesso pubblico sono attive a livello di account e bucket.
- **AWS Identity and Access Management (IAM):** crea utenti IAM per il tuo Account AWS per gestire l'accesso alle risorse di Amazon S3. Ad esempio, puoi utilizzare IAM con Amazon S3 per controllare il tipo di accesso che un utente o un gruppo di utenti ha a un bucket S3 di proprietà del tuo Account AWS.
- **Policy di bucket:** utilizza il linguaggio delle policy basato su IAM per configurare le autorizzazioni basate su risorse per i bucket S3 e gli oggetti che contengono.
- **Punto di accesso Amazon S3:** configura gli endpoint di rete denominati con policy di accesso dedicate per gestire l'accesso ai dati su vasta scala per set di dati condivisi in Amazon S3.
- **Liste di controllo degli accessi (ACL):** concedi autorizzazioni di lettura e scrittura per singoli bucket e oggetti agli utenti autorizzati. Come regola generale, è consigliabile utilizzare policy basate sulle risorse S3 (policy di bucket e policy dei punti di accesso) o policy IAM per il controllo degli accessi anziché ACL. Le ACL sono un meccanismo di controllo degli accessi che precede le policy basate sulle risorse e le policy IAM.
- **S3 Object Ownership –** Disabilita le ACL e assumi la proprietà di ogni oggetto nel tuo bucket, semplificando la gestione degli accessi per i dati archiviati in Amazon S3. In qualità di proprietario del bucket, possiedi automaticamente e hai il pieno controllo su ogni oggetto nel tuo bucket e il controllo degli accessi per i tuoi dati è basato su policy.
- **Access Analyzer per S3:** valuta e monitora le policy di accesso al bucket S3, assicurando che forniscano solo l'accesso previsto alle risorse S3.

### 3.1.4 Elaborazione dei dati

Per trasformare i dati e attivare i flussi di lavoro in modo che automatizzino una serie di altre attività di elaborazione su larga scala, puoi utilizzare le seguenti caratteristiche.

- **S3 Object Lambda:** aggiungi il tuo codice alle richieste GET di S3 per modificare ed elaborare i dati quando vengono restituiti a un'applicazione. Questa caratteristica consente di filtrare righe, ridimensionare dinamicamente immagini, oscurare dati riservati e molto altro ancora.
- **Notifiche degli eventi:** consente di attivare flussi di lavoro che utilizzano Amazon Simple Notification Service (Amazon SNS), Amazon Simple Queue Service (Amazon SQS) e AWS Lambda quando viene apportata una modifica alle risorse S3.

### 3.1.5 Registrazione e monitoraggio dell'archiviazione

Amazon S3 fornisce strumenti di registrazione e monitoraggio che puoi utilizzare per monitorare e controllare come vengono utilizzate le tue risorse Amazon S3.

#### Strumenti di monitoraggio automatici:

- **Parametri Amazon CloudWatch di Amazon S3:** consentono di monitorare l'integrità operativa delle risorse S3 e configurare gli avvisi di fatturazione quando gli addebiti stimati raggiungono una soglia definita dall'utente.
- **AWS CloudTrail:** registra le operazioni eseguite da un utente, un ruolo o un Servizio AWS in Amazon S3. I registri CloudTrail forniscono il tracking dettagliato delle API per le operazioni a livello di bucket S3 e di oggetto.

#### Strumenti di monitoraggio manuali

- **Registrazione degli accessi al server:** fornisce registri dettagliati per le richieste effettuate a un bucket. Puoi utilizzare i registri di accesso al server per molti casi d'uso, come eseguire verifiche di sicurezza e accesso, conoscere la tua base clienti o capire meglio la fattura Amazon S3.
- **AWSTrusted Advisor:** consente di valutare il tuo account utilizzando controlli delle best practice AWS per identificare modi per ottimizzare l'infrastruttura AWS, migliorare la sicurezza e le prestazioni, ridurre i costi e monitorare le quote di servizio. Puoi quindi seguire i suggerimenti per ottimizzare i servizi e le risorse.

### 3.1.6 Analisi dei dati e informazioni dettagliate

Amazon S3 offre caratteristiche per aiutarti a ottenere visibilità sull'utilizzo dello spazio di archiviazione, che ti consente di comprendere, analizzare e ottimizzare meglio lo spazio di archiviazione su larga scala.

- **Amazon S3 Storage Lens:** consente di comprendere, analizzare e ottimizzare l'archiviazione. S3 Storage Lens fornisce oltre 29 parametri di utilizzo e attività e pannelli di controllo interattivi per aggregare i dati per l'intera organizzazione, account specifici, Regioni AWS, bucket o prefissi.
- **Analisi della classe di storage:** consente di analizzare i modelli di accesso all'archiviazione per decidere quando è il momento di spostare i dati in una classe più conveniente.
- **S3 Inventory con report di Inventory:** consente di verificare e creare report sugli oggetti e sui relativi metadati e configurare altre caratteristiche di Amazon S3 per intervenire sui report di Inventory. Ad esempio, puoi creare report sullo stato di replica e

crittografia degli oggetti. Per un elenco di tutti i metadata disponibili per ogni oggetto nei report di Amazon S3 Inventory, consulta questa sezione.

### 3.1.7 Forte coerenza

Amazon S3 fornisce una coerenza di lettura dopo scrittura per le operazioni PUT e DELETE per gli oggetti nel tuo bucket Amazon S3 in tutte le Regioni AWS. Questo comportamento vale sia per le scritture dei nuovi oggetti che per le richieste PUT che sovrascrivono gli oggetti esistenti e le richieste DELETE. Inoltre, le operazioni di lettura su Amazon S3 Select, le liste di controllo accessi (ACL) Amazon S3, i tag oggetto Amazon S3 e i metadati degli oggetti (ad esempio, l'oggetto HEAD) sono fortemente coerenti.

## 3.2 Come funziona Amazon S3

Amazon S3 è un servizio che consente di archiviare dati come **oggetti** nei bucket. Un oggetto è un file con tutti i metadati che lo descrivono. Un **bucket** è un container per oggetti o file.

Per archiviare dati in Amazon S3, per prima cosa devi creare un bucket e specificarne nome e Regione AWS. quindi devi caricare i dati nel bucket come oggetti in Amazon S3. Ogni oggetto contiene una chiave (o nome chiave), che è l'identificatore univoco dell'oggetto nel bucket.

S3 fornisce funzionalità che puoi configurare per supportare il tuo caso d'uso specifico. Puoi utilizzare Controllo versioni S3 per mantenere più versioni di un oggetto in un unico bucket e consentire di ripristinare oggetti che vengono accidentalmente eliminati o sovrascritti.

I bucket e gli oggetti che contengono sono privati e accessibili solo se concedi esplicitamente le autorizzazioni di accesso. Puoi utilizzare policy di bucket, policy AWS Identity and Access Management (IAM), liste di controllo accessi (ACL) e punti di accesso S3 per gestire gli accessi.

### 3.2.1 Bucket

Un bucket è un container per gli oggetti archiviati in Amazon S3. Puoi archiviare un numero qualsiasi di oggetti in un bucket e avere fino a 100 bucket nel tuo account (è possibile richiedere un aumento).

Ogni oggetto è contenuto in un bucket. Ad esempio, se l'oggetto denominato `photos/puppy.jpg` è archiviato nel bucket `DOC-EXAMPLE-BUCKET` della regione Stati Uniti occidentali (Oregon), è indirizzabile tramite l'URL:

`https://DOC-EXAMPLE-BUCKET.s3.us-west-2.amazonaws.com/photos/puppy.jpg`

### 3.2.2 Oggetti

Gli oggetti sono le entità fondamentali archiviate in Amazon S3 e sono composti da dati e **metadata**. I metadata sono invece un set di coppie nome-valore che descrivono l'oggetto. Queste coppie includono alcuni metadati di default, ad esempio la data dell'ultima modifica, e metadati HTTP standard, come Content-Type. È anche possibile specificare metadata personalizzati al momento dell'archiviazione dell'oggetto.

Un oggetto viene identificato in modo univoco in un bucket tramite un (nome) chiave e un ID versione (se Controllo versioni S3 è abilitato nel bucket).

### 3.2.3 Chiavi

Una chiave oggetto (o nome chiave) è l'identificatore univoco di un oggetto in un bucket. Per ogni oggetto in un bucket è presente esattamente una chiave. La combinazione di bucket, chiave oggetto e, facoltativamente, ID versione (se il Controllo versioni S3 è abilitato per il bucket) identifierà in modo univoco ogni oggetto. Quindi puoi pensare ad Amazon S3 come a una mappa di dati di base tra "bucket + chiave + versione" e l'oggetto stesso.

Si può fare riferimento in modo univoco a ogni oggetto in Amazon S3 tramite la combinazione di endpoint del servizio Web, nome del bucket, chiave e, facoltativamente, una versione. Ad esempio, nell'URL

<https://DOC-EXAMPLE-BUCKET.s3.us-west-2.amazonaws.com/photos/puppy.jpg>,  
DOC-EXAMPLE-BUCKET è il nome del bucket e /photos/puppy.jpg è la chiave.

### 3.3 Creazione di un Bucket, [1]

- Premere su **Crea bucket**.

The screenshot shows the AWS S3 Buckets page. At the top, there is a header with the text "Bucket (9) Info" and a link "Ulteriori informazioni". Below the header are four buttons: "C" (refresh), "Copia ARN" (Copy ARN), "Vuoto" (Empty), and "Elimina" (Delete). A prominent orange button labeled "Crea bucket" is highlighted with a red arrow pointing to it. Below these buttons is a search bar with the placeholder text "Cerca bucket in base al nome". Underneath the search bar is a table with columns: Nome, Regione AWS, Accesso, and Data di creazione. The table contains one row for a bucket named "unict-lm18-bucket" located in "Stati Uniti orientali (Virginia settentrionale) us-east-1" with "Bucket e oggetti non pubblici" access and created on "18 Mar 2022 05:54:08 PM CET".

- Si aprirà una schermata dove vengono chieste alcune informazioni sul bucket da creare come:

- ★ **Bucket name (Nome bucket)**, immettere un nome conforme a DNS per il bucket.  
Il nome del bucket deve:
  - Essere univoco in tutto Amazon S3.
  - Deve contenere da 3 a 63 caratteri.
  - Non contiene caratteri maiuscoli.
  - Iniziare con una lettera minuscola o un numero.
- Una volta creato il bucket, non è possibile modificarne il nome.
- ★ In **Regione** scegli la regione Regione AWS in cui si desidera che il bucket risieda. Scegliere una regione nelle vicinanze per ridurre al minimo la latenza e i costi o rispondere a requisiti normativi. Gli oggetti archiviati in una regione non la lasciano mai a meno che non vengano trasferiti esplicitamente in un'altra regione.
- ★ Copiare le impostazioni da un'altro bucket esistente.

# Crea bucket Info

I bucket sono contenitori per i dati archiviati in S3. [Ulteriori informazioni](#)

## Configurazione generale

Nome bucket

myawsbucket

1

Il nome del bucket deve essere univoco e non deve contenere spazi o lettere maiuscole. [Consulta le regole per la denominazione dei bucket](#)

Regione AWS

Stati Uniti orientali (Virginia settentrionale) us-east-1

2

Copia le impostazioni dal bucket esistente: *facoltativo*

Vengono copiate solo le impostazioni del bucket nella configurazione seguente.

Scegli bucket

3

- ★ Proprietà dell'oggetto: è possibile abilitare o disabilitare le liste di controllo di accessi (ACL).

## Proprietà dell'oggetto Info

Controlla la proprietà degli oggetti scritti in questo bucket da altri account AWS e l'uso delle liste di controllo accessi (ACL). La proprietà degli oggetti determina chi può specificare l'accesso agli oggetti.

### ACL disabilitate (consigliato)

Tutti gli oggetti in questo bucket sono di proprietà di questo account. L'accesso a questo bucket e ai relativi oggetti viene specificato utilizzando solo le policy.

### ACL abilitate

Gli oggetti in questo bucket possono essere di proprietà di altri account AWS. L'accesso a questo bucket e ai relativi oggetti può essere specificato utilizzando le ACL.

Proprietà dell'oggetto

Bucket owner enforced

- ★ Impostazioni di blocco dell'accesso pubblico al bucket.

## Impostazioni di blocco dell'accesso pubblico per questo bucket

L'accesso pubblico viene concesso a bucket e oggetti attraverso le liste di controllo accessi (ACL), le policy del bucket, le policy del punto di accesso o tutte le opzioni precedenti. Per assicurarti di bloccare l'accesso pubblico a questo bucket e ai relativi oggetti, attiva il blocco di tutti gli accessi pubblici. Queste impostazioni si applicano solo a questo bucket e ai rispettivi punti di accesso. AWS consiglia di attivare il blocco di tutti gli accessi pubblici. Prima di confermare una qualsiasi di queste impostazioni, assicurati che le applicazioni funzionino correttamente senza l'accesso pubblico. Se hai bisogno di qualche livello di accesso pubblico per questo bucket e per gli oggetti presenti al suo interno, puoi personalizzare le impostazioni individuali sottostanti per adattarle ai casi d'uso di storage specifici. [Ulteriori informazioni](#)



### Blocca tutti gli accessi pubblici

L'attivazione di questa impostazione equivale all'attivazione di tutte e quattro le impostazioni seguenti. Ognuna delle impostazioni seguenti è indipendente l'una dall'altra.

#### Blocca gli accessi pubblici a bucket e oggetti concessi tramite le **nuove liste di controllo accessi (ACL)**

S3 bloccherà le autorizzazioni di accesso pubblico applicate ai nuovi bucket e oggetti e impedirà la creazione di nuove liste di controllo accessi per l'accesso pubblico a oggetti e bucket esistenti. Questa impostazione non modifica nessuna autorizzazione esistente che permette l'accesso pubblico alle risorse S3 utilizzando le ACL.

#### Blocca gli accessi pubblici a bucket e oggetti concessi tramite **qualsiasi lista di controllo accessi (ACL)**

S3 ignorerà tutte le ACL che concedono accesso pubblico a bucket e oggetti.

#### Blocca gli accessi pubblici a bucket e oggetti concessi tramite le **nuove policy pubbliche del bucket o del punto d'accesso**

S3 bloccherà le nuove policy del bucket e del punto di accesso che concedono l'accesso pubblico a bucket e oggetti. Questa impostazione non modifica nessuna policy esistente che permette l'accesso pubblico alle risorse S3.

#### Blocca gli accessi pubblici a bucket e tra account a bucket e oggetti concessi tramite **qualsiasi policy pubblica del bucket o del punto di accesso**

S3 ignorerà l'accesso pubblico su più account per bucket e punti di accesso con policy che autorizzano l'accesso pubblico a bucket e oggetti.

## ★ Abilitare il versioning.

### Versioni multiple per il bucket

La funzione versioni multiple consente di conservare più varianti di un oggetto nello stesso bucket. È possibile utilizzare la funzione versioni multiple per conservare, recuperare e ripristinare ogni versione di ogni oggetto archiviato nel bucket Amazon S3. Grazie a questa funzione, è possibile eseguire facilmente il ripristino da operazioni involontarie degli utenti ed errori delle applicazioni. [Ulteriori informazioni](#)

#### Versioni multiple per il bucket

- Disabilita  
 Abilita

## ★ Attribuire un Tag.

### Tag (0) - **facoltativo**

Monitora i costi di storage o altri criteri mediante l'applicazione di tag al bucket. [Ulteriori informazioni](#)

Nessun tag associato a questo bucket.

[Aggiungi tag](#)

## ★ Abilitare o meno la crittografia dei dati.

## Crittografia predefinita

Crittografa automatica i nuovi oggetti archiviati in questo bucket. [Ulteriori informazioni](#)

Crittografia lato server

- Disabilita
- Abilita

★ Blocco oggetti. Infine selezionare **Create bucket (Crea bucket)**.

## ▼ Impostazioni avanzate

Blocco oggetti

Archivia gli oggetti utilizzando un modello WORM (write-once-read-many) per evitare che gli oggetti vengano eliminati o sovrascritti per un arco di tempo fisso o a tempo indeterminato. [Ulteriori informazioni](#)

- Disabilita
- Abilita

Permette di bloccare definitivamente gli oggetti presenti in questo bucket. Al termine della creazione del bucket, è necessaria una configurazione di blocco degli oggetti aggiuntiva per impedire l'eliminazione e la sovrascrittura degli oggetti in esso contenuti.

ⓘ Il blocco degli oggetti funziona solo nei bucket con versione. L'abilitazione del blocco degli oggetti abilita automaticamente la funzione Versioni multiple per il bucket.

ⓘ Dopo aver creato il bucket, è possibile caricarvi file e cartelle e configurare ulteriori impostazioni del bucket.

Annulla

**Crea bucket**

## 3.4 Caricare un oggetto in un Bucket, [1]

Dopo aver creato un bucket in Amazon S3, è possibile caricare un oggetto nel bucket. Un oggetto può essere qualsiasi tipo di file: file di testo, immagine, video e così via.

Per caricare un oggetto in un bucket:

- Nell'elenco **Bucket** selezionare il nome del bucket in cui si desidera caricare l'oggetto.

### Bucket (9) [Info](#)

I bucket sono contenitori per i dati archiviati in S3. [Ulteriori informazioni](#)

[!\[\]\(4ad5a38ba6b9203ba3fdd31e30ea8da0\_img.jpg\) C](#) [!\[\]\(97370e524e176c70a5b5eb82b31cce64\_img.jpg\) Copia ARN](#) [Vuoto](#) [Elimina](#) [Crea bucket](#)

Cerca bucket in base al nome < 1 > ⌂

Nome	Regione AWS	Accesso	Data di creazione
<input checked="" type="radio"/> unict-lm18-bucket	Stati Uniti orientali (Virginia settentrionale) us-east-1	Bucket e oggetti non pubblici	18 Mar 2022 05:54:08 PM CET

- Nella scheda **Oggetti** del bucket selezionare **Carica**.

# unict-lm18-bucket [Info](#)

Oggetti

Proprietà

Autorizzazioni

Parametri

Gestione

Punti di accesso

## Oggetti (1)

Gli oggetti sono le entità fondamentali archiviate in Amazon S3. Per ottenere un elenco di tutti gli oggetti nel bucket, puoi utilizzare l'[inventario di Amazon S3](#). Per consentire ad altri utenti di accedere ai tuoi oggetti, è necessario concedere loro le autorizzazioni esplicitamente. [Ulteriori informazioni](#)



Copia URI S3

Copia URL

Scarica

Apri

Elimina

Operazioni ▾

Crea cartella

Carica

Trova oggetti per prefisso

< 1 > ⚙

<input type="checkbox"/>	Nome	Tipo	Ultima modifica	Dimensioni	Classe di storage
	poste.png	png	18 Mar 2022 05:56:15 PM CET	38.6 KB	Standard

### 3. In File e cartelle, selezionare Aggiungi file.

## Carica [Info](#)

Aggiungi i file e le cartelle che desideri caricare in S3. Per caricare un file di dimensioni superiori a 160 GB, utilizza AWS CLI, l'SDK AWS o l'API REST di Amazon S3. [Ulteriori informazioni](#)

Trascina i file e le cartelle che desideri caricare qui oppure scegli Aggiungi file o Aggiungi cartella.

## File e cartelle (0)

Rimuovi

Aggiungi file

Aggiungi cartella

Tutti i file e le cartelle in questa tabella verranno caricati.

Trova per nome

< 1 >

<input type="checkbox"/>	Nome	Cartella	Tipo	Dimensioni
Nessun file o cartelle				

Nessun file o cartelle  
Non hai scelto alcun file o cartella da caricare.

### 4. Selezionare un file da caricare, quindi scegliere Apri.

### 5. Scegliere Carica.

### 3.5 Download di un oggetto, [1]

Dopo aver caricato un oggetto in un bucket, è possibile visualizzare le informazioni sull'oggetto e scaricare l'oggetto nel computer locale.

- Nell'elenco **Buckets (Bucket)** scegliere il nome del bucket dal quale si desidera scaricare un oggetto.

**Bucket (9) [Info](#)**

I bucket sono contenitori per i dati archiviati in S3. [Ulteriori informazioni](#)

[Copia ARN](#) [Vuoto](#) [Elimina](#) [Crea bucket](#)

Cerca bucket in base al nome

Nome	Regione AWS	Accesso	Data di creazione
<a href="#">unict-lm18-bucket</a>	Stati Uniti orientali (Virginia settentrionale) us-east-1	Bucket e oggetti non pubblici	18 Mar 2022 05:54:08 PM CET

- È possibile scaricare un oggetto da un bucket S3 in uno qualsiasi dei modi seguenti:

  - Dal pannello **Oggetti** selezionare l'oggetto da scaricare e premere **Scarica**.

[Copia URI S3](#)  [Copia URL](#)  [Scarica](#) [Apri](#) [Elimina](#)

[Operazioni](#) [Crea cartella](#) [Carica](#) 2

Trova oggetti per prefisso

<input checked="" type="checkbox"/>	Nome	Tipo	Ultima modifica	Dimensioni	Classe di storage
<span style="border: 1px solid red; border-radius: 50%; padding: 5px 10px; display: inline-block;">1</span>	<a href="#">poste.png</a>	png	18 Mar 2022 05:56:15 PM CET	38.6 KB	Standard

- Selezionare il nome dell'oggetto che si desidera scaricare.

Nella pagina **Panoramica** seleziona l'oggetto e, dal menù **Operazioni**, scegli **Scarica come** se si desidera scaricare l'oggetto in una cartella specifica.

[1](#) [Scarica come](#) 2

- Condividere con un URL prefirmato
- Calcola la dimensione totale
- Copia
- Sposta
- Avvia il ripristino
- Query con S3 Select

**Operazioni di modifica**

- Rinomina l'oggetto
- Modifica la classe di storage
- Modifica la crittografia lato server
- Modifica metadati
- Modifica tag

[Operazioni](#) 1

- Se si desidera scaricare una versione specifica dell'oggetto, scegliere il nome dell'oggetto che si desidera scaricare. Scegliere la scheda **Versioni**, quindi dal menù **Operazioni** scegli **Scarica** o **Scarica come** se si desidera scaricare l'oggetto in una cartella specifica.

#### 4. IAM (AWS Identity and Access Management), [1]

AWS Identity and Access Management (IAM) è un servizio Web che consente di controllare in modo sicuro l'accesso alle risorse AWS. Utilizza IAM per controllare chi è autenticato (accesso effettuato) e autorizzato (dispone di autorizzazioni) per l'utilizzo di risorse.

Quando crei un Account AWS per la prima volta, inizi con una singola identità di accesso che ha accesso completo a tutti i servizi e le risorse AWS nell'account. Tale identità è detta utente root Account AWS e puoi accedervi con l'indirizzo e-mail e la password utilizzati per creare l'account. È vivamente consigliato di non utilizzare l'utente root per le attività quotidiane, anche quelle amministrative. Rispetta piuttosto la best practice di utilizzare l'utente root soltanto per creare il tuo primo utente IAM. Quindi conserva al sicuro le credenziali dell'utente root e utilizzale per eseguire solo alcune attività di gestione dell'account e del servizio.

The screenshot shows the AWS Management Console interface. At the top, there's a navigation bar with a 'Servizi' button (circled in red, labeled 1), a search bar ('Ricerca di servizi, caratteristiche, blog, documenti [Opzione+S]'), and other icons for notifications and help. On the left, a sidebar lists various service categories: Gestione e governance, Integrazione di applicazioni, Internet of Things, Machine Learning, Migrazione e trasferimento, Quantum Technologies, Realtà aumentata e realtà virtuale, Reti e distribuzione di contenuti, Robotica, Satellite, Servizi multimediali, Sicurezza, Identità, conformità (circled in red, labeled 2), Storage, Strumenti per sviluppatori, and Sviluppo di videogiochi. A large red circle with the number 3 highlights the 'IAM' service card in the main content area. The 'IAM' card has a star icon and the text 'Gestisci gli accessi alle risorse AWS'. Other visible services include Hosting e gestione di Active Directory, AWS Firewall Manager, GuardDuty, Inspector, Key Management Service, Amazon Macie, and AWS Network Firewall.

IAM offre le seguenti funzionalità:

1. **Accesso condiviso al tuo account AWS.** Puoi concedere ad altri utenti le autorizzazioni per amministrare e utilizzare le risorse nel tuo account AWS senza la necessità di condividere la password o la chiave di accesso.
2. **Autorizzazioni granulari.** Puoi concedere autorizzazioni diverse a diverse persone per diverse risorse. Ad esempio, potresti consentire ad alcuni utenti un accesso completo ad Amazon Elastic Compute Cloud (Amazon EC2), Amazon Simple Storage Service

(Amazon S3), Amazon DynamoDB, Amazon Redshift e altri servizi AWS. Per gli altri utenti, puoi consentire l'accesso in sola lettura ad alcuni bucket S3, oppure l'autorizzazione ad amministrare solo alcune istanze EC2 oppure l'autorizzazione ad accedere alle informazioni di fatturazione, ma nient'altro.

3. **Accesso sicuro alle risorse AWS per applicazioni che funzionano su Amazon EC2.**  
Puoi utilizzare le funzionalità di IAM per fornire in maniera sicura le credenziali per le applicazioni che funzionano su istanze EC2. Queste credenziali forniscono le autorizzazioni alla tua applicazione AWS per accedere ad altre risorse. Alcuni esempi includono i bucket S3 e le tabelle DynamoDB.
4. **Autenticazione a più fattori (MFA).** Puoi aggiungere l'autenticazione a due fattori per il tuo account e per i singoli utenti per maggiore sicurezza. Con MFA tu o i tuoi utenti dovranno fornire non solo una password o la chiave di accesso che funziona con il tuo account, ma anche un codice da un dispositivo appositamente configurato.
5. **Federazione delle identità.** Puoi consentire agli utenti che utilizzano già le password altrove, ad esempio nella tua rete aziendale o con un provider di identità Internet, di ottenere l'accesso temporaneo al tuo account AWS.
6. **Informazioni d'identità per la sicurezza.** Se utilizzi AWS CloudTrail riceverai i record del log che includono le informazioni su chi effettua le richieste per le risorse nel tuo account. Queste informazioni sono basate sulle identità IAM.
7. **Integrato con molti servizi AWS.** Per un elenco di servizi AWS che funzionano con IAM, consulta [AWS Servizi che funzionano con IAM](#).
8. **Consistente finale.** IAM, come molti altri servizi AWS, è a consistenza finale. IAM raggiunge un'alta disponibilità replicando i dati su più server nei data center di Amazon di tutto il mondo. Se una richiesta per modificare alcuni dati ha successo, la modifica viene completata e memorizzata in maniera sicura. Tuttavia, le modifiche devono essere replicate su IAM e questo può richiedere tempo. Tali modifiche includono la creazione o l'aggiornamento di utenti, gruppi, ruoli, o policy. Si consiglia di non includere tali modifiche IAM nei percorsi critici e ad alta disponibilità del codice dell'applicazione. Al contrario, apporta modifiche IAM in un'inizializzazione separata o in una routine di configurazione che si esegue meno frequentemente. Inoltre, assicurarsi di verificare che le modifiche siano state propagate prima che i flussi di lavoro di produzione dipendano da esse.
9. **Utilizzo gratis.** AWS Identity and Access Management (IAM) e AWS Security Token Service (AWS STS) sono funzionalità dell'account AWS offerte senza costi aggiuntivi.

#### 4.1 Creazione di un gruppo di utenti, [1]

1. Accedi alla console IAM come proprietario dell'account scegliendo **Utente root**.  
**Nota:** È fortemente consigliato rispettare la best practice sull'utilizzo dell'utente IAM **Administrator** di seguito e conservare in un luogo sicuro le credenziali dell'utente root. Accedere come utente root solo per eseguire alcune attività di gestione dell'account e del servizio.
2. Nel pannello di controllo laterale andare alla voce **Gestione degli accessi** selezionare **Gruppi di utenti** e successivamente **Crea gruppo**.

**Ricerca IAM**

Pannello di controllo

Gestione degli accessi

**Gruppi di utenti** (4) [Informazioni](#)

Un gruppo di utenti è una raccolta di utenti IAM. Utilizzare i gruppi per specificare le autorizzazioni per una raccolta di utenti.

Filtra i gruppi di utenti in base alla proprietà o al nome del gruppo e premi Invio

	Nome del gruppo	Utenti	Autorizzazioni
<input type="checkbox"/>	[REDACTED]	1	Definito
<input type="checkbox"/>	[REDACTED]	3	Definito
<input type="checkbox"/>	[REDACTED]	13	Definito
<input type="checkbox"/>	[REDACTED]	3	Definito

3. Nella pagina **Creare gruppo di utenti** inseriamo:

- Il nome del gruppo di utenti:

### Assegnare un nome al gruppo

#### Nome del gruppo di utenti

Inserisci un nome significativo per identificare questo gruppo.

CloudDevelopers2022

Massimo 128 caratteri. Utilizzare caratteri alfanumerici e i seguenti caratteri speciali: "+=,.@-\_"

- (facoltativo) Selezionare gli utenti da aggiungere al gruppo.

**Aggiungi utenti al gruppo - Facoltativo (20)** [Informazioni](#)

Un utente IAM è un'entità che crei in AWS per rappresentare la persona o l'applicazione che la utilizza per interagire con AWS. Un utente può appartenere a un massimo di 10 gruppi.

Cerca

	Nome utente	Gruppi	Ultima attività	Ora creazione
<input type="checkbox"/>	[REDACTED]	1	20 giorni fa	20 giorni fa
<input type="checkbox"/>	[REDACTED]	1	9 mesi fa	10 mesi fa
<input type="checkbox"/>	[REDACTED]	1	Nessuno	20 giorni fa

- (facoltativo) Collegare le policy (fino a 10) al gruppo.

## Collega policy di autorizzazione - Facoltativo

(769)



Crea policy

### Informazioni

Puoi collegare fino a 10 policy a questo gruppo di utenti. Tutti gli utenti in questo gruppo disporranno delle autorizzazioni definite nelle policy selezionate.

Filtra le policy in base alla proprietà o al nome della policy e premi Invio

29 corrispondenze

< 1 2 >



"EC2"

Annulla filtri

<input type="checkbox"/>	Nome della policy	Tipo
<input type="checkbox"/>	EC2Access4Developers	Gestite dal clie...
<input type="checkbox"/>	EC2perSviluppatori	Gestite dal clie...
<input type="checkbox"/>	EC2PolicyXSviluppatori	Gestite dal clie...
<input type="checkbox"/>	policy-EC2-2019	Gestite dal clie...
<input type="checkbox"/>	AmazonEC2FullAccess	Gestite da AWS

d. Premere infine su **Crea gruppo**.

[Piccola parentesi sulle Policy]

Ogni policy di autorizzazione è descritta mediante una sintassi json. Per esempio la policy relativa a **AmazonEC2FullAccess** è:

```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Action": "ec2:*",  
      "Effect": "Allow",  
      "Resource": "*"  
    },  
    {  
      "Effect": "Allow",  
      "Action": "elasticloadbalancing:*",  
      "Resource": "*"  
    },  
    {  
      "Effect": "Allow",  
      "Action": "cloudwatch:*",  
      "Resource": "*"  
    },  
    {  
      "Effect": "Allow",  
      "Action": "autoscaling:*",  
      "Resource": "*"  
    }]
```

```
        },
        {
            "Effect": "Allow",
            "Action": "iam:CreateServiceLinkedRole",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "iam:AWSServiceName": [
                        "autoscaling.amazonaws.com",
                        "ec2scheduled.amazonaws.com",
                        "elasticloadbalancing.amazonaws.com",
                        "spot.amazonaws.com",
                        "spotfleet.amazonaws.com",
                        "transitgateway.amazonaws.com"
                    ]
                }
            }
        }
    ]
}
```

## 4.2 Creazione di un utente, [1]

1. Nel pannello di controllo laterale andare alla voce **Gestione degli accessi** selezionare **Utenti** e successivamente **Aggiungi utenti**.

The screenshot shows the AWS IAM service interface. On the left sidebar, under 'Gestione degli accessi', the 'Utenti' option is highlighted with a red box and a red circle containing the number '1'. At the top right, there are two buttons: a grey 'Elimina' button and a blue 'Aggiungi utenti' button with a red border and a red circle containing the number '2'. The main content area has a heading 'Utenti (20) Informazioni' and a sub-instruction: 'Un utente IAM è un'identità con credenziali a lungo termine che viene utilizzata per interagire con AWS in un account.' Below this is a search bar with placeholder text 'Trova gli utenti tramite nome utente o chiave di accesso'. The main table lists 20 users, with the first three rows visible. Each row contains a checkbox, the user name (redacted), their assigned groups (redacted), and their last activity time (e.g., '21 minuti fa', '4 giorni fa').

2. Digita un nome utente per il nuovo utente. Questo è il nome di accesso per AWS. Se desideri aggiungere più utenti, scegli **Aggiungi un altro utente** per ogni utente aggiuntivo e digita il rispettivo nome utente. È possibile aggiungere fino a 10 utenti simultaneamente

## Imposta dettagli dell'utente

Puoi aggiungere più utenti contemporaneamente con lo stesso tipo di accesso e autorizzazioni. [Ulteriori informazioni](#)

<b>Nome utente*</b>	<input type="text" value="prova"/>
<a href="#" style="color: blue; font-weight: bold;">+ Aggiungi un altro utente</a>	

3. Selezionare il tipo di accesso che questo insieme di utenti avrà. Puoi selezionare l'accesso programmatico, l'accesso alla AWS Management Console o entrambi.
  - a. Seleziona **Accesso programmatico** se gli utenti richiedono l'accesso all'API, alla AWS CLI o a Tools for Windows PowerShell. In questo modo viene creata una chiave di accesso per ogni nuovo utente. Puoi visualizzare o scaricare le chiavi di accesso quando arrivi alla pagina **Final (Finale)**.
  - b. Seleziona **AWS Management Console access (Accesso alla console)** se gli utenti richiedono l'accesso alla AWS Management Console. In questo modo viene creata una password per ogni nuovo utente.
    - i. Per **Console password (Password console)**, scegliere una delle opzioni seguenti:
      1. **Autogenerated password (Password autogenerata)**. Ogni utente ottiene una password casuale che soddisfa la policy sulle password dell'account. È possibile visualizzare o scaricare le password quando si arriva alla pagina **Final (Finale)**.
      2. **Custom password (Password personalizzata)**. A ogni utente viene assegnata la password digitata nella casella.
    - ii. (Facoltativo) Consigliamo di selezionare **Require password reset (Richiedi reset della password)** per assicurarsi che gli utenti siano costretti a modificare la propria password al primo accesso.

**Nota:** Se un amministratore ha attivato l'impostazione di policy sulle password dell'account Consenti agli utenti di modificare la propria password, questa casella di controllo non esegue alcuna operazione. In caso contrario, viene allegata automaticamente una policy AWS gestita denominata IAMUserChangePassword ai nuovi utenti. La policy concede agli utenti l'autorizzazione a modificare le proprie password.

#### Seleziona Next: Permissions (Successivo: Autorizzazioni).

##### Seleziona il tipo di accesso AWS

Seleziona il modo in cui questi utenti accederanno principalmente ad AWS. Se scegli solo l'accesso programmatico, ciò NON impedirà agli utenti di accedere alla console utilizzando un ruolo assunto. Le chiavi di accesso e le password generate automaticamente vengono fornite nell'ultima fase.

[Ulteriori informazioni](#)

**Selezione il tipo di credenziali**

**AWS\***

**1**  Chiave di accesso - Accesso programmatico  
Abilita una ID chiave di accesso e una chiave di accesso segreta per le API di AWS, l'interfaccia a riga di comando, SDK e altri strumenti di sviluppo.

**2**  Password - Accesso alla Console di gestione AWS  
Abilita una password che consente agli utenti di effettuare l'accesso alla console di gestione AWS.

**Password console\***

**3**  Password autogenerata  
 Password personalizzata

**Richiesta reimpostazione della password**

**4**  L'utente deve creare una nuova password al prossimo accesso  
Gli utenti ottengono automaticamente la policy [IAMUserChangePassword](#) per consentire loro di modificare la propria password.

\* Campo obbligatorio

Annulla

**Successivo: Autorizzazioni**

4. Nella pagina **Set permissions (Imposta autorizzazioni)**, specifica il modo in cui

desideri assegnare le autorizzazioni a questo insieme di nuovi utenti. Seleziona una delle seguenti tre opzioni:

- Add user to group (Aggiungi utente al gruppo)**. Scegli questa opzione se desideri assegnare gli utenti a uno o più gruppi che hanno già le policy di autorizzazione. IAM mostra un elenco dei gruppi nell'account, insieme alle loro policy collegate. Puoi selezionare uno o più gruppi esistenti oppure selezionare **Create group (Crea gruppo)** per creare un nuovo gruppo.

▼ Imposta autorizzazioni



Aggiungi un utente a un gruppo esistente o creane uno nuovo. L'utilizzo dei gruppi costituisce una procedura consigliata per gestire le autorizzazioni dell'utente in base alle funzioni lavorative. [Ulteriori informazioni](#)

### Add user to group (Aggiungi utente al gruppo)

This screenshot shows the "Add user to group" interface:

- A search bar at the top.
- A dropdown menu labeled "Gruppo" with "ML-Users" selected.
- A list of users with checkboxes:
  - ML-Users (selected)
  - Developers2022

- Copy permissions from existing user (Copia le autorizzazioni dall'utente esistente)**. Scegli questa opzione per copiare tutte le appartenenze ai gruppi, le policy gestite collegate, le policy in linea integrate e qualsiasi limite delle autorizzazioni esistente da un utente esistente ai nuovi utenti. IAM mostra l'elenco degli utenti nel tuo account. Seleziona quello le cui autorizzazioni corrispondono il più possibile alle esigenze dei nuovi utenti.

▼ Imposta autorizzazioni



Seleziona un utente da cui copiare policy e appartenenza ai gruppi.

### Copia le autorizzazioni dall'utente esistente

This screenshot shows the "Copia le autorizzazioni dall'utente esistente" interface:

- A search bar at the top.
- A dropdown menu labeled "Nome utente" with "ML-Users" selected.
- A list of users with checkboxes:
  - ML-Users (selected)
  - [Redacted]

- Attach existing policies directly (Collega direttamente le policy esistenti)**. Scegli questa opzione per visualizzare l'elenco delle policy gestite da AWS e dal cliente nell'account. Seleziona le policy che desideri collegare ai nuovi utenti oppure scegli **Create policy (Crea policy)** per aprire una nuova scheda del browser e creare una nuova policy da zero. Una volta creata la policy, chiudi la scheda e torna alla scheda originale per aggiungere la policy al nuovo utente.

## ▼ Imposta autorizzazioni

Add user to group (Aggiungi utente al gruppo)

Copia le autorizzazioni dall'utente esistente

Collega direttamente le policy esistenti

**Crea policy**

Filtra policy ▾ Cerca Visualizzazione di 769 risultati

Nome policy ▲	Tipo	Utilizzata come
<input type="checkbox"/> AdministratorAccess	Funzione lavorativa	Nessuna
<input type="checkbox"/> AdministratorAccess-Amplify	Gestita da AWS	Nessuna
<input type="checkbox"/> AdministratorAccess-AWSElasticBeanstalk	Gestita da AWS	Permissions policy (1)
<input type="checkbox"/> AlexaForBusinessDeviceSetup	Gestita da AWS	Nessuna
<input type="checkbox"/> AlexaForBusinessFullAccess	Gestita da AWS	Nessuna

Annula Precedente Successivo: Tag

Come best practice, si consiglia invece di collegare le policy per un gruppo e quindi rendere gli utenti membri dei gruppi appropriati.

Scegliere **Next: Tags (Successivo: Tag)**.

5. (Facoltativo) Aggiungere metadati all'utente collegando i tag come coppie chiave-valore (massimo 50):

### Aggiungi tag (opzionale)

I tag IAM sono coppie chiave-valore che puoi aggiungere a utente. I tag possono includere informazioni sull'utente come, ad esempio, un indirizzo e-mail, oppure possono essere descrittivi come, ad esempio, una posizione professionale. È possibile usare i tag per organizzare, monitorare o controllare l'accesso per questo utente. [Ulteriori informazioni](#)

Chiave	Valore (facoltativo)	Rimuovi
Aggiungi una nuova chiave		

Puoi aggiungere 50 altri tag.

Annula Precedente Successivo: Verifica

6. Scegli **Next: Review (Successivo: revisione)** per visualizzare tutte le scelte eseguite fino a questo momento. Quando sei pronto per continuare, seleziona **Create user (Crea utente)**.
7. Per visualizzare le chiavi di accesso degli utenti (ID chiave di accesso e chiavi di accesso segrete), scegli **Show (Mostra)** accanto a ciascuna password e chiave di accesso che desideri visualizzare. Per salvare le chiavi di accesso, scegliere **Download .csv (Scarica .csv)** e quindi salvare il file in una posizione sicura.  
**Importante:** Questa è l'unica occasione per visualizzare o scaricare le chiavi di accesso segrete e devi fornire queste informazioni ai tuoi utenti prima che possano utilizzare l'API AWS. Salva i nuovi ID chiave di accesso e Secret Access Key dell'utente in un luogo sicuro. **Successivamente a questa fase non sarà più possibile accedere alle chiavi segrete.**
8. Fornire a ciascun utente le proprie credenziali. Nella pagina finale puoi selezionare **Send email (Invia e-mail)** accanto a ciascun utente. Il client di posta elettronica locale si apre con una bozza che è possibile personalizzare e inviare. Il modello dell'e-mail include i

seguenti dettagli per ciascun utente:

- a. Nome utente;
- b. URL della pagina per l'accesso all'account. Utilizza il seguente esempio, sostituendo il numero ID dell'account corretto o l'alias dell'account:

<https://AWS-account-ID or alias.signin.aws.amazon.com/console>

The screenshot shows a success message from the AWS Management Console: "You successfully created the users shown below. You can view and download user security credentials. You can also email users instructions for signing in to the AWS Management Console. This is the last time these credentials will be available to download. However, you can create new credentials at any time." Below this, a link to "Users with AWS Management Console access can sign-in at: https://272755869213.signin.aws.amazon.com/console" is highlighted with a red box. A yellow arrow points from this link down to a "Download .csv" button. Another yellow arrow points from the "Console login link" column in a LibreOffice Calc spreadsheet to the same link in the AWS message.

User	Password
giu.pap	2xX9@z }T+Ko Hide

credentials.csv - LibreOffice Calc

User name	Password	Console login link	Access key ID	Secret access key
giu.pap	dhJXZoiewRqk	https://272755869213.signin.aws.amazon.com/console		

### 4.3 Chiavi di accesso, [1]

Le chiavi di accesso sono credenziali a lungo termine per un utente IAM o l'utente root dell'Account AWS. Puoi utilizzare le chiavi di accesso per firmare le richieste programmatiche inviate all'AWS CLI o all'API AWS (direttamente o tramite l'SDK AWS).

Le chiavi di accesso sono composte da due parti: un ID chiave di accesso (ad esempio AKIAIOSFODNN7EXAMPLE) e una chiave di accesso segreta (ad esempio, wJalrXUtnFEMI/K7MDENG/bPxRficiYEXAMPLEKEY). Come un nome utente e una password, è necessario utilizzare sia l'ID chiave di accesso sia la chiave di accesso segreta insieme per autenticare le richieste dell'utente. Gestisci le tue chiavi di accesso in modo sicuro mentre crei il nome utente e la password.

**Nota:** Come best practice, utilizza credenziali di sicurezza temporanee (ruoli IAM) invece delle chiavi di accesso e disabilita qualsiasi chiave di accesso dell'utente root dell'Account AWS.

Se devi comunque utilizzare chiavi di accesso a lungo termine, puoi creare, modificare, visualizzare o ruotare le chiavi di accesso (gli ID chiave di accesso e le chiavi di accesso segrete). È possibile avere massimo due chiavi di accesso. In questo modo è possibile ruotare le chiavi attive in base alle best practice.

Se crei una coppia di chiavi di accesso, salva l'ID chiave di accesso e la chiave di accesso segreta in una posizione sicura. La chiave di accesso segreta è disponibile solo al momento della creazione. Se perdi la chiave di accesso segreta, è necessario eliminarla e crearne una nuova.

#### 4.3.1 Autorizzazioni richieste

Per creare le chiavi di accesso per l'utente IAM, è necessario disporre delle autorizzazioni dalla policy seguente:

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Sid": "CreateOwnAccessKeys",  
            "Effect": "Allow",  
            "Action": [  
                "iam:CreateAccessKey",  
                "iam:GetUser",  
                "iam>ListAccessKeys"  
            ],  
            "Resource": "arn:aws:iam::*:user/${aws:username}"  
        }  
    ]  
}
```

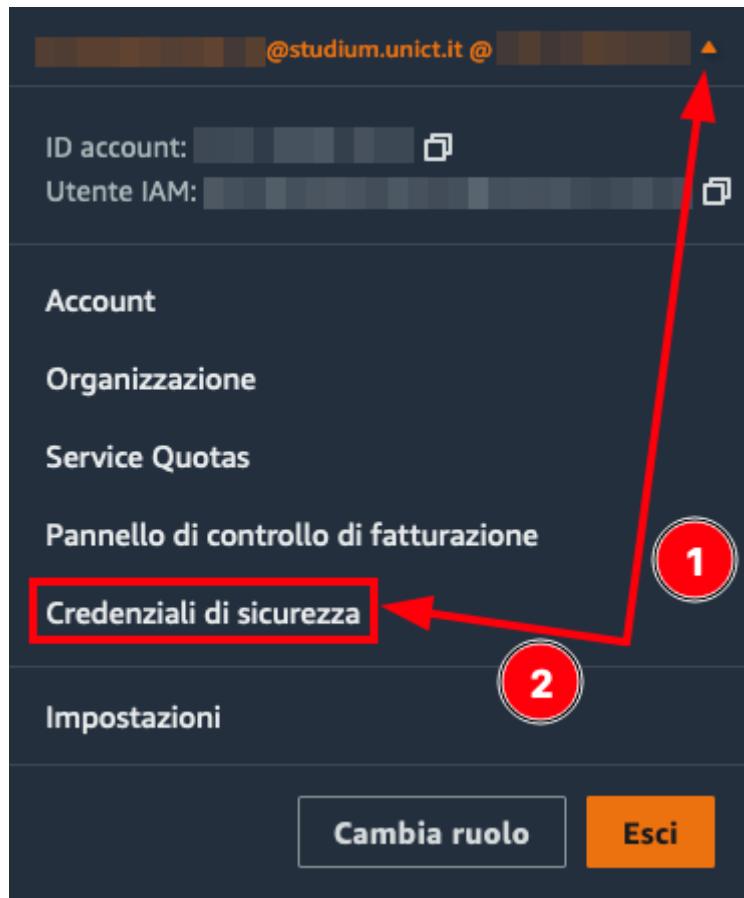
Per ruotare le chiavi di accesso per l'utente IAM, è necessario disporre delle autorizzazioni dalla policy seguente:

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Sid": "ManageOwnAccessKeys",  
            "Effect": "Allow",  
            "Action": [  
                "iam:CreateAccessKey",  
                "iam>DeleteAccessKey",  
                "iam:GetAccessKeyLastUsed",  
                "iam:GetUser",  
                "iam>ListAccessKeys",  
                "iam:UpdateAccessKey"  
            ],  
            "Resource": "arn:aws:iam::*:user/${aws:username}"  
        }  
    ]  
}
```

#### 4.3.2 Come creare, modificare o eliminare le chiavi di accesso di un utente IAM (console), [1]

È possibile utilizzare la AWS Management Console per gestire le chiavi di accesso di un utente IAM.

1. Selezionare il nome utente in alto a destra nella barra di navigazione e selezionare **My Security Credentials (Le mie credenziali di sicurezza)**.



2. Effettuare una delle seguenti operazioni:

- Per creare una chiave di accesso, selezionare **Crea nuova chiave di accesso**.

Se questa caratteristica è disattivata, è necessario eliminare una delle chiavi esistenti prima di poter creare una nuova chiave. Un avviso spiega che si ha solo questa opportunità per visualizzare o scaricare la Secret Access Key. Per copiare la chiave e incollarla da qualche altra parte per conservarla, scegli **Mostra chiave di accesso**. Quindi scegliere **Scarica il file .csv** per salvare l'ID della chiave di accesso e la chiave di accesso segreta in un file .csv in un percorso protetto sul computer.

#### Chiavi di accesso per l'interfaccia a riga di comando, l'SDK e l'accesso alle API

Utilizza le chiavi di accesso per effettuare chiamate programmatiche ad AWS dall'interfaccia a riga di comando AWS (CLI), strumenti per PowerShell, SDK di AWS o chiamate API AWS dirette. Puoi avere un massimo di due chiavi di accesso (attive o inattive) alla volta.

Per motivi di sicurezza, non condividere mai le chiavi segrete con altri. Come best practice, si suggerisce di effettuare una rotazione frequente delle chiavi.

**La chiave segreta può essere visualizzata o scaricata solo al momento della creazione. Crea una nuova chiave d'accesso se hai perso quella esistente. Ulteriori informazioni**

**Crea chiave di accesso**

ID chiave di accesso	Stato	Data creazione	Ultimo utilizzo	Operazioni
[REDACTED]	Attivo	2022-04-13 21:34 UTC+0200	N/A	Rendi inattivo   Elimina

## Crea chiave di accesso

X

✓ La nuova chiave di accesso è ora disponibile.

Questa è l'unica volta in cui la chiave di accesso segreta può essere visualizzata o scaricata.

Non è possibile recuperarla successivamente. Tuttavia, puoi creare nuove chiavi di accesso in qualsiasi momento.

 Scarica il file .csv

ID chiave di accesso 

Chiave di accesso segreta   
Nascondi chiave di accesso segreta

Chiudi

	A	B	C	D	E	F	G	H
1	Access key ID,Secret access key							
2		,						
3								

b. Per disabilitare una chiave di accesso attiva, selezionare **Rendi inattivo**.

Chiavi di accesso per l'interfaccia a riga di comando, l'SDK e l'accesso alle API

Utilizza le chiavi di accesso per effettuare chiamate programmatiche ad AWS dall'interfaccia a riga di comando AWS (CLI), strumenti per PowerShell, SDK di AWS o chiamate API AWS dirette. Puoi avere un massimo di due chiavi di accesso (attive o inattive) alla volta.

Per motivi di sicurezza, non condividere mai le chiavi segrete con altri. Come best practice, si suggerisce di effettuare una rotazione frequente delle chiavi.

La chiave segreta può essere visualizzata o scaricata solo al momento della creazione. Crea una nuova chiave d'accesso se hai perso quella esistente. Ulteriori informazioni

Chiavi di accesso				
ID chiave di accesso	Stato	Data creazione	Ultimo utilizzo	Operazioni
	Attivo	2022-04-13 21:34 UTC+0200	N/A	<a href="#">Rendi inattivo</a>   <a href="#">Elimina</a>

## Disattiva AKIAT7AMEIYOYWEB3B7R?

X

Disattivare la chiave di accesso [REDACTED]? Non puoi utilizzare una chiave disabilitata per effettuare chiamate API AWS, ma puoi attivarla nuovamente in un secondo momento.

Access key last used

Mai

IAM user

[REDACTED]@studium.unict.it

Account

[REDACTED]

[Annulla](#) [Disattiva](#)

### c. Per riabilitare una chiave di accesso inattiva, scegli **Make Active (Rendi attivo)**.

#### Chiavi di accesso per l'interfaccia a riga di comando, l'SDK e l'accesso alle API

Utilizza le chiavi di accesso per effettuare chiamate programmatiche ad AWS dall'interfaccia a riga di comando AWS (CLI), strumenti per PowerShell, SDK di AWS o chiamate API AWS dirette. Puoi avere un massimo di due chiavi di accesso (attive o inattive) alla volta.

Per motivi di sicurezza, non condividere mai le chiavi segrete con altri. Come best practice, si suggerisce di effettuare una rotazione frequente delle chiavi.

**La chiave segreta può essere visualizzata o scaricata solo al momento della creazione. Crea una nuova chiave d'accesso se hai perso quella esistente.** [Ulteriori informazioni](#)

[Crea chiave di accesso](#)

ID chiave di accesso	Stato	Data creazione	Ultimo utilizzo	Operazioni
[REDACTED]	Inattivo	2022-04-13 21:34 UTC+0200	N/A	<a href="#">Rendi attivo</a> <a href="#">Elimina</a>

#### Attiva chiave di accesso

X

**Sei sicuro di voler attivare questa chiave di accesso?**

qualsiasi codice o sistema che le utilizza non sarà in grado di chiamare le API AWS. È possibile riattivare questa chiave in un secondo momento, se necessario.

[Annulla](#) [Rendi attivo](#)

### d. Per eliminare la chiave di accesso, scegli **Elimina**. AWS consiglia di disattivare la chiave e verificare che non sia più in uso prima di eseguire questa operazione. Quando si utilizza la AWS Management Console, è necessario disattivare la chiave prima di eliminarla.

## Chiavi di accesso per l'interfaccia a riga di comando, l'SDK e l'accesso alle API

Utilizza le chiavi di accesso per effettuare chiamate programmatiche ad AWS dall'interfaccia a riga di comando AWS (CLI), strumenti per PowerShell, SDK di AWS o chiamate API AWS dirette. Puoi avere un massimo di due chiavi di accesso (attive o inattive) alla volta.

Per motivi di sicurezza, non condividere mai le chiavi segrete con altri. Come best practice, si suggerisce di effettuare una rotazione frequente delle chiavi.

**La chiave segreta può essere visualizzata o scaricata solo al momento della creazione. Crea una nuova chiave d'accesso se hai perso quella esistente. Ulteriori informazioni**

Crea chiave di accesso

ID chiave di accesso	Stato	Data creazione	Ultimo utilizzo	Operazioni
[REDACTED]	Inattivo	2022-04-13 21:34 UTC+0200	N/A	Rendi attivo   Elimina

Eliminare AKIAT7AMEIYOYWEB3B7R?

Eliminare definitivamente la chiave di accesso [REDACTED] ? Tutte le chiamate API AWS effettuate utilizzando questa chiave non verranno eseguite. Prima di disabilitare o eliminare una chiave di accesso, accertati che non sia più in uso. Non puoi recuperare una chiave di accesso dopo averla eliminata.

Access key last used

⚠ Mai

IAM user

[REDACTED] @studium.unict.it

Account

1 [REDACTED]  
2 [REDACTED]

Per confermare l'eliminazione, immetti l'ID chiave di accesso nel campo di input di testo.

Annula Elimina

[Reimpostazione delle password o delle chiavi di accesso perse o dimenticate per AWS.](#)

### 4.4 Utilizzo dell'autenticazione a più fattori (MFA) in AWS, [1]

L'autenticazione MFA garantisce una maggiore sicurezza poiché richiede agli utenti di fornire autenticazione univoca da un meccanismo MFA supportato da AWS, in aggiunta alle normali credenziali di accesso, quando questi ultimi accedono ai siti Web o ai servizi di AWS:

- **Dispositivi MFA virtuali.** Un'app software in esecuzione su un cellulare o altro dispositivo e che emula un dispositivo fisico. Il dispositivo genera un codice numerico di sei cifre basato su un algoritmo di password monouso sincronizzato nel tempo. L'utente deve immettere un codice valido dal dispositivo su una seconda pagina Web durante la procedura di accesso. Ogni dispositivo MFA virtuale assegnato a un utente deve essere univoco. Per eseguire l'autenticazione, gli utenti non possono digitare un codice generato dal dispositivo MFA virtuale di un altro utente. Poiché possono essere eseguiti su dispositivi mobili non sicuri, i dispositivi MFA virtuali potrebbero non offrire lo stesso livello di sicurezza dei dispositivi U2F o dei dispositivi MFA hardware. È consigliabile

utilizzare un dispositivo MFA virtuale nell'attesa dell'approvazione di un acquisto hardware o della consegna del dispositivo hardware.

- **Chiave di sicurezza U2F.** Un dispositivo che è possibile collegare alla porta USB del computer. U2F è uno standard di autenticazione aperto ospitato dalla FIDO Alliance. Se abiliti una chiave di sicurezza U2F, puoi effettuare l'accesso inserendo le credenziali e toccando il dispositivo anziché inserire manualmente un codice.
- **Dispositivo hardware MFA.** Un dispositivo hardware che genera un codice numerico di sei cifre basato su un algoritmo di password monouso sincronizzato nel tempo. L'utente deve immettere un codice valido dal dispositivo su una seconda pagina Web durante la procedura di accesso. Ogni dispositivo MFA assegnato a un utente deve essere univoco. Per essere autenticati, gli utenti non possono digitare un codice generato dal dispositivo di un altro utente.
- **Autenticazione MFA basata su SMS.** Un tipo di autenticazione MFA in cui le impostazioni dell'utente IAM includono il numero di telefono del dispositivo mobile compatibile con gli SMS dell'utente. Quando l'utente esegue l'accesso, AWS invia un SMS contenente un codice numerico di sei cifre al dispositivo mobile dell'utente. L'utente deve immettere il codice ricevuto in una seconda pagina Web durante la procedura di accesso. Nota che l'autenticazione MFA basata su SMS è disponibile solo per gli utenti IAM. Non è possibile utilizzare questo tipo di MFA con l'utente root dell'Account AWS.

#### 4.4.1 Come abilitare un dispositivo MFA virtuale per un utente IAM (console), [1]

Puoi utilizzare un telefono o un altro dispositivo come dispositivo Multi-Factor Authentication (MFA) virtuale.

La maggior parte delle applicazioni MFA virtuali supportano la creazione di più dispositivi virtuali, che consente di utilizzare la stessa app per più account o utenti AWS. Tuttavia, è possibile abilitare un solo dispositivo MFA per utente.

1. Nel pannello di navigazione, seleziona Users (Utenti).



2. Nell'elenco **Nome utente**, selezionare il nome dell'utente MFA in questione.

## Utenti (Selezionati 1/20) [Informazioni](#)

Un utente IAM è un'identità con credenziali a lungo termine che viene utilizzata per interagire con AWS in un account.

[Aggiungi utenti](#)



Elimina

<input checked="" type="checkbox"/>	Nome utente	Gruppi	Ultima attività	MFA	Validità pas
<input checked="" type="checkbox"/>	[REDACTED]	Developers2022	3 ore fa	Nessuno	21 giorni

3. Selezionare la scheda **Security Credentials (Credenziali di sicurezza)**. Accanto ad **Assigned MFA device (Dispositivo MFA assegnato)**, selezionare **Gestione**.

## Riepilogo

[Elimina utente](#)



ARN utente arn:aws:iam::[REDACTED]@studium.unict.it

Percorso /

Data di creazione 2022-03-23 16:00 UTC+0200

Autorizzazioni Gruppi (1) Tag (1) Credenziali di sicurezza Consulente accessi

**Credenziali di accesso**

Riepilogo • Collegamento di accesso alla console: https://[REDACTED].signin.aws.amazon.com/console

Password console Abilitata (ultimo accesso Oggi) | [Gestione](#)

Dispositivo MFA assegnato Non assegnato | [Gestione](#)

Certificati di firma Nessuna

1

2

4. Nella procedura guidata **Manage MFA Device (Gestisci dispositivo MFA)**, selezionare **Virtual MFA device (Dispositivo MFA virtuale)** e scegliere **Continua**.

Gestione dispositivo MFA

Scegli il tipo di dispositivo MFA da assegnare:

**Dispositivo MFA virtuale**  
L'app Authenticator è installata sul tuo dispositivo mobile o computer

**Chiave di sicurezza U2F**  
YubiKey o qualsiasi altro dispositivo conforme a U2F

**Altro dispositivo MFA basato su hardware**  
Token Gemalto

Per ulteriori informazioni sui dispositivi MFA supportati, consulta [AWS Multi-Factor Authentication](#)

[Annulla](#) [Continua](#)

1

2

IAM genera e visualizza le informazioni di configurazione per il dispositivo MFA virtuale, tra cui il codice grafico QR. Il grafico è una rappresentazione della "chiave di configurazione segreta" disponibile per l'inserimento manuale sui dispositivi che non supportano i codici QR.

5. Aprire l'app MFA virtuale.
6. Determinare se l'app MFA supporta i codici QR e procedere in uno dei seguenti modi:

- a. Nella procedura guidata, scegliere **Show QR code (Mostra codice QR)** ed eseguire la scansione del codice QR tramite l'app. Ad esempio, è possibile selezionare l'icona della fotocamera o un'opzione simile a **Scan code (Scannerizza codice)** ed eseguire la scansione del codice tramite la fotocamera del dispositivo.
- b. Nella procedura guidata **Manage MFA Device (Gestisci dispositivo MFA)**, selezionare **Show secret key (Mostra chiave segreta)** e digitare la chiave segreta nell'app MFA.

Al termine, il dispositivo MFA virtuale avvia la generazione di password una tantum.

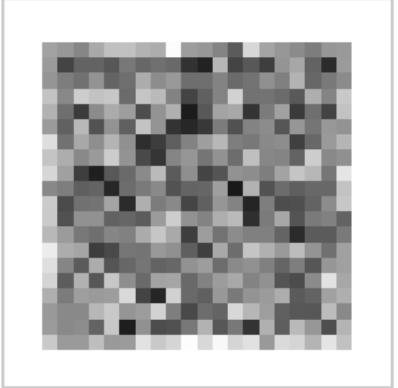
Imposta dispositivo MFA virtuale ×

---

1. **Installa un'app compatibile sul tuo dispositivo mobile o computer**  
 Vedi un [elenco di applicazioni compatibili](#)

---

2. **Usa l'app MFA virtuale e la videocamera del tuo dispositivo per analizzare il codice QR**


×

In alternativa, puoi digitare la chiave segreta. [Mostra chiave segreta](#)

7. Nella procedura guidata **Manage MFA Device (Gestisci dispositivo MFA)**, nella casella **MFA code 1 (Codice MFA 1)**, digitare la password monouso visualizzata nel dispositivo MFA virtuale. Attendere fino a un massimo di 30 secondi prima che il dispositivo generi una nuova password una tantum. Quindi, digitare la seconda password monouso nella casella **MFA code 2 (Codice MFA 2)**. Scegliere **Assign MFA (Assegna MFA)**.

---

3. **Digita di seguito due codici MFA consecutivi**

Codice MFA	<input type="text"/>
1	
Codice MFA	<input type="text"/>
2	

Annulla
Precedente
Assegna MFA

#### 4.5 Policy e autorizzazioni in IAM, [1], [esempi di policy]

È possibile gestire l'accesso in AWS mediante la creazione di policy e il relativo collegamento a identità IAM (utenti, gruppi di utenti o ruoli) o risorse AWS. Una policy è un oggetto in AWS che, quando associato a un'identità o a una risorsa, ne definisce le autorizzazioni. AWS valuta queste policy quando un principale IAM (utente o ruolo) effettua una richiesta. Le autorizzazioni

nella policy determinano l'approvazione o il rifiuto della richiesta. La maggior parte delle policy viene archiviata in AWS come documenti JSON. AWS supporta sei tipi di policy: policy basate su identità, policy basate su risorse, e altre.

Le policy IAM definiscono le autorizzazioni relative a un'operazione, indipendentemente dal metodo utilizzato per eseguirla. Ad esempio, se una policy consente l'operazione GetUser, un utente con tale policy può ottenere le informazioni utente dalla AWS Management Console, dall'AWS CLI o dall'API AWS.

La **logica di valutazione** per una richiesta all'interno di un singolo account segue queste regole:

- Per impostazione predefinita, tutte le richieste vengono negate implicitamente (In alternativa, per impostazione predefinita, l'utente root dell'Account AWS ha accesso completo).
- Un'autorizzazione esplicita in una policy basata su identità o basata su risorse sostituisce questa impostazione predefinita.
- Se è presente un limite delle autorizzazioni, una SCP di Organizations oppure una policy di sessione, potrebbe sovrascrivere l'autorizzazione con un rifiuto隐式的.
- Un rifiuto esplicito in una policy sostituisce qualsiasi permesso.

### Tipi di policy

I tipi di policy elencati di seguito in ordine da quello utilizzato più frequentemente a quello meno frequentemente sono disponibili per l'uso in AWS.

#### 4.5.1 Policy basate su identità

Le policy basate su identità sono documenti di policy di autorizzazione JSON che controllano quali operazioni un'identità (utenti, gruppi di utenti e ruoli) può eseguire, su quali risorse e in quali condizioni. Le policy basate su identità possono essere ulteriormente suddivise:

- **Policy gestite**: le policy autonome basate sulle identità che possono essere collegate a più utenti, gruppi o ruoli nell'account AWS. Sono disponibili due tipi di policy gestite:
  - **Policy gestite da AWS**: le policy gestite che sono create e gestite da AWS.
  - **Policy gestite dal client**: le policy gestite che sono create e gestite nell'account AWS. Le policy gestite dal cliente forniscono un controllo più preciso rispetto a quelle gestite da AWS.
- **Policy in linea**: le policy che vengono aggiunte direttamente a un singolo utente, gruppo o ruolo. Le policy in linea sono utili per mantenere una stretta relazione uno a uno tra una policy e un'identità. Vengono eliminate quando elimini l'identità.

#### 4.5.2 Policy basate su risorse

Le policy basate sulle risorse sono documenti di policy JSON che collega a una risorsa, come ad esempio un bucket Amazon S3. Queste policy concedono all'entità principale specificata l'autorizzazione per eseguire operazioni specifiche sulla risorsa e definiscono le condizioni in cui ciò si applica. Le policy basate su risorse sono policy inline. Non esistono policy basate su risorse gestite.

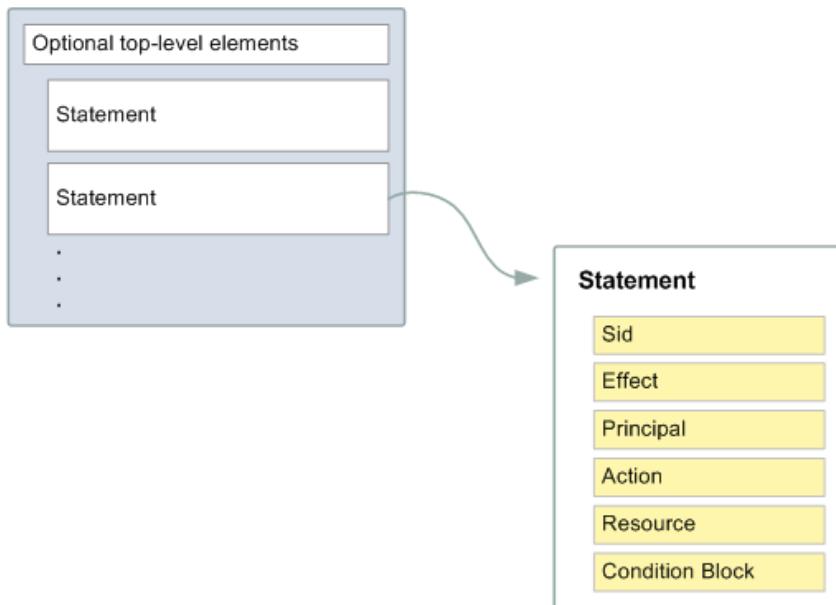
#### 4.5.3 Struttura dei documenti di policy JSON, [1], [2 specifico]

La maggior parte delle policy viene memorizzata in AWS sotto forma di documenti JSON. Non è necessario conoscere la sintassi JSON. Grazie all'editor grafico in AWS Management Console, puoi creare e modificare le policy gestite del cliente senza utilizzare JSON.

Come illustrato nella figura di seguito, un documento di policy JSON include questi elementi:

- Informazioni opzionali sulla policy nella parte superiore del documento;
- Una o più istruzioni singole.

Ogni istruzione include informazioni su una singola autorizzazione. Se una policy include più istruzioni, AWS applica un operatore logico OR tra le istruzioni al momento della valutazione. Se a una richiesta possono essere applicate più policy, AWS inserisce un operatore logico OR tra tutte le policy al momento della valutazione.



Le informazioni di un'istruzione sono contenute all'interno di una serie di elementi.

- **Version**. Specifica la versione del linguaggio di policy che desideri utilizzare. Come best practice, è consigliabile usare la versione più recente **2012-10-17**.
- **Statement**. Utilizza questo elemento principale della policy come container per i seguenti elementi. Puoi includere più istruzioni in una policy.
- **Sid (facoltativo)**: includi un ID istruzione opzionale per distinguere le varie istruzioni.
- **Effect**: utilizza **Allow** o **Deny** per indicare se la policy consente l'accesso o lo rifiuta.
- **Principal** (obbligatorio solo in alcune circostanze): se crei una policy basata sulle risorse, devi indicare l'account, l'utente, il ruolo o l'utente federato a cui desideri consentire o rifiutare l'accesso. Nella creazione di una policy di autorizzazioni IAM da collegare a un utente o un ruolo, non è possibile includere questo elemento. L'entità principale è implicita come l'utente o il ruolo.
- **Action**. Includi un elenco delle operazioni consentite o rifiutate dalla policy.
- **Resource** (obbligatorio solo in alcune circostanze): se crei una policy di autorizzazioni IAM, devi specificare un elenco di risorse a cui si applicano le operazioni. Se crei una policy basata sulle risorse, questo elemento è facoltativo. Se non includi questo elemento, la risorsa a cui si applica l'operazione è la risorsa a cui è collegata la policy.
- **Condition** (facoltativo): specifica le circostanze in base alle quali la policy concede l'autorizzazione (per esempio: ora del giorno, indirizzi ip ammessi, presenza di tag nelle risorse, ecc..).

#### 4.6 Creazione di policy IAM, [1]

È possibile creare una policy gestita dal cliente nella AWS Management Console utilizzando uno dei seguenti metodi:

- **JSON:** incolla e personalizza un esempio di policy basata sull'identità;
- **Editor visivo:** è possibile creare una nuova policy da zero nell'editor visivo. Se si utilizza l'editor visivo, non è necessario comprendere la sintassi JSON;
- **Importa:** importa e personalizza una policy gestita dall'account. È possibile importare una policy AWS gestita o una policy gestita dal cliente creato in precedenza.

#### 4.6.1 Creazione di policy nella scheda JSON, [1]

1. Nel riquadro di navigazione a sinistra, selezionare **Policies (Policy)**. Scegliere **Create Policy (Crea policy)**.

The screenshot shows the AWS IAM Policies page. On the left sidebar, under 'Policy', there is a red circle with the number '1'. At the top right, there is a red circle with the number '2' next to the 'Crea policy' button. The main area displays a table of existing policies with columns for 'Nome della policy' and 'Tipo'.

Nome della policy	Tipo
AmazonSageMaker-ExecutionPolicy-20200630T113926	Gestite dal cliente
AmazonSageMakerServiceCatalogProductsUseRole-20210519T175236	Gestite dal cliente
aws-full-log	Gestite dal cliente
AWSLambdaBasicExecutionRole-326b82bc-ea69-44ce-a8ad-e23e3a7d9962	Gestite dal cliente

2. Scegliere la scheda **JSON**. Digitare o incollare un documento di policy JSON.
3. Risolvi eventuali avvisi di sicurezza, errori o avvisi generali generati durante la convalida delle policy quindi scegli **Rivedi policy**. Quando hai terminato, seleziona **Next: Tags (Successivo: Tag)**. (Facoltativo) Aggiungere metadati alla policy collegando i tag come coppie chiave-valore.

Una policy definisce le autorizzazioni AWS che è possibile assegnare a un utente, un gruppo o un ruolo. È possibile creare e modificare una policy nell'editor visivo e utilizzando JSON. [Ulteriori informazioni](#)

The screenshot shows the 'Create Policy' JSON editor. A red box highlights the JSON tab, which has a red circle with the number '1'. Below it, the JSON code is shown in a code editor with a red circle '2' highlighting the 'Statement' field. At the bottom, there are status indicators: Sicurezza: 0, Errori: 0, Avvisi: 0, Suggerimenti: 0.

Numero di caratteri: 39 di 6.144.

[Annulla](#)

[Successivo: Tag](#)

4. Nella pagina **Review policy (Esamina policy)**, digitare un **Name (Nome)** e una **Description (Descrizione)** (facoltativa) per la policy in fase di creazione. Consulta il **Summary (Riepilogo)** della policy per visualizzare le autorizzazioni concesse dalla policy. Seleziona **Create policy (Crea policy)** per salvare il proprio lavoro.

Dopo aver creato una policy, è possibile collegarla ai gruppi, utenti o ruoli.

#### 4.6.2 Creazione di policy con l'editor visivo, [1]

- Nel riquadro di navigazione a sinistra, selezionare **Policies (Policy)**. Scegliere **Create Policy (Crea policy)**.

The screenshot shows the AWS IAM Policies page. On the left sidebar, under 'Identity and Access Management (IAM)', the 'Policy' option is selected and highlighted with a red box and a number '1'. At the top right, there is a 'Crea policy' button with a red box and a number '2'.

Nome della policy	Tipo
AmazonSageMaker-ExecutionPolicy-20200630T113926	Gestite dal cliente
AmazonSageMakerServiceCatalogProductsUseRole-20210519T175236	Gestite dal cliente
aws-full-log	Gestite dal cliente
AWSLambdaBasicExecutionRole-326b82bc-ea69-44ce-a8ad-e23e3a7d9962	Gestite dal cliente

- Nella scheda **Editor visivo** selezionare **Scegli un servizio**, quindi scegliere un servizio AWS. È possibile utilizzare la casella di ricerca in alto per limitare i risultati nell'elenco di servizi. È possibile selezionare solo un servizio nel blocco di autorizzazione di un editor visivo. Per concedere l'accesso a più di un servizio, aggiungere più blocchi di autorizzazioni selezionando **Add additional permissions (Aggiungi ulteriori autorizzazioni)**.

The screenshot shows the AWS IAM Visual Editor interface. At the top left, the 'Editor visivo' tab is selected and highlighted with a red box and a number '1'. At the top right, there is a 'Importa policy gestita' button. Below the tabs, there is an 'Espandi tutto' (Expand all) button with a red box and a number '1'.

**Selezione di seguito un servizio**

**Servizio**  2 **Immetti il servizio manualmente**

**Operazioni** Scegli un servizio prima di definire le operazioni

**Risorse** Scegli le operazioni prima di applicare le risorse

**Condizioni della richiesta** Scegli le operazioni prima di specificare le condizioni

**Aggiungi altre autorizzazioni**

- Per **Operazioni**, scegliere le operazioni da aggiungere al criterio. È possibile selezionare operazioni nei modi seguenti:
  - Selezionare la casella di controllo per tutte le operazioni.

▼ Operazioni chiudi Specifica le operazioni consentite in EC2 ⓘ Cambia per negare le autorizzazioni ⓘ

Filtra operazioni

Operazioni manuali (aggiungi operazione)

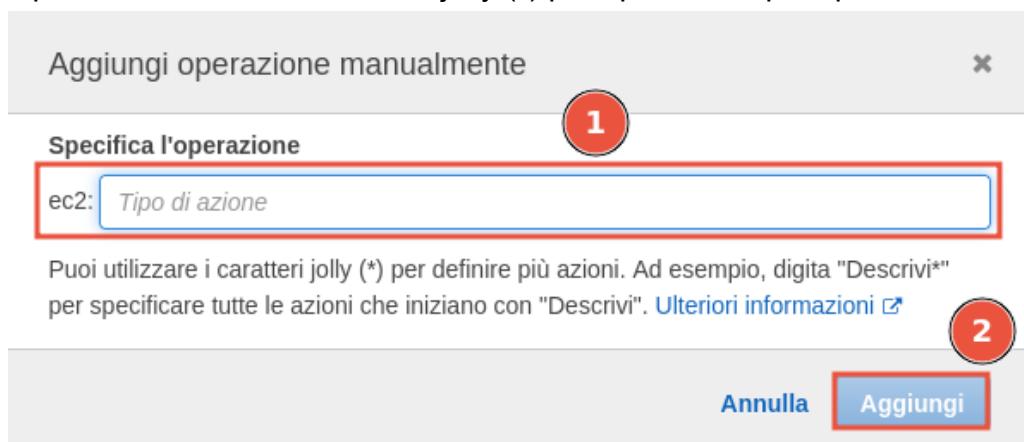
Tutte le operazioni EC2 (ec2:\*)

Livello di accesso

- ▶  Elenco (139 selezionati)
- ▶  Lettura (37 selezionati)
- ▶  Applicazione di tag (2 selezionati)
- ▶  Scrittura (348 selezionati)
- ▶  Gestione autorizzazioni (5 selezionati)

Espandi tutto | Comprimi tutto

- Scegliere **aggiungi operazione** per digitare il nome di un'operazione specifica. È possibile utilizzare i caratteri jolly (\*) per specificare più operazioni.



- Selezionare uno dei gruppi di livelli di accesso per scegliere tutte le azioni per il livello di accesso, ad esempio Lettura, Scrittura o Elenco.

▼ Operazioni chiudi Specifica le operazioni consentite in EC2 ⓘ Cambia per negare le autorizzazioni ⓘ

Filtra operazioni

Operazioni manuali (aggiungi operazione)

Tutte le operazioni EC2 (ec2:\*)

Livello di accesso

- ▶  Elenco (139 selezionati)
- ▶  Lettura (37 selezionati)
- ▶  Applicazione di tag
- ▶  Scrittura
- ▶  Gestione autorizzazioni

Espandi tutto | Comprimi tutto

- Espandere ciascuno dei gruppi **Access level (Livello di accesso)** per selezionare singole operazioni.

▼ Operazioni Specifica le operazioni consentite in EC2 ⓘ Cambia per negare le autorizzazioni ⓘ

chiudi

Filtra operazioni

Operazioni manuali (aggiungi operazione)

Tutte le operazioni EC2 (ec2:\*)

Livello di accesso

▶  Elenco

▶  Lettura

▶  Aplicazione di tag

Espandi tutto | Comprimi tutto

▶  CreateTags ⓘ

▶  DeleteTags ⓘ

▶  Scrittura

▶  Gestione autorizzazioni

Come impostazione predefinita, la policy che si sta creando utilizza le operazioni selezionate. Per rifiutare invece le operazioni scelte, selezionare **Cambia per negare le autorizzazioni**. Poiché IAM rifiuta per impostazione predefinita, si consiglia come best practice di sicurezza di consentire le autorizzazioni solo alle operazioni e alle risorse necessarie per un utente. È necessario creare un'istruzione JSON per negare le autorizzazioni solo se si desidera sostituire un'autorizzazione separatamente consentita da un'altra istruzione o policy. Si consiglia di limitare al minimo il numero di autorizzazioni di rifiuto perché possono aumentare la difficoltà di risoluzione dei problemi relativi alle autorizzazioni.

4. Per **Risorse**, se il servizio e le azioni selezionati nei passaggi precedenti non supportano la scelta di risorse specifiche, tutte le risorse sono consentite e non è possibile modificare questa sezione.

Se si selezionano una o più operazioni che supportano le autorizzazioni a livello di risorsa, l'editor visivo elenca tali risorse. È possibile selezionare Risorse per specificare le risorse per la policy.

È possibile specificare le risorse nei seguenti modi:

- Selezionare **Aggiungi ARN** per specificare le risorse in base al loro Amazon Resource Name (ARN). È possibile utilizzare l'editor ARN visivo o elencare manualmente gli ARN.
- Scegliere **Qualsiasi** accanto a una risorsa per concedere autorizzazioni a qualsiasi risorsa di quel tipo.
- Selezionare **Tutte le risorse** per selezionare tutte le risorse per quel servizio.

**Risorse**  Specifico  Tutte le risorse **chiudi**

capacity-reservati...	Non hai specificato la risorsa con tipo <b>capacity-reservation</b> <a href="#">Aggiungi l'ARN</a> di limitare l'accesso	<input type="checkbox"/> Qualsiasi in questo account
elastic-gpu	Non hai specificato la risorsa con tipo <b>elastic-gpu</b> <a href="#">Aggiungi l'ARN</a> di limitare l'accesso	<input type="checkbox"/> Qualsiasi in questo account
elastic-inference	Non hai specificato la risorsa con tipo <b>elastic-inference</b> <a href="#">Aggiungi l'ARN</a> di limitare l'accesso	<input type="checkbox"/> Qualsiasi in questo account
group	Non hai specificato la risorsa con tipo <b>group</b> <a href="#">Aggiungi l'ARN</a> di limitare l'accesso	<input type="checkbox"/> Qualsiasi in questo account
image	Specifica ARN della risorsa <b>image</b> per <b>RunInstances</b> operazione. <a href="#">Aggiungi l'ARN</a> di limitare l'accesso	<input type="checkbox"/> Qualsiasi

5. (Opzionale) Scegliere **Condizioni della richiesta** per aggiungere condizioni alla policy che si sta creando. Le condizioni limitano l'effetto di una dichiarazione di policy JSON. Ad esempio, puoi specificare che un utente può eseguire le operazioni sulle risorse solo quando la richiesta dell'utente viene effettuata entro un determinato intervallo di tempo. È inoltre possibile utilizzare le condizioni comuni per limitare se un utente deve essere autenticato utilizzando un dispositivo multi-factor authentication (MFA). In alternativa, è possibile richiedere che la richiesta provenga da un determinato intervallo di indirizzi IP. È possibile selezionare le condizioni nei modi seguenti:

- Utilizzare le caselle di controllo per selezionare le condizioni di utilizzo comune.

**Condizioni della richiesta** **chiudi**

<input checked="" type="checkbox"/> <b>MFA obbligatorio</b> Richiede che gli utenti della console e quelli con credenziali temporanee effettuino l'autenticazione con un dispositivo MFA per queste operazioni. <a href="#">Ulteriori informazioni</a>
<input checked="" type="checkbox"/> <b>IP di origine</b> Consenti l'accesso alle operazioni specificate solo quando la richiesta proviene da un determinato intervallo di indirizzi IP.  <input type="text"/> <b>Intervallo IP</b> Esempio: 210.75.12.75/16  <a href="#">+ Aggiungi un altro intervallo IP</a>

[Aggiungi condizione](#)

- Selezionare **Add condition (Aggiungi condizione)** per specificare altre condizioni. Selezionare **Condition Key (Chiave condizione)**, **Qualifier (Qualificatore)** e **Operator (Operatore)** della condizione e digitare un **Value (Valore)**. Per aggiungere più di un valore, selezionare **Aggiungi un altro valore della condizione**. È possibile valutare i valori come se fossero connessi da un operatore logico "OR". Al termine, selezionare **Add (Aggiungi)**. Per aggiungere più di una condizione, selezionare **Add condition (Aggiungi condizione)**. Ripetere come necessario. Ogni condizione si applica solo a questo blocco di autorizzazione di un editor visivo. Tutte le condizioni devono essere vere

per il blocco di autorizzazioni per essere considerato una corrispondenza. In altre parole, considerare le condizioni da connettere con un operatore logico "AND".

Aggiungi condizione della richiesta

Chiave di condizione: 1 aws:CalledVia

Qualificatore: 2 Per qualsiasi valore nella richiesta

Operatore: 3 StringEquals  Se esiste

Valore: 4

5 + Aggiungi un altro valore della condizione

6 Annulla Aggiungi

6. Per aggiungere più blocchi di autorizzazioni, selezionare **Add additional permissions (Aggiungi altre autorizzazioni)**. Per ogni blocco, ripetere le fasi da 1 a 3.
7. Quando hai terminato, seleziona **Next: Tags (Successivo: Tag)**. (Facoltativo) Aggiungere metadati alla policy collegando i tag come coppie chiave-valore. Al termine, selezionare **Next: Review (Successivo: Verifica)**.

#### Aggiungi tag (Facoltativo)

I tag sono coppie chiave-valore che è possibile aggiungere alle risorse AWS per facilitare l'identificazione, l'organizzazione o la ricerca di risorse.

Nessun tag associato alla risorsa.

1 Aggiungi tag

Puoi aggiungere fino a 50 altri tag

2 Annulla Precedente Successivo: Verifica

8. Nella pagina **Verifica policy**, digitare un **Name (Nome)** e una **Description (Descrizione)** (facoltativa) per la policy in fase di creazione. Esaminare il riepilogo della policy per assicurarsi di aver concesso le autorizzazioni corrette e selezionare **Create policy (Crea policy)** per salvare la nuova policy.

Verifica policy

Nome*	<input type="text"/> 1																
Utilizza caratteri alfanumerici e i seguenti simboli: '+=.,@-_ ' Massimo 128 caratteri.																	
Descrizione	<input type="text"/> 2																
Massimo 1000 caratteri. Utilizza caratteri alfanumerici e i seguenti simboli: '+=.,@-_ '																	
Riepilogo	<table border="1"> <thead> <tr> <th colspan="4">Filtro:</th> </tr> <tr> <th>Servizio</th> <th>Livello di accesso</th> <th>Risorsa</th> <th>Condizione della richiesta</th> </tr> </thead> <tbody> <tr> <td>Consenti (1 dei servizi 323) Mostra 322 rimanenti</td> <td>EC2</td> <td>Limitato: Scrittura</td> <td>Tutte le risorse</td> </tr> <tr> <td></td> <td></td> <td></td> <td>aws:MultiFactorAuthPresent   Bool   true (If Exists)</td> </tr> </tbody> </table>	Filtro:				Servizio	Livello di accesso	Risorsa	Condizione della richiesta	Consenti (1 dei servizi 323) Mostra 322 rimanenti	EC2	Limitato: Scrittura	Tutte le risorse				aws:MultiFactorAuthPresent   Bool   true (If Exists)
Filtro:																	
Servizio	Livello di accesso	Risorsa	Condizione della richiesta														
Consenti (1 dei servizi 323) Mostra 322 rimanenti	EC2	Limitato: Scrittura	Tutte le risorse														
			aws:MultiFactorAuthPresent   Bool   true (If Exists)														
Tag	<table border="1"> <thead> <tr> <th>Chiave</th> <th>Valore</th> </tr> </thead> <tbody> <tr> <td colspan="2">Nessun tag associato alla risorsa.</td> </tr> </tbody> </table>	Chiave	Valore	Nessun tag associato alla risorsa.													
Chiave	Valore																
Nessun tag associato alla risorsa.																	

\* Campo obbligatorio Annulla Precedente Crea policy 3

Dopo aver creato una policy, è possibile collegarla ai gruppi, utenti o ruoli.

#### 4.6.3 Aggiunta e rimozione di autorizzazioni per identità IAM, [1]

Puoi utilizzare le policy per definire le autorizzazioni per una identità (utente, gruppo di utenti o ruolo). Puoi aggiungere o rimuovere autorizzazioni collegando o scollegando policy IAM per un'identità tramite la AWS Management Console, AWS Command Line Interface (AWS CLI) oppure l'API AWS.

- Nel pannello di navigazione, selezionare **Policies (Policy)**. Nell'elenco di policy selezionare la casella di controllo accanto al nome della policy da collegare. Puoi utilizzare la casella di ricerca per filtrare l'elenco delle policy.

The screenshot shows the AWS IAM Policies page. On the left, there's a sidebar with links like 'Ricerca IAM', 'Pannello di controllo', 'Gestione degli accessi', 'Utenti', 'Ruoli', and 'Policy'. The 'Policy' link is highlighted with a red box and a '1' circle. In the main area, there's a search bar with the placeholder 'Ricerca IAM' and a filter bar with the text '"policytag"'. A red box highlights the search bar and a '2' circle is placed on it. Below the search bar, there's a table with columns 'Nome della policy' and 'Tipo'. One row in the table has a red box around the 'PolicyTag' link and a '3' circle is placed on it. The top right of the page has buttons for 'Operazioni' and 'Crea policy'.

- Scegli **Utilizzo della policy** e seleziona **Collega**.

Autorizzazioni    Utilizzo della policy **1**    Tag    Versioni della policy    Consulente accessi

▼ Autorizzazioni

Collega **2**   Scollega

Filtro Filtro: **2**   Cerca: Visualizzazione di 0 risultati

<input type="checkbox"/> Nome	Tipo
Nessun risultato	

▶ Limiti di autorizzazioni

3. Seleziona una o più identità per collegarle alla policy. Puoi usare la casella di ricerca per filtrare l'elenco delle entità principali. Dopo aver selezionato le identità, scegliere **Attach policy (Collega policy)**.

### Collega policy

Attachez la stratégie à des utilisateurs, des groupes ou des rôles figurant dans votre compte.

Filtro Filtro: **1**   Cerca: Visualizzazione di 6 risultati

<input type="checkbox"/> Nome	Tipo
<input type="checkbox"/>	Ruolo
<input type="checkbox"/>	Ruolo
<input type="checkbox"/>	Gruppo
<input type="checkbox"/>	Gruppo
<input checked="" type="checkbox"/> Developers2022 <b>2</b>	Gruppo
<input type="checkbox"/>	Gruppo

**3**   Annula   **Collega policy**

Naturalmente è possibile anche assegnare una policy selezionando prima un gruppo o utente e successivamente "attaccargli" una policy. Vediamo per un gruppo:

1. Nel pannello di navigazione, seleziona **Gruppi di utenti**. Nell'elenco, scegli il nome del gruppo di utenti in cui integrare una policy.

IAM > Gruppi di utenti

**Gruppi di utenti (4) Informazioni**  
Un gruppo di utenti è una raccolta di utenti IAM. Utilizzare i gruppi per specificare le autorizzazioni per una raccolta di utenti.

Filtra i gruppi di utenti in base alla proprietà o al nome del gruppo e premi Invio

1 corrispondenza < 1 > ⌂

"2022" X Annulla filtri

Nome del gruppo	Utenti	Autorizza
Developers2022	13	Definito

2. Seleziona la scheda **Autorizzazioni**, quindi **Aggiungi autorizzazioni** e scegli una delle seguenti operazioni:

- Collega policy. Collega una policy gestita dal cliente o gestita da AWS.
- Crea policy inline. Scegli una delle seguenti operazioni:
  - Seleziona la scheda Editor visivo per creare la policy.
  - Scegli la scheda JSON per creare la policy. Per ulteriori informazioni, consulta . Creazione di policy nella scheda JSON.
  - Al termine, scegliere **Create policy (Crea policy)**.

Ricerca IAM

Pannello di controllo

Gestione degli accessi

**Gruppi di utenti** 1

Utenti Ruoli Policy Provider di identità Impostazioni account Report di accesso

Utenti Autorizzazioni Consulente accessi

**Policy di autorizzazione (5) Informazioni**  
Puoi collegare fino a 10 policy gestite.

Filtra le policy in base alla proprietà o al nome della policy e premi Invio

Nome della policy	Tipo	Descrizione
AmazonEC2FullAccess	Gestite da AWS	Provides full access to Amazon EC2 via the AWS Management Console.
IAMFullAccess	Gestite da AWS	Provides full access to IAM via the AWS Management Console.
AmazonS3FullAccess	Gestite da AWS	Provides full access to all buckets via the AWS Management Console.
AmazonVPCFullAccess	Gestite da AWS	Provides full access to Amazon VPC via the AWS Management Console.
EC2InstanceConnect	Gestite da AWS	Allows customers to call EC2 Instance Connect to publish ephemeral keys to their EC2 inst...

Come per i gruppi anche agli utenti è possibile attaccare policy:

Pannello di controllo

Gestione degli accessi

**Utenti** 1

Ruoli Policy Provider di identità Impostazioni account Report di accesso Analizzatore di accessi

Autorizzazioni Gruppi (1) Tag (1) Credenziali di sicurezza Consulente accessi

Policy di autorizzazione (6 policy applicate)

Aggiungi autorizzazioni 3

Nome policy ▾

Collegate direttamente

IAMUserChangePassword Policy gestita da AWS

Collegata dal gruppo

Mostra altri 5

Aggiungi policy incorporata

#### 4.7 Amazon Resource Names (ARN), [1]

Amazon Resource Name (ARN) identifica in modo univoco le risorse AWS. È richiesto un ARN quando è necessario specificare una risorsa in modo inequivocabile in tutto AWS, ad esempio nelle policy IAM, nei tag Amazon Relational Database Service (Amazon RDS) e nelle chiamate API.

#### 4.6.1 Formato ARN

Di seguito sono riportati i formati generali per gli ARN. I formati specifici dipendono dalla risorsa. Per utilizzare un ARN, sostituire il testo **in corsivo** con le informazioni specifiche delle risorse. Tenere presente che gli ARN per alcune risorse omettono la regione, l'ID account o la regione e l'ID account.

```
arn:partition:service:region:account-id:resource-id
arn:partition:service:region:account-id:resource-type/resource-id
arn:partition:service:region:account-id:resource-type:resource-id
```

- **partition**. La partizione in cui si trova la risorsa. Una partizione è un gruppo di regioni AWS. Ogni account AWS ha l'ambito di una partizione.  
Di seguito sono riportate le partizioni supportate:
  - aws – RegioniAWS
  - aws-cn: regioni Cina
  - aws-us-gov – RegioniAWS GovCloud (US)
- **service**. Lo spazio dei nomi del servizio che identifica il prodotto AWS. Ad esempio, s3 per Amazon S3. Per individuare uno spazio dei nomi del servizio, aprire [Service Authorization Reference](#), aprire la pagina per il servizio e individuare la frase "prefisso del servizio" nella prima frase. Ad esempio, nella prima frase della pagina di Amazon S3 viene visualizzato il seguente testo:

```
(service prefix: s3)
```

- **region**. Il codice della regione. Ad esempio, us-east-2 per Stati Uniti orientali (Ohio). Per l'elenco dei codici delle regioni, consultare [Endpoint regionali](#).  
**Nota:** Alcuni servizi, ad esempio IAM, non supportano le regioni. Gli endpoint di questi servizi non includono una regione. Altri servizi, ad esempio Amazon EC2, supportano le regioni ma consentono di specificare un endpoint che non include una regione, ad esempio <https://ec2.amazonaws.com>. Quando si utilizza un endpoint senza regione, AWSindirizza la richiesta Amazon EC2 agli Stati Uniti orientali (Virginia settentrionale) (us-east-1), che è la regione predefinita per le chiamate API.
- **account-id**. L'ID dell'account AWS proprietario della risorsa, senza trattini. Ad esempio, 123456789012.
- **resource-id**. L'identificatore di risorsa. Questa parte dell'ARN può essere il nome o l'ID della risorsa o un percorso di risorsa. Ad esempio, user/Bob per un utente IAM o instance/i-1234567890abcdef0 per un'istanza EC2.

#### 4.6.2 Percorsi negli ARN

Alcuni ARN delle risorse possono includere un percorso. Ad esempio, in Amazon S3, l'identificatore di risorsa è un nome oggetto che può includere barre (/) per creare un percorso. allo stesso modo, anche i nomi utente e i nomi di gruppo IAM possono includere percorsi. I percorsi possono includere un carattere jolly, vale a dire un asterisco (\*). Ad esempio, se si sta scrivendo una policy IAM, è possibile specificare tutti gli utenti IAM che dispongono del percorso product\_1234 utilizzando un carattere jolly come segue:

```
arn:aws:iam::123456789012:user/Development/product_1234/*
```

Analogamente, puoi specificare user/\* per indicare tutti gli utenti o group/\* per indicare tutti i gruppi, come negli esempi seguenti:

```
"Resource": "arn:aws:iam::123456789012:user/*"  
"Resource": "arn:aws:iam::123456789012:group/*"
```

L'esempio seguente mostra gli ARN per un bucket Amazon S3 in cui il nome della risorsa include un percorso:

```
arn:aws:s3:::my_corporate_bucket/*  
arn:aws:s3:::my_corporate_bucket/Development/*
```

## 4.8 Ruoli, [1]

Un ruolo IAM è un'identità IAM che puoi creare nel tuo account e che dispone di autorizzazioni specifiche. Un ruolo IAM è simile a un utente IAM, in quanto è un'identità AWS con policy di autorizzazioni che determinano ciò che l'identità può e non può fare in AWS. Tuttavia, invece di essere associato in modo univoco a una persona, un ruolo è destinato a essere assunto da chiunque. Inoltre, un ruolo non ha credenziali a lungo termine standard associate (password o chiavi di accesso). Tuttavia, quando assumi un ruolo, vengono fornite le credenziali di sicurezza provvisorie per la sessione del ruolo.

Puoi utilizzare i ruoli per delegare l'accesso agli utenti, alle applicazioni o ai servizi che generalmente non hanno accesso alle tue risorse AWS. Ad esempio, è possibile concedere agli utenti nel tuo account AWS l'accesso alle risorse a cui generalmente non è concesso oppure concedere agli utenti in un account AWS l'accesso alle risorse in un altro account. In alternativa, è possibile consentire a un'applicazione mobile l'utilizzo delle risorse AWS, senza tuttavia incorporare le chiavi AWS all'interno dell'app (dove la rotazione può essere difficile e dove gli utenti possono potenzialmente estrarle).

I ruoli possono essere utilizzati da:

- Un utente IAM nello stesso account AWS del ruolo;
- Un utente IAM in un account AWS diverso dal ruolo;
- Un servizio Web offerto da AWS come Amazon Elastic Compute Cloud (Amazon EC2);
- Un utente esterno autenticato da un fornitore di servizi di identità (IdP) compatibile con SAML 2.0 o OpenID Connect o un gestore identità creato appositamente.

### 4.8.1 Scenari comuni, [1]

#### 1- Fornire l'accesso a un utente IAM in un altro account AWS di proprietà dell'utente, [1]

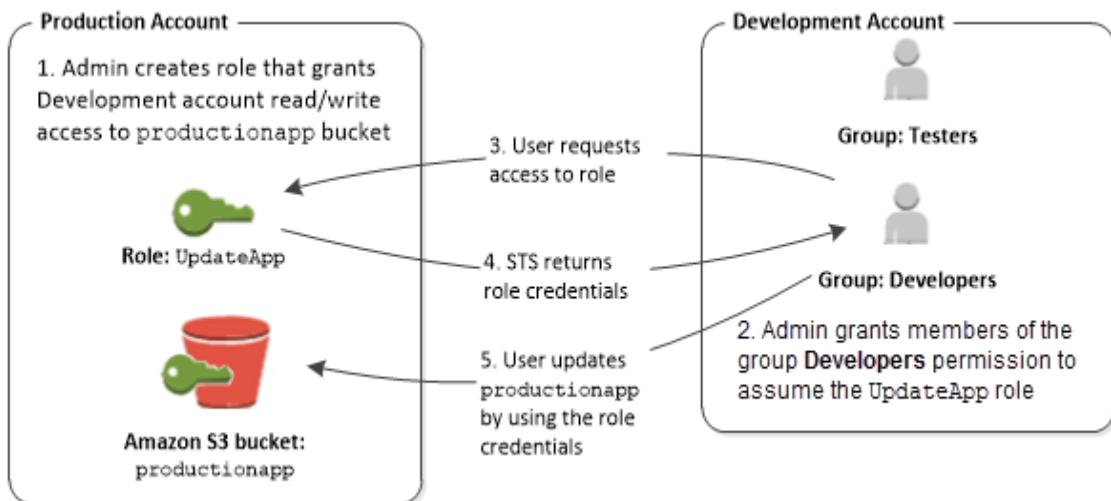
Puoi concedere agli utenti IAM l'autorizzazione per passare da un ruolo all'altro all'interno del tuo account AWS o ai ruoli definiti in altri account AWS di tua proprietà.

Immaginiamo di avere delle istanze Amazon EC2 critiche per la tua organizzazione. Invece di concedere direttamente agli utenti l'autorizzazione a terminare le istanze, è possibile creare un ruolo con tali privilegi. Quindi consentire agli amministratori di passare al ruolo quando è necessario terminare un'istanza. In questo modo si aggiungono i seguenti livelli di protezione alle istanze:

- È necessario concedere esplicitamente agli utenti il permesso di assumere quel ruolo.
- Gli utenti devono passare attivamente al ruolo utilizzando la AWS Management Console o assumerlo tramite la AWS CLI o l'API AWS.
- È possibile aggiungere una Multi-Factor Authentication (MFA) al ruolo, in modo che solo gli utenti che accedono con un dispositivo MFA possano assumere quel ruolo.

Un utente in un account può passare a un ruolo dello stesso o di un altro account. Mentre si usa il ruolo, l'utente è in grado di eseguire solo le azioni e accedere solo alle risorse consentite dal

ruolo; le loro autorizzazioni utente originali sono sospese. Quando l'utente esce dal ruolo, le autorizzazioni utente originali vengono ripristinate.



[Esempio→](#)

## 2- Fornire l'accesso agli account AWS di proprietà di terze parti, [1]

Quando terze parti richiedono l'accesso alle risorse AWS della tua organizzazione, puoi usare i ruoli per delegare l'accesso. Ad esempio, una terza parte potrebbe fornire un servizio per la gestione delle risorse AWS. Con i ruoli IAM, puoi concedere a queste terze parti l'accesso alle tue risorse AWS senza condividere le credenziali di sicurezza AWS. La terza parte può accedere alle risorse AWS assumendo un ruolo creato da te nel tuo account AWS.

Le terze parti devono fornirti le informazioni seguenti per permetterti di creare un ruolo che possa essere da loro assunto:

- ID account AWS della terza parte. Puoi specificare il loro ID account AWS come entità principale quando definisci la policy di affidabilità per il ruolo.
- ID esterno da associare in modo univoco con il ruolo. L'ID esterno può essere qualsiasi identificatore segreto noto a te e alla terza parte. Puoi ad esempio usare un ID di fattura tra te e la terza parte, ma non devi usare qualcosa che sia possibile indovinare, ad esempio il nome o il numero di telefono della terza parte. Devi specificare questo ID quando definisci la policy di affidabilità per il ruolo. La terza parte deve fornire questo ID quando assume il ruolo.
- Autorizzazioni di cui la terza parte necessita per usare le risorse AWS. Devi specificare queste autorizzazioni quando definisci la policy di autorizzazione del ruolo. Questa policy definisce le operazioni consentite e le risorse a cui è possibile accedere.

Dopo aver creato il ruolo, devi fornire l'Amazon Resource Name (ARN) del ruolo alla terza parte. L'ARN del ruolo è necessario per assumere il ruolo.

## 3- Fornire l'accesso a un servizio AWS, [1]

Molti servizi AWS richiedono l'utilizzo di ruoli per controllare ciò a cui può accedere quel servizio. Un ruolo che un servizio assume per eseguire operazioni a tuo nome viene chiamato ruolo del servizio. Quando un ruolo fornisce uno scopo specializzato per un servizio, può essere categorizzato come ruolo del servizio per le istanze EC2 o come ruolo collegato al servizio.

## 4- Fornire l'accesso a utenti autenticati esternamente (federazione delle identità), [1]

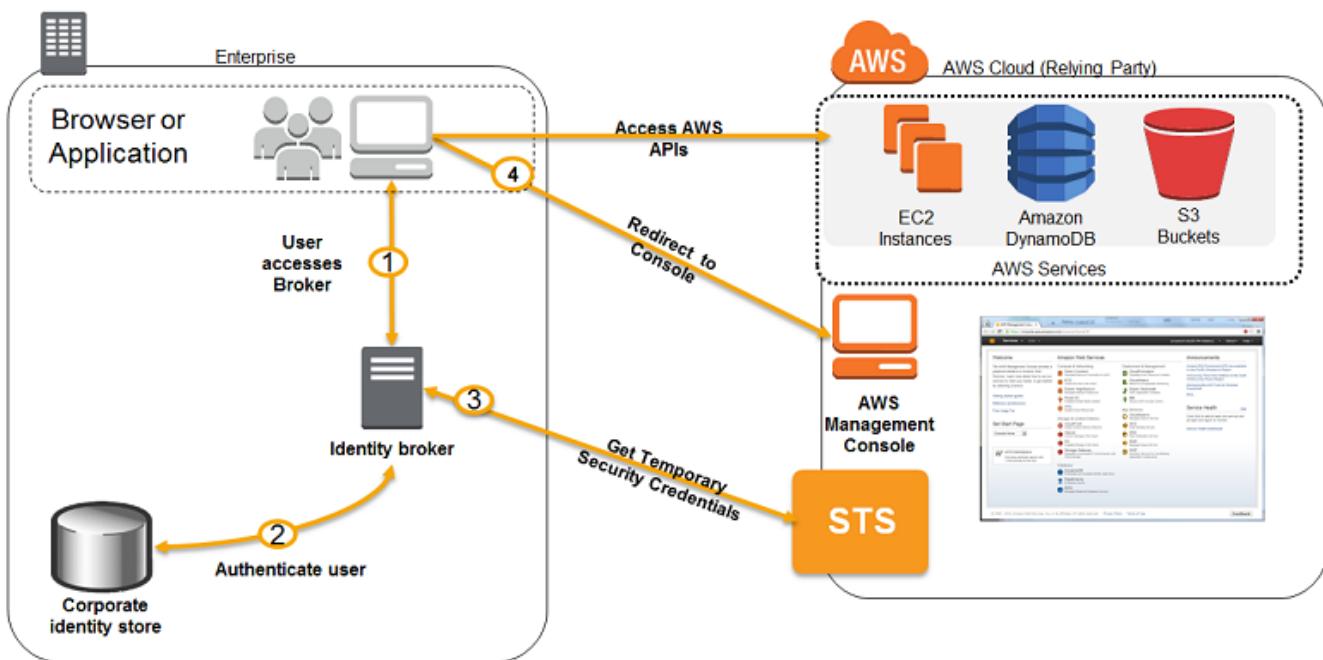
È possibile utilizzare un ruolo IAM per specificare le autorizzazioni per gli utenti la cui identità è federata dalla propria organizzazione o da un provider di identità di terze parti (IdP).

## Esempio:

Ad esempio, Example Corp. ha molti dipendenti che necessitano di eseguire applicazioni interne che accedono alle risorse AWS dell'azienda. I dipendenti hanno già identità nel sistema di identità e autenticazione dell'azienda ed Example Corp. non desidera creare un utente IAM separato per ogni dipendente dell'azienda.

Bob è uno sviluppatore presso Example Corp. Per abilitare le applicazioni interne di Example Corp. in modo che possano accedere alle risorse AWS dell'azienda, Bob sviluppa un'applicazione personalizzata di gestione di identità. L'applicazione verifica che i dipendenti abbiano effettuato l'accesso nel sistema di identità e autenticazione esistente. L'applicazione del gestore identità quindi ottiene le credenziali di sicurezza provvisorie per i dipendenti.

Per ottenere le credenziali di sicurezza provvisorie, l'applicazione del gestore identità chiama `AssumeRole` o `GetFederationToken` per ottenere le credenziali di sicurezza provvisorie, a seconda di come Bob desidera gestire le policy per gli utenti e quando scadono le credenziali provvisorie. La chiamata restituisce le credenziali di sicurezza temporanee, ovvero l'ID chiave di accesso AWS, una chiave di accesso segreta e un token di sessione. L'applicazione del gestore identità rende tali credenziali di sicurezza provvisorie disponibili all'applicazione aziendale interna. L'applicazione può quindi utilizzare le credenziali provvisorie per effettuare chiamate a AWS direttamente. L'app memorizza le credenziali finché non scadono e in seguito richiede un nuovo set di credenziali temporanee. L'immagine seguente illustra questo scenario:



Questo scenario ha quindi i seguenti attributi:

- L'applicazione del gestore identità ha le autorizzazioni per accedere all'API di servizio token IAM (STS) per creare le credenziali di sicurezza temporanee;
- L'applicazione del gestore identità è in grado di verificare che i dipendenti siano autenticati nel sistema di autenticazione esistente;
- Gli utenti sono in grado di ottenere un URL temporaneo che offre loro l'accesso alla Console di gestione AWS (noto come Single Sign-On).

## 4.8.2 Creazione di ruoli IAM, [1]

- Nel riquadro di navigazione della console, selezionare **Roles (Ruoli)** e **Crea ruolo**.

The screenshot shows the AWS IAM Roles page. On the left, there's a navigation sidebar with 'Identity and Access Management (IAM)' selected. Under 'Gestione degli accessi', 'Ruoli' is selected and highlighted with a red circle labeled '1'. At the top right, there are buttons for 'Crea ruolo' (highlighted with a red circle labeled '2'), 'Elimina', and other options. Below the header, there's a search bar and a table with columns for 'Nome ruolo' and 'Entità attendibili'. The table contains several rows of blurred data.

- Nella schermata Seleziona un'entità attendibile selezionare:

- Tipo di entità attendibile**:
- Caso d'uso**. I casi d'uso comuni sono istanze EC2 e Lambda, questo perché i casi più comuni per cui si utilizzano i ruoli sono quelli dove c'è del codice.

Premere quindi **Successivo**.

The screenshot shows the 'Selezione un'entità attendibile' step of the IAM role creation wizard. On the left, there are three phases: 'Fase 1 Selezione un'entità attendibile', 'Fase 2 Aggiungi autorizzazioni', and 'Fase 3 Nomina, verifica e crea'. The main area is titled 'Selezione un'entità attendibile' and contains a section 'Tipo di entità attendibile' with five options. The first option, 'Servizio AWS', is selected and highlighted with a red box labeled '1'. Below it is a section 'Caso d'uso' with two options: 'EC2' (selected) and 'Lambda'. A dropdown menu 'Casi d'uso per altri servizi AWS:' is also shown. At the bottom right are 'Annulla' and 'Successivo' buttons, with 'Successivo' highlighted with a red box labeled '3'.

- Selezionare una o piu' **Policy di autorizzazione**, nell'esempio AmazonS3ReadOnlyAccess, successivamente (facoltativo) **Imposta il limite delle autorizzazioni**, premere quindi **Successivo**.

## Aggiungi autorizzazioni

**Policy di autorizzazione (Selezionati 1/774)**  
Scegli una o più policy da collegare al nuovo ruolo.

**1** **Crea policy**

**2** **Annulla filtri**

**3** **Gestite ...** **Provides read only access to all buckets...**

**4** **Imposta il limite delle autorizzazioni - facoltativo**  
Imposta un limite delle autorizzazioni per controllare il numero massimo di autorizzazioni che può avere questo ruolo. Questa non è un'impostazione comune, ma puoi utilizzarla per delegare la gestione delle autorizzazioni ad altri utenti.

**Crea un ruolo senza un limite delle autorizzazioni**  
 Utilizza un limite delle autorizzazioni per controllare il numero massimo di autorizzazioni del ruolo

**5** **Successivo**

**Annulla** **Precedente** **Successivo**

4. Nella schermata **Nomina, verifica e crea**, inserisci:
- Dettagli del ruolo: **Nome e Descrizione**.

### Nomina, verifica e crea

#### Dettagli del ruolo

##### Nome del ruolo

Inserisci un nome significativo per identificare questo ruolo.

**1**

Massimo 128 caratteri. Utilizza caratteri alfanumerici e i seguenti caratteri speciali: "+=,.@-\_".

##### Descrizione

Aggiungi una breve spiegazione per questa policy.

**2**

Allows EC2 instances to call AWS services on your behalf.

Massimo 1000 caratteri. Utilizza caratteri alfanumerici e i seguenti caratteri speciali: "+=,.@-\_".

- Selezione entità attendibili. Qui è possibile eventualmente modificare il file json.

## Fase 1: seleziona entità attendibili

Modifica

```
1 [ {  
2     "Version": "2012-10-17",  
3     "Statement": [  
4         {  
5             "Effect": "Allow",  
6             "Action": [  
7                 "sts:AssumeRole"  
8             ],  
9             "Principal": {  
10                 "Service": [  
11                     "ec2.amazonaws.com"  
12                 ]  
13             }  
14         }  
15     ]  
16 }
```

- c. Aggiungere eventuali altre autorizzazioni premendo su **Modifica**.

## Fase 2: Aggiungi autorizzazioni

Modifica

Riepilogo della policy delle autorizzazioni		
Nome della policy	Tipo	Collegato come
AmazonS3ReadOnlyAccess	Gestite da AWS	Policy di autorizzazione

- d. (facoltativo) Aggiungere un tag premendo su **Aggiungi tag**. Infine premere **Crea ruolo** per creare il ruolo.

Tag

### Aggiungi tag (Facoltativo)

I tag sono coppie chiave-valore che è possibile aggiungere alle risorse AWS per facilitare l'identificazione, l'organizzazione o la ricerca di risorse.

Nessun tag associato alla risorsa.

The diagram illustrates the process of creating a role. It starts with a red arrow pointing from the 'Aggiungi tag' button (labeled 1) to the 'Crea ruolo' button (labeled 2). A second red arrow points from the 'Crea ruolo' button to the 'Crea ruolo' button, indicating a self-referencing step.

Aggiungi tag

1

2

Puoi aggiungere fino a 50 altri tag

Annulla

Precedente

Crea ruolo

### 4.8.3 Utilizzo di un ruolo IAM per concedere autorizzazioni alle applicazioni in esecuzione su istanze Amazon EC2, [1]

Le applicazioni eseguite su un'istanza EC2 devono includere le credenziali AWS nelle richieste di API AWS. Puoi chiedere agli sviluppatori di salvare le credenziali AWS direttamente nell'istanza EC2, perché possono essere utilizzate dalle applicazioni di tale istanza. Tuttavia, in questo caso, gli sviluppatori dovrebbero gestire le credenziali, accertarsi che vengano passate

in modo sicuro a ciascuna istanza e aggiornare ogni istanza EC2 al momento di ruotare le credenziali. Si tratta di una notevole quantità di lavoro aggiuntivo.

In alternativa, puoi (e devi) utilizzare un ruolo IAM per gestire credenziali temporanee per le applicazioni eseguite in un'istanza EC2. Quando utilizzi un ruolo, non devi necessariamente distribuire credenziali a lungo termine (come, ad esempio, nome utente e password oppure chiavi di accesso) per un'istanza EC2. Al contrario, il ruolo fornisce autorizzazioni provvisorie che possono essere utilizzate dalle applicazioni durante le chiamate ad altre risorse AWS. Quando avvii un'istanza EC2, devi specificare un ruolo IAM da associare ad essa. Le applicazioni eseguite nell'istanza possono quindi utilizzare le credenziali provvisorie fornite dal ruolo per firmare le richieste API.

Questo tipo di utilizzo dei ruoli offre diversi vantaggi. Dato che le credenziali dei ruoli sono provvisorie e vengono ruotate automaticamente, non dovrà preoccuparti della gestione né dei rischi di sicurezza a lungo termine. Inoltre, se utilizzi un singolo ruolo per più istanze, quando apporti una modifica a un ruolo, queste si propagano automaticamente a tutte le istanze.

#### Procedura:

1. Selezionare un'istanza in stato di esecuzione dal menu delle **Istanze**, nella barra in alto selezionare **Stato dell'istanza** → **Sicurezza** → **Modifica il ruolo IAM**.

The screenshot shows the AWS Instances page. A single instance is selected. The 'Sicurezza' (Security) option is highlighted with a red box and a circled number 3. The 'Modifica il ruolo IAM' (Modify IAM role) link is also highlighted with a red box and a circled number 4. Other options like 'Modifica i gruppi di sicurezza' (Modify security groups) and 'Ottieni la password di Windows' (Get Windows password) are visible.

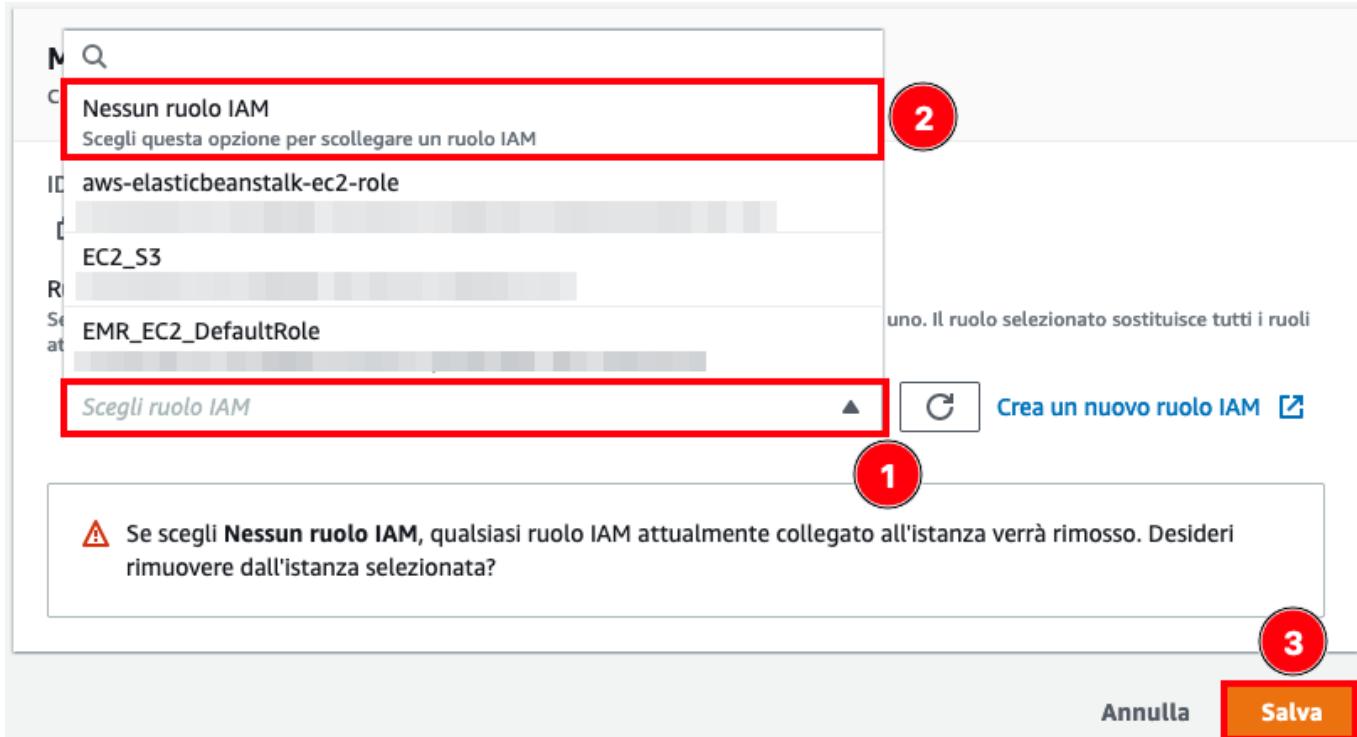
2. Selezionare un ruolo IAM precedentemente creato. Nell'esempio "EC2\_S3".  
Eventualmente è possibile creare un nuovo ruolo IAM dall'apposito link **Crea un nuovo ruolo IAM**. Premere infine **Salva** per attuare le modifiche.

The screenshot shows the 'Modifica il ruolo IAM' (Modify IAM role) dialog. The 'Ruolo IAM' (IAM role) dropdown is set to 'EC2\_S3' and is highlighted with a red box and circled number 1. The 'Crea un nuovo ruolo IAM' (Create new IAM role) link is also present. The 'Salva' (Save) button at the bottom right is highlighted with a red box and circled number 2.

3. Dal terminale dell'istanza EC2 è possibile utilizzare il comando seguente per visualizzare l'elenco dei bucket S3 dell'account AWS. Questo è possibile grazie all'assegnazione del ruolo "EC2\_S3" all'istanza in questione.

```
$ aws s3 ls
```

4. È possibile rimuovere il ruolo dall'istanza recandosi nuovamente nel menù indicato dal punto 1. e dal menù a tendina selezionare **Nessun ruolo IAM**:



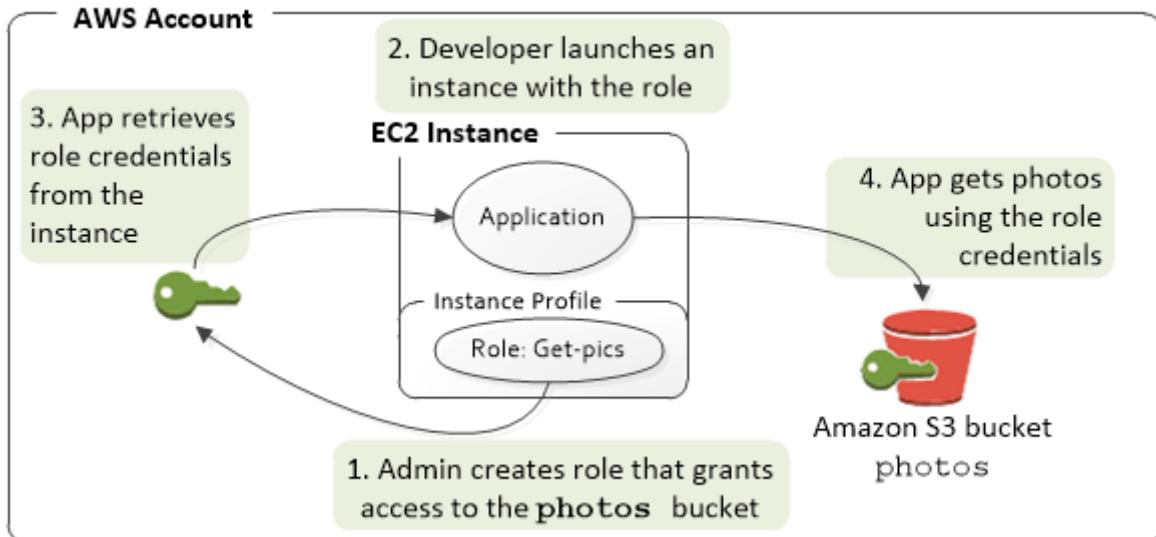
5. Ricordiamo che il ruolo in questione "EC2\_S3" ha la seguente policy `AmazonS3FullAccess`:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "s3:*",
        "s3-object-lambda:*
```

### Funzionamento dei ruoli per le istanze EC2, [1]

Nella figura di seguito, uno sviluppatore esegue un'applicazione su un'istanza EC2 che richiede l'accesso al bucket S3 denominato photos. Un amministratore crea il ruolo del servizio Get-pics e lo collega all'istanza EC2. Il ruolo include una policy di autorizzazione che consente l'accesso

in sola lettura al bucket S3 specificato. Include anche una policy di affidabilità che consente all'istanza EC2 di assumere il ruolo e recuperare le credenziali provvisorie. Quando l'applicazione viene eseguita sull'istanza, può utilizzare le credenziali provvisorie del ruolo per accedere al bucket delle foto. L'amministratore non ha bisogno di concedere allo sviluppatore l'autorizzazione di accedere al bucket delle foto e lo sviluppatore non si trova mai nella necessità di condividere o gestire credenziali.



1. L'amministratore utilizza IAM per creare il ruolo **Get-pics**. Nella policy di affidabilità del ruolo l'amministratore specifica che solo le istanze EC2 possono assumere quel ruolo. Nella policy di autorizzazione del ruolo l'amministratore specifica autorizzazioni di sola lettura per il bucket **photos**.
2. Uno sviluppatore avvia un'istanza EC2 e assegna il ruolo **Get-pics** all'istanza.
3. Quando l'applicazione è in esecuzione, raccoglie le credenziali di sicurezza provvisorie dai metadati dell'istanza Amazon EC2.
4. Grazie all'utilizzo delle credenziali provvisorie recuperate, l'applicazione può accedere al bucket delle foto. In virtù della policy collegata al ruolo **Get-pics** (Ottieni foto), l'applicazione dispone di autorizzazioni di sola lettura. Le credenziali di sicurezza provvisorie disponibili nell'istanza ruotano automaticamente prima della scadenza, in modo da avere un set valido sempre disponibile. L'applicazione deve solo assicurarsi di ottenere un nuovo set di credenziali dai metadati dell'istanza prima della scadenza di quelle esistenti.

## 5. VPC (Amazon Virtual Private Cloud), [1]

Amazon Virtual Private Cloud (Amazon VPC) consente di avviare risorse AWS in una rete virtuale definita. Questa rete virtuale è simile a una comune rete da gestire all'interno del proprio data center, ma con i vantaggi dell'infrastruttura scalabile di AWS.

### Concetti di Amazon VPC

Amazon VPC è il livello di rete per Amazon EC2. Di seguito sono elencati i concetti fondamentali relativi ai VPC:

- Virtual Private Cloud (VPC). Una rete virtuale dedicata all'account AWS.
- Sottorete. Una sottorete è un intervallo di indirizzi IP nel VPC.
- Blocco CIDR: Classless Inter-Domain Routing. Metodologia di route aggregation e allocazione di indirizzi IP.
- Tabella di routing. Contiene un insieme di regole denominate route che consentono di determinare la direzione del traffico di rete.
- Set opzioni DHCP: le informazioni di configurazione (come il nome di dominio e il server del nome di dominio) trasferite alle istanze EC2 quando vengono avviate nelle sottoreti VPC.
- Gateway Internet. Un gateway collegato al VPC per consentire la comunicazione tra le risorse del VPC e Internet.
- Gateway Internet egress-only: un tipo di gateway Internet che consente a un'istanza EC2 in una sottorete di accedere a Internet, ma impedisce alle risorse su Internet di avviare la comunicazione con l'istanza.
- Endpoint VPC: consente di connettere privatamente il VPC a servizi AWS supportati e servizi endpoint VPC powered by PrivateLink senza richiedere un Internet Gateway, un dispositivo NAT, una connessione VPN o una connessione AWS Direct Connect. Le istanze nel VPC non richiedono indirizzi IP pubblici per comunicare con risorse nel servizio.
- Gateway NAT: un servizio AWS gestito che consente alle istanze EC2 di sottoreti private di connettersi a Internet, ad altri VPC o a reti On-Premise.
- Istanza NAT: un'istanza EC2 in una sottorete pubblica che consente alle istanze in sottoreti private di connettersi a Internet, ad altri VPC o a reti On-Premise.
- Gateway Carrier: per sottoreti nelle zone Wavelength, questo tipo di gateway consente il traffico in ingresso da una rete carrier di telecomunicazioni in una posizione specifica e il traffico in uscita verso una rete carrier di telecomunicazioni e verso Internet.
- Elenchi di prefissi: una raccolta di blocchi CIDR che possono essere utilizzati per configurare gruppi di sicurezza VPC, tabelle di routing VPC e tabelle di routing di AWS Transit Gateway e possono essere condivisi con altri account AWS utilizzando Resource Access Manager (RAM).
- Gruppi di sicurezza: fungono da firewall virtuale per controllare il traffico in entrata e quello in uscita per una risorsa AWS, ad esempio un'istanza EC2. Ogni VPC include un gruppo di sicurezza predefinito; è possibile creare gruppi di sicurezza aggiuntivi. Un gruppo di sicurezza può essere utilizzato solo nel VPC per il quale viene creato.
- Liste di controllo degli accessi (ACL) di rete: un livello di sicurezza facoltativo per il VPC che funge da firewall per il controllo del traffico in entrata e in uscita nelle sottoreti.

### 5.1 Come funziona Amazon VPC, [1]

1. **VPC e sottoreti.** Un cloud privato virtuale (Virtual Private Cloud, VPC) è una rete virtuale dedicata al tuo account AWS. Il VPC è isolato a livello logico dalle altre reti virtuali del

cloud AWS. Puoi avviare le risorse AWS, ad esempio le istanze Amazon EC2, nel VPC. Puoi specificare un intervallo di indirizzi IP per il VPC, aggiungere sottoreti, associare gruppi di sicurezza e configurare tabelle di routing.

Una sottorete è un intervallo di indirizzi IP nel VPC. Puoi avviare le risorse AWS in una sottorete specifica. Utilizza una sottorete pubblica per le risorse che devono essere connesse a Internet e una sottorete privata per le risorse da non connettere a Internet.

Per proteggere le risorse AWS in ciascuna sottorete, puoi usare diversi livelli di sicurezza, compresi gruppi di sicurezza e liste di controllo accessi (ACL) alla rete.

Puoi scegliere di associare un blocco CIDR IPv6 al VPC e di assegnare gli indirizzi IPv6 alle istanze nel VPC.

2. **VPC predefiniti e non predefiniti.** Se l'account è stato creato dopo il 04/12/2013, è dotato di un VPC predefinito che ha una sottorete predefinita in ciascuna zona di disponibilità. Un VPC predefinito presenta i vantaggi delle caratteristiche avanzate offerte da EC2-VPC ed è subito pronto all'uso. Se disponi di un VPC predefinito e non specifichi una sottorete all'avvio dell'istanza, questa verrà avviata nel VPC predefinito. Puoi avviare un'istanza nel VPC predefinito senza alcuna conoscenza di Amazon VPC.

Puoi inoltre creare il tuo VPC e configuralo in base alle esigenze. Questo è il cosiddetto VPC non predefinito. Le sottoreti create nel VPC non predefinito e le altre sottoreti create nel VPC predefinito vengono chiamate sottoreti non predefinite.

3. **Assegnazione di indirizzi IP.** Gli indirizzi IP permettono alle risorse nel VPC di comunicare tra loro e con le risorse su Internet.

Quando si crea un VPC, bisogna assegnargli un blocco CIDR IPv4 (un intervallo di indirizzi IPv4 privati), un blocco CIDR IPv6 o entrambi (dual-stack). Gli indirizzi IPv4 privati non sono raggiungibili tramite Internet. Gli indirizzi IPv6 sono univoci a livello globale, e possono essere configurati per rimanere privati o essere raggiungibili via Internet.

Il VPC può operare in modalità dual-stack. Ciò significa che le risorse possono comunicare via IPv4, IPv6 o entrambi IPv4 e IPv6. Gli indirizzi IPv4 e IPv6 sono indipendenti l'uno dall'altro; devi aggiungere route e gruppi di sicurezza separati per IPv4 e IPv6.

4. **Indirizzi IPv4 privati.** Gli indirizzi IPv4 privati (in questo argomento chiamati anche indirizzi IP privati) non sono raggiungibili tramite Internet e si possono utilizzare per la comunicazione tra le istanze presenti nel VPC. Quando avvii un'istanza in un VPC, all'interfaccia di rete predefinita (eth0) dell'istanza viene assegnato un indirizzo IP privato primario dall'intervallo di indirizzi IPv4 della sottorete. Ciascuna istanza riceve anche un nome host DNS privato (interno) che si risolve nell'indirizzo IP privato dell'istanza. Il nome host può essere di due tipi: basato su risorse o basato su IP.

Se non specifichi un indirizzo IP privato primario, sarà AWS a selezionare per conto tuo un indirizzo IP disponibile nell'intervallo della sottorete.

5. **Indirizzi IPv4 pubblici.** Tutte le sottoreti hanno un attributo che determina se un'interfaccia di rete creata nella sottorete riceve automaticamente un indirizzo IPv4 pubblico (in questo argomento denominato indirizzo IP pubblico). Di conseguenza, quando avvii un'istanza in una sottorete dotata di questo attributo, all'interfaccia di rete primaria (eth0) creata per l'istanza viene assegnato un indirizzo IP pubblico. Un indirizzo IP pubblico è associato all'indirizzo IP privato primario tramite conversione degli indirizzi di rete (network address translation, NAT).

Puoi controllare se la tua istanza riceve un indirizzo IP pubblico eseguendo da CLI:

```
$ curl ipecho.net/plain; echo
```

L'indirizzo IP pubblico viene assegnato alla tua istanza dal pool di indirizzi IP pubblici di Amazon; non è associato al tuo account. Quando un indirizzo IP pubblico viene disassociato dalla tua istanza, viene reinserito nel pool di indirizzi e non potrai più utilizzarlo. Non puoi associare o disassociare manualmente un indirizzo IP pubblico.

Se ti occorre un indirizzo IP pubblico persistente allocato sul tuo account che puoi assegnare o rimuovere dalle istanza in base alle tue esigenze, è preferibile utilizzare un indirizzo IP elastico.

6. **Tabelle di routing.** Una tabella di routing contiene un insieme di regole, denominate route, che consentono di determinare la direzione del traffico di rete proveniente dal VPC. Puoi associare esplicitamente una sottorete a una particolare tabella di routing. In caso contrario, la sottorete è implicitamente associata alla tabella di routing principale. Ogni route in una tabella di routing specifica l'intervallo di indirizzi IP in cui si desidera instradare il traffico (la destinazione) e il gateway, l'interfaccia di rete o la connessione attraverso cui inviare il traffico (il target).
7. **Accesso a Internet.** Puoi controllare il modo in cui le istanze che avvii in un VPC accedono alle risorse Esterne al VPC. Un VPC predefinito include un Internet gateway e ogni sottorete predefinita è una sottorete pubblica. Ciascuna istanza avviata in una sottorete predefinita ha un indirizzo IPv4 privato e uno pubblico. Queste istanze possono comunicare con Internet tramite l'Internet gateway. Un Internet gateway permette alle istanze di connettersi a Internet tramite l'edge della rete Amazon EC2.

Per impostazione predefinita, tutte le istanze avviate in una sottorete non predefinita hanno un indirizzo IPv4 privato ma non hanno indirizzi IPv4 pubblici, a meno che tu non gliene assegni uno in fase di avvio o non modifichi l'attributo dell'indirizzo IP pubblico. Queste istanze possono comunicare tra di loro, ma non possono accedere a Internet.

Puoi abilitare l'accesso Internet di un'istanza avviata in una sottorete non predefinita collegando un Internet gateway al VPC (se il VPC non è predefinito) e associando all'istanza un indirizzo IP elastico.

In alternativa, per consentire a un'istanza nel VPC di avviare connessioni in uscita a Internet ma impedire connessioni in entrata indesiderate da Internet, puoi utilizzare un dispositivo di network address translation (NAT). NAT associa più indirizzi IPv4 privati a un solo indirizzo IPv4 pubblico. Puoi configurare il dispositivo NAT con un indirizzo IP elastico e connetterlo a Internet tramite un gateway Internet. Ciò consente a un'istanza di una sottorete privata di connettersi a Internet tramite il dispositivo NAT, che instrada il traffico dall'istanza al gateway Internet e le risposte all'istanza.

8. **Accesso a una rete domestica o aziendale.** Facoltativamente, puoi connettere il VPC al tuo data center aziendale utilizzando una connessione AWS Site-to-Site VPN IPsec, che rende AWS Cloud un'estensione del data center.  
Una connessione Site-to-Site VPN è costituita da due tunnel VPN tra un dispositivo gateway virtuale privato o gateway di transito sul lato AWS e un dispositivo gateway del cliente situato nel data center. Un dispositivo gateway del cliente è un dispositivo fisico o un'appliance software che puoi configurare sul tuo lato della connessione Site-to-Site VPN.

## 5.2 Creazione di un VPC, [1]

**AWS Cloud Map**  
Crea una mappa dinamica del cloud

**CloudFront**  
Rete globale di distribuzione dei contenuti

**Direct Connect**  
Connessione di rete dedicata ad AWS

**Global Accelerator**  
Migliora la disponibilità e le prestazioni della tua applicazione utilizzando la rete globale AWS

**Route 53**  
DNS scalabile e registrazione dei nomi dei domini

**VPC**  
Risorse cloud isolate

**Amazon VPC IP Address Manager**  
Servizio di gestione degli indirizzi IP gestiti

### Creare solo un VPC, [1]:

1. Nel riquadro di navigazione, selezionare Your VPCs (I tuoi VPC), Create VPC (Crea VPC).

	Name	ID VPC	Stato
<input type="checkbox"/>	-	vpc-bd9106d9	Available
<input type="checkbox"/>	[REDACTED]	[REDACTED]	Available

2. Sotto Resources to Create (Risorse da creare), scegliere VPC only (Solo VPC).

3. Specificare i seguenti dettagli VPC in base alle esigenze.

- a. **Name tag (Tag nome)**: fornire facoltativamente un nome per il VPC. In questo modo viene creato un tag con una chiave di Name e il valore specificato.
- b. **IPv4 CIDR block (Blocco CIDR IPv4)**: specificare un blocco CIDR IPv4 (o un intervallo di indirizzi IP) per il VPC. Seleziona una delle seguenti opzioni:
  - i. **Input manuale CIDR IPv4**: immettere manualmente un CIDR IPv4. La dimensione del blocco CIDR deve essere compresa tra /16 e /28. Ti

- consigliamo di specificare un blocco CIDR dagli intervalli di indirizzi IP privati (non instradabili pubblicamente) come specificato in [RFC 1918](#); ad esempio, 10.0.0.0/16 o 192.168.0.0/16.
- ii. **IPAM-allocated IPv4 CIDR block (Blocco CIDR IPv4 assegnato da IPAM)**: se in questa regione è disponibile un pool di indirizzi IPv4 di Amazon VPC IP Address Manager (IPAM), è possibile ottenere un CIDR da un pool IPAM. Se si seleziona un pool IPAM, la dimensione del CIDR è limitata dalle regole di allocazione sul pool IPAM (minimo consentito, massimo consentito e di default).
  - c. **Blocco CIDR IPv6**: Facoltativamente, puoi associare un blocco CIDR IPv6 al VPC. Scegliere una delle seguenti opzioni, quindi scegliere Selezione CIDR:
    - i. **No IPv6 CIDR block (Nessun blocco CIDR IPv6)**: nessun CIDR IPv6 verrà sottoposto a provisioning per questo VPC.
    - ii. **IPAM-allocated IPv6 CIDR block (Blocco CIDR IPv6 assegnato da IPAM)**: se in questa regione è disponibile un pool di indirizzi IPv6 di Amazon VPC IP Address Manager (IPAM), è possibile ottenere un CIDR da un pool IPAM. Se si seleziona un pool IPAM, la dimensione del CIDR è limitata dalle regole di allocazione sul pool IPAM (minimo consentito, massimo consentito e di default).
    - iii. **Amazon-provided IPv6 CIDR block (Blocco CIDR IPv6 fornito da Amazon)**: richiede un blocco CIDR IPv6 da un pool di indirizzi IPv6 di Amazon. Per **Gruppo di confine di rete**, selezionare il gruppo da cui AWS pubblica gli indirizzi IP. Amazon fornisce una dimensione fissa del blocco CIDR IPv6 di /56. Non è possibile configurare le dimensioni del CIDR IPv6 fornito da Amazon
    - iv. **IPv6 CIDR owned by me (CIDR IPv6 di mia proprietà)**: ([BYOIP](#)) Alloca un blocco CIDR IPv6 dal pool di indirizzi IPv6. Per **Pool**, scegliere il pool di indirizzi IPv6 da cui allocare il blocco CIDR IPv6.
  - d. **Tenancy (Locazione)**: scegliere l'opzione di tenancy per questo VPC.
    - i. Selezionare **Default** per garantire che le istanze EC2 avviate in questo VPC utilizzino l'attributo della locazione delle istanze EC2 specificato all'avvio dell'istanza EC2.
    - ii. Selezionare **Dedicated (Dedicato)** per garantire che le istanze EC2 avviate in questo VPC siano eseguite su istanze dedicate a tenant singolo indipendentemente dall'attributo di locazione specificato all'avvio.

## Impostazioni VPC

Risorse da creare [Informazioni](#)

Crea solo la risorsa VPC o crea VPC, sottoreti, ecc.

Solo VPC

1

VPC, sottoreti, ecc.

Tag del nome - *facoltativo*

Crea un tag con una chiave del 'nome' e un valore specificato.

DS2-esercizio

Blocco CIDR IPv4 [Informazioni](#)

Input manuale CIDR IPv4

2

Blocco CIDR IPv4 allocato IPAM

CIDR IPv4

10.2.0.0/16

3

4

Blocco CIDR IPv6 [Informazioni](#)

Nessun blocco CIDR IPv6

5

Blocco CIDR IPv6 allocato IPAM

Blocco CIDR IPv6 fornito da Amazon

CIDR IPv6 di mia proprietà

Tenancy [Informazioni](#)

di default

6

- e. **Tags (Tag):** aggiungi tag facoltativi sul VPC. Un tag è un'etichetta che assegna a una risorsa AWS. Ciascun tag è formato da una chiave e da un valore opzionale. Puoi utilizzare i tag per cercare e filtrare le risorse o monitorare i costi AWS.

### 4. Seleziona **Create VPC (Crea VPC)**.

#### Tag

Un tag è un'etichetta assegnata a una risorsa AWS. Ogni tag è composto da una chiave e da un valore facoltativo. È possibile usare i tag per ricercare e filtrare le risorse o monitorare i tuoi costi di AWS.

Chiave

Valore - *facoltativo*

Name

X

DS2-esercizio

X

Rimuovi

1

**Aggiungi nuovo tag**

È possibile aggiungere 49 altri tag.

2

Annulla

**Crea VPC**

**Creare un VPC, sottoreti e altre risorse VPC, [1]:**

- Nel riquadro di navigazione, selezionare **Your VPCs (I tuoi VPC)**, **Create VPC (Crea VPC)**.

The screenshot shows the AWS VPC console interface. On the left, there's a sidebar with a 'New VPC Experience' toggle and several navigation options: Sottoreti (highlighted with a red box and circled '1'), Tabelle di routing, Gateway Internet, Gateway Internet Egress Only, Set opzioni DHCP, and IP elastiche. The main content area is titled 'I tuoi VPC (2) Informazioni'. It features a search bar 'Filtro VPC' and a table with columns: Name, ID VPC, and Stato. Two rows are listed: one with '-' and 'Available' status, and another with a blurred name and 'Available' status. At the top right of the main area is a 'C' button, an 'Operazioni' dropdown, and a prominent orange 'Crea VPC' button circled '2'.

- Sotto **Resources to Create (Risorse da creare)**, scegli **VPC, subnets, etc. (Sottoreti, VPC e altro)**.

- Modifica le opzioni secondo necessità:

- Name tag auto-generation (Generazione automatica del tag nome)**: scegli un tag nome che verrà applicato alle risorse create. Il tag può essere generato automaticamente oppure puoi definire il valore. Il valore definito verrà utilizzato per generare il tag nome in tutte le risorse come "name-resource". Ad esempio, se inserisci "Preproduction", ogni sottorete verrà contrassegnata con un tag nome "Preproduction-subnet". Un tag è un'etichetta che assegna a una risorsa AWS. Ciascun tag è formato da una chiave e da un valore opzionale. Puoi utilizzare i tag per cercare e filtrare le risorse o monitorare i costi AWS.
- IPv4 CIDR block (Blocco CIDR IPv4)**: scegli un CIDR IPv4 per il VPC. Questa opzione è obbligatoria.
- IPv6 CIDR block (Blocco CIDR IPv6)**: lascia **No IPv6 CIDR block (Nessun blocco CIDR IPv6)** selezionato per questo esercizio.
- Tenancy**: scegli **Default** per questo esercizio per garantire che le istanze EC2 avviate in questo VPC utilizzino l'attributo della locazione delle istanze EC2 specificato all'avvio dell'istanza EC2.
- Availability Zones (AZs) (Zone di disponibilità (AZs))**: scegli 1 per questo esercizio. Una AZ consiste in uno o più data center separati con alimentazione, rete e connettività ridondanti in una regione AWS. Le AZ consentono di gestire applicazioni e database di produzione con maggiore disponibilità, tolleranza ai guasti e scalabilità rispetto a un singolo data center. Se partizionerai le applicazioni in esecuzione nelle sottoreti tra le AZ, sarai meglio isolato e protetto da problemi come interruzioni di corrente, fulmini, tornado, terremoti e altro ancora.
- Customize AZs (Personalizza AZ)**: lascia selezionate le opzioni predefinite per questo esercizio.
- Number of public subnets (Numero di sottoreti pubbliche)**: scegli 1 per questo esercizio. Una sottorete "pubblica" è una sottorete associata a una tabella di routing che punta a un gateway Internet. Ciò consente alle istanze EC2 in esecuzione nella sottorete di essere accessibili pubblicamente su Internet.
- Customize public subnets CIDR blocks (Personalizza blocchi CIDR delle sottoreti pubbliche)**: lascia selezionate le opzioni predefinite per questo esercizio.

- i. **Number of private subnets (Numero di sottoreti private)**: scegli 1 per questo esercizio. Una sottorete “privata” è una sottorete che non dispone di una tabella di routing che punta a un Internet Gateway. Usa sottoreti private per proteggere le risorse back-end che non devono essere accessibili pubblicamente su Internet.
  - j. **Customize private subnets CIDR blocks (Personalizza blocchi CIDR delle sottoreti private)**: lascia selezionate le opzioni predefinite per questo esercizio.
  - k. **NAT gateways (Gateway NAT)**: scegli **None (Nessuna)** per questo esercizio. Un gateway NAT è un servizio gestito da AWS che consente alle istanze EC2 nelle sottoreti private di inviare traffico in uscita a Internet. Tuttavia, le risorse su Internet non possono stabilire una connessione con le istanze. Tieni presente che esiste un costo associato ai gateway NAT.
  - l. **VPC endpoints (Endpoint VPC)**: scegli **None (Nessuno)** per questo esercizio. Un endpoint VPC consente di connettere privatamente il VPC ai servizi AWS supportati come Amazon S3. Gli endpoint VPC consentono di creare un VPC isolato chiuso da Internet pubblico. L'utilizzo di endpoint gateway non comporta costi supplementari. Ciò può aiutare a evitare i costi associati ai gateway NAT.
  - m. **DNS options (Opzioni DNS)**: seleziona entrambe le opzioni di risoluzione dei nomi di dominio per le istanze EC2 avviate in questo VPC.
    - i. **Enable DNS hostnames (Abilita nomi host DNS)**: consente di eseguire il provisioning dei nomi host per gli indirizzi IPv4 pubblici dell'istanza EC2.
    - ii. **Enable DNS resolution (Abilita risoluzione DNS)**: consente ai nomi host di eseguire il provisioning degli indirizzi IPv4 pubblici dell'istanza EC2 e di abilitare la risoluzione dei nomi di dominio dei nomi host.
4. Nel riquadro **Preview (Anteprima)**, è possibile visualizzare il VPC, le sottoreti, le tabelle di routing e le connessioni di rete che verranno create. Una connessione di rete -igw rappresenta un gateway Internet che verrà creato. Una voce di routing che punta al gateway Internet verrà inoltre aggiunta alla tabella di routing associata alla sottorete pubblica.
5. Seleziona **Create VPC (Crea VPC)**.

### 5.3 Sottoreti per il VPC, [1]

Una sottorete è un intervallo di indirizzi IP nel VPC. Puoi avviare le risorse AWS in una sottorete specifica. Utilizza una sottorete pubblica per le risorse che devono essere connesse a Internet e una sottorete privata per le risorse da non connettere a Internet.

Per proteggere le risorse AWS in ciascuna sottorete, puoi usare diversi livelli di sicurezza, compresi gruppi di sicurezza e liste di controllo accessi (ACL) alla rete.

Una sottorete è un intervallo di indirizzi IP nel VPC. È possibile avviare le risorse AWS, ad esempio le istanze EC2, in una sottorete specifica. Quando si crea una sottorete, è necessario specificare il blocco CIDR IPv4 per la sottorete, che è un sottoinsieme del blocco CIDR del VPC. Ogni sottorete deve risiedere totalmente all'interno di una zona di disponibilità e non può estendersi in altre zone. Avviando le istanze in zone di disponibilità separate, sarà possibile proteggere le applicazioni dai fallimenti di una singola posizione.

A seconda della configurazione del VPC, le sottoreti possono essere considerate pubbliche, private o solo per VPN:

- Sottorete pubblica: il traffico IPv4 o IPv6 della sottorete viene instradato a un gateway Internet o a un gateway Internet egress-only e può raggiungere la rete Internet pubblica.
- Sottorete privata: il traffico IPv4 o IPv6 della sottorete non viene instradato a un gateway Internet o a un gateway Internet egress-only e non può raggiungere la rete internet pubblica.
- Sottorete solo per VPN: la sottorete non dispone di una route al gateway internet, ma il suo traffico viene instradato a un gateway virtuale privato per una connessione Site-to-Site VPN.

A prescindere dal tipo di sottorete, l'intervallo di indirizzi IPv4 interni della sottorete è sempre privato; il blocco dell'indirizzo non viene annunciato su Internet.

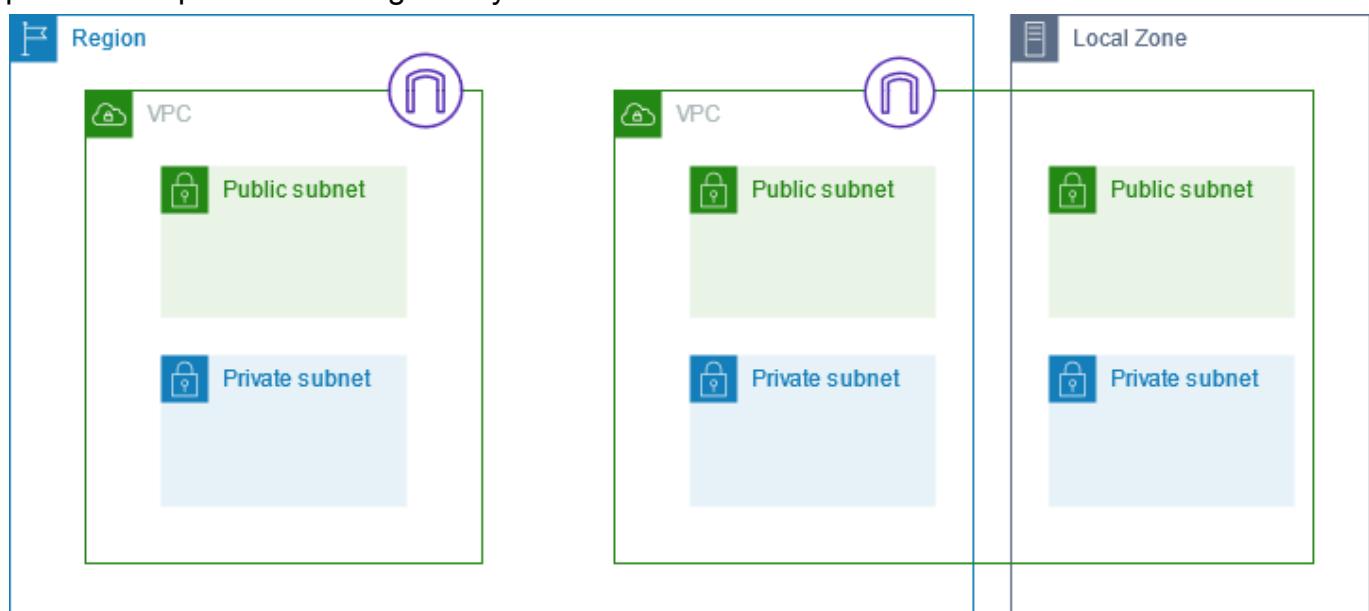
### Impostazioni sottorete:

Tutte le sottoreti hanno un attributo modificabile che determina se all'interfaccia di rete creata nella sottorete viene assegnato un indirizzo IPv4 pubblico e, se possibile, un indirizzo IPv6. Questo include l'interfaccia di rete primaria (eth0) creata per l'istanza quando questa viene avviata nella sottorete. Indipendentemente dall'attributo della sottorete, puoi comunque sostituire questa impostazione per un'istanza specifica durante il suo avvio.

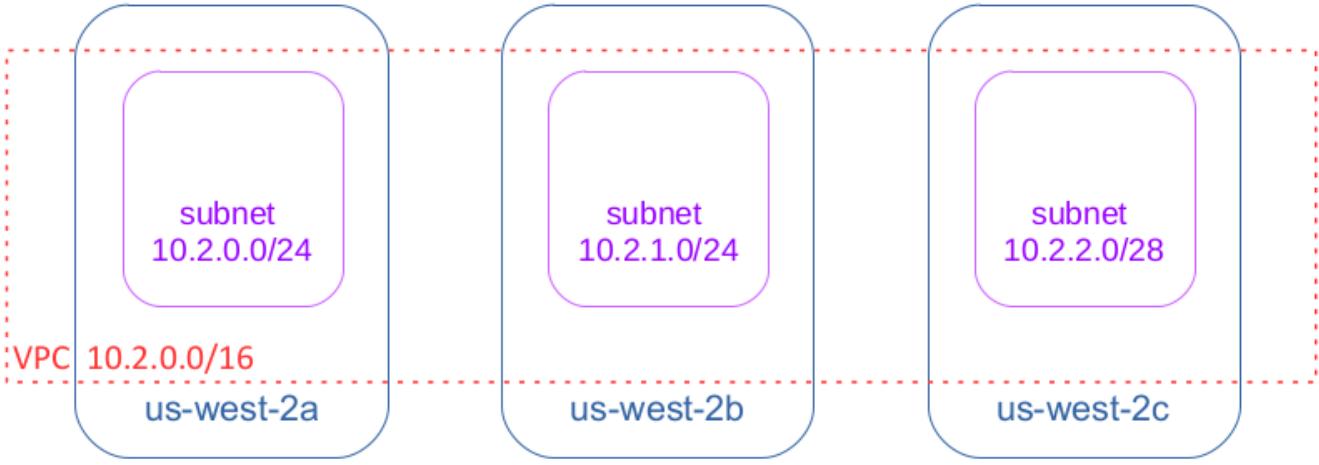
Una volta creata una sottorete, è possibile modificare le seguenti impostazioni per la sottorete.

- Impostazioni di assegnazione automatica IP: consente di configurare le impostazioni di assegnazione automatica IP per richiedere automaticamente un indirizzo IPv4 o IPv6 pubblico per una nuova interfaccia di rete in questa sottorete.
- Impostazioni RBN (Resource-based Name): consente di specificare il tipo di nome host per le istanze EC2 in questa sottorete e di configurare il modo in cui vengono gestite le query dei registri DNS A e AAAA.

Il seguente diagramma mostra due VPC in una regione. Ogni VPC dispone di sottoreti pubbliche e private e di un gateway Internet. Il VPC si estende anche alla zona locale.



Altro esempio:



### 5.3.1 Dimensionamento sottorete, [1]

Il blocco CIDR di una sottorete può essere identico al blocco CIDR per il VPC (per una sottorete singola nel VPC) o una sottorete del blocco CIDR per il VPC (per creare più sottoreti nel VPC). Le dimensioni di blocco consentite devono essere comprese tra una netmask /28 e una netmask /16. Se crei più di una sottorete in un VPC, i blocchi CIDR delle sottoreti non possono sovrapporsi.

Ad esempio, se crei un VPC con blocco CIDR **10.0.0.0/24**, supporta 256 indirizzi IP. Puoi suddividere questo blocco CIDR in due sottoreti, ciascuna delle quali supporta 128 indirizzi IP. Una sottorete utilizza il blocco CIDR **10.0.0.0/25** (per indirizzi 10.0.0.0 - 10.0.0.127) e l'altra utilizza il blocco CIDR **10.0.0.128/25** (per indirizzi 10.0.0.128 - 10.0.0.255).

Ci sono strumenti disponibili su Internet per aiutarti a calcolare e creare blocchi CIDR di sottoreti IPv4, come [\[QUESTO\]](#).

I primi quattro indirizzi IP e l'ultimo indirizzo IP in ogni blocco CIDR della sottorete non sono disponibili per l'utilizzo e non possono essere assegnati a una risorsa, ad esempio a un'istanza EC2. Ad esempio, in una sottorete con blocco CIDR **10.0.0.0/24**, i cinque indirizzi IP seguenti sono riservati:

- 10.0.0.0: indirizzo di rete;
- 10.0.0.1: riservato da AWS per il router VPC;
- 10.0.0.2: riservato da AWS. L'indirizzo IP del server DNS è la base dell'intervallo di rete VPC più due. Per VPC con più blocchi CIDR, l'indirizzo IP del server DNS si trova nel CIDR principale. Ci riserviamo anche la base di ogni intervallo di sottorete più due per tutti i blocchi CIDR nel VPC. Per ulteriori informazioni, consulta Server DNS Amazon;
- 10.0.0.3: riservato da AWS per uso futuro;
- 10.0.0.255: indirizzo di trasmissione di rete. Non supportiamo la trasmissione in un VPC, pertanto riserviamo questo indirizzo.

### Routing della sottorete:

Ogni sottorete deve essere associata a una tabella di routing, che specifica le route consentite per il traffico in uscita che lascia la sottorete. Ogni sottorete creata viene automaticamente associata alla tabella di routing principale per il VPC. Puoi modificare l'associazione E modificare il contenuto della tabella di routing principale.

### 5.3.2 Creazione di una sottorete nel VPC, [1]

Per aggiungere una nuova sottorete al VPC, devi specificare un blocco CIDR IPv4 per la sottorete dall'intervallo del VPC. Puoi specificare la zona di disponibilità in cui si deve trovare la

sottorete. La stessa zona di disponibilità può contenere più sottoreti. Per aggiungere una sottorete al VPC:

1. Accedere alla console Amazon VPC all'indirizzo: <https://console.aws.amazon.com/vpc/>
2. Nel riquadro di navigazione, scegliere **Subnets (Sottoreti)**.
3. Scegliere **Create subnet (Crea sottorete)**.

The screenshot shows the AWS VPC Subnets page. On the left, there's a sidebar with options like 'Nuova esperienza VPC', 'Pannello di controllo VPC', 'Visualizzazione globale di EC2', 'Filtro per VPC', 'Cloud privato virtuale', 'I tuoi VPC', and 'Sottoreti' (which is highlighted with a red box and circled with a red number 1). The main area is titled 'Sottoreti (5)'. It has a search bar 'Filtra sottoreti', a 'Crea sottorete' button (circled with a red box), and a table with columns: Name, ID sottorete, Stato, and VPC. The table lists five subnets: 'subnet-b85211d0', 'subnet-466e0e3c', 'subnet-8e8109f4', 'subnet-f8f9c3b2', and 'subnet-31391c59', all marked as 'Available' in the VPC 'vpc-4c1b9a24'.

4. Per **VPC ID (ID VPC)**: scegliere il VPC per la sottorete.

This screenshot shows the first step of the 'Create subnet' wizard. It's titled 'VPC'. There's a dropdown menu labeled 'ID VPC' containing 'vpc-4c1b9a24', which is highlighted with a red box and circled with a red number 2. Below it, there's a section for 'CIDR VPC associati' with 'CIDR IPv4' set to '172.31.0.0/16'.

5. (Opzionale) Per **Subnet name (Nome sottorete)** inserisci un nome per la sottorete. In questo modo viene creato un tag con una chiave di Name e il valore specificato.
6. Per **Availability zone (Zona di disponibilità)**, puoi scegliere una zona per la sottorete oppure lasciare il valore predefinito **No Preference (Nessuna preferenza)** per consentire a AWS di sceglierne uno per conto tuo.
7. Se la sottorete deve essere una sottorete solo IPv6, scegliere **IPv6-only (Solo IPv6)**. Questa opzione è disponibile solo se il VPC dispone di un blocco CIDR IPv6 associato. Se si sceglie questa opzione, non è possibile associare un blocco CIDR IPv4 alla sottorete.
8. Per **IPv4 CIDR block (Blocco CIDR IPv4)**: inserire un blocco CIDR IPv4 per la sottorete. Ad esempio, **10.0.1.0/24**. Se è stato scelto Solo IPv6, questa opzione non è disponibile.
9. Per **IPv6 CIDR block (Blocco CIDR IPv6)**, scegli **Custom IPv6 CIDR (CIDR IPv6 personalizzato)** e specificare il valore della coppia esadecimale (ad esempio **00**). Questa opzione è disponibile solo se il VPC dispone di un blocco CIDR IPv6 associato.
10. Scegliere **Create subnet (Crea sottorete)**.

**Impostazioni sottorete**  
Specifica i blocchi CIDR e la zona di disponibilità per la sottorete.

**Sottorete 1 di 1**

**Nome sottorete**  
Crea un tag con una chiave del 'nome' e un valore specificato.  
**my-subnet-01**

Il nome può contenere fino a 256 caratteri.

**Zona di disponibilità** [Informazioni](#)  
Scegli la zona in cui risiederà la sottorete o lascia che Amazon ne scelga una per te.  
**Nessuna preferenza**

**Blocco CIDR IPv4** [Informazioni](#)  
**10.0.0.0/24**

**▼ Tag - facoltativo**  
Nessun tag associato alla risorsa.

**Aggiungi nuovo tag** **4**  
È possibile aggiungere 50 altri tag.

**Rimuovi**

**Aggiungi nuova sottorete** **5**

**Annulla** **Crea sottorete**

### Fasi successive:

Dopo aver creato una sottorete, è possibile configurarla come segue:

- Configurare il routing. È quindi possibile creare una tabella di routing personalizzata e una route per inviare il traffico a un gateway associato al VPC, ad esempio un gateway Internet.
- Modificare il comportamento di assegnazione di indirizzi IP. È possibile specificare se le istanze avviate in tale sottorete ricevono un indirizzo IPv4 pubblico, un indirizzo IPv6 o entrambi.
- Modifica le impostazioni del nome basato sulle risorse (RBN).
- Creare o modificare le liste di controllo accessi di rete.
- Condividere la sottorete con altri account.

### 5.3.3 Eliminare una sottorete, [1]

Se una sottorete non è più necessaria, è possibile eliminarla. Non è possibile eliminare una sottorete se contiene interfacce di rete. Ad esempio, è necessario terminare tutte le istanze in una sottorete prima di poterla eliminare. Per eliminare una sottorete:

1. Apri la console Amazon EC2 all'indirizzo <https://console.aws.amazon.com/ec2/>
2. **Terminare** tutte le istanze nella sottorete.

The screenshot shows the AWS EC2 Instances page. A red box labeled 1 highlights the 'Istanze' link in the navigation bar. A red box labeled 2 highlights the checkbox for selecting instances. A red box labeled 3 highlights the 'Termina Istanza' (Terminate) button. A red box labeled 4 highlights the 'Termina Istanza' button in the context menu.

3. Accedere alla console Amazon VPC all'indirizzo <https://console.aws.amazon.com/vpc/>
4. Nel riquadro di navigazione, scegliere **Subnets (Sottoreti)**.
5. Selezionare la sottorete e scegliere **Actions (Operazioni), Delete Subnet (Elimina sottorete)**.

The screenshot shows the AWS VPC Subnets page. A red box labeled 1 highlights the 'Sottoreti' link in the navigation bar. A red box labeled 2 highlights the checkboxes for selecting subnets. A red box labeled 3 highlights the 'Operazioni' (Actions) button. A red box labeled 4 highlights the 'Elimina sottorete' (Delete subnet) button.

6. Quando viene richiesta la conferma, digitare **elimina** e quindi scegliere **Delete (Elimina)**.

#### 5.4 ACL (Liste di controllo accessi di rete), [1]

Una lista di controllo accessi di rete è un livello di sicurezza opzionale per il VPC che agisce come un firewall per controllare il traffico in entrata e in uscita da una o più sottoreti. Si possono impostare liste di controllo accessi di rete con regole simili a quelle del gruppo di sicurezza, in modo tale da aggiungere un ulteriore livello di sicurezza al VPC.

Di seguito sono riportate le nozioni di base che occorre sapere sulle liste di controllo accessi di rete:

- Il VPC viene fornito automaticamente con una lista di controllo accessi di rete modificabile. Per impostazione predefinita, consente tutto il traffico IPv4 in entrata e in uscita e, se applicabile, il traffico IPv6.

The screenshot shows the AWS VPC Network ACLs page. A red box highlights the 'Regole in entrata (2)' (Inbound rules (2)) section. A red box highlights the 'Modifica le regole in entrata' (Edit inbound rules) button. A red box highlights the 'Consenti/Rifiuta' (Allow/Deny) column. The table shows two rules:

Numero di regola	Tipo	Protocollo	Intervallo di porte	Origine	Consenti/Rifiuta
100	Tutto il traffico	Tutto	Tutto	0.0.0.0/0	Allow
*	Tutto il traffico	Tutto	Tutto	0.0.0.0/0	Deny

- Puoi creare una lista di controllo accessi di rete personalizzata e associarla a una sottorete. Per impostazione predefinita, ogni lista di controllo accessi di rete personalizzata rifiuta tutto il traffico in entrata e in uscita finché non si aggiungono regole.

Regole in entrata (1)						Modifica le regole in entrata
	Filtra regole in entrata	< 1 >	Reset			
Numero di regola	Tipo	Protocollo	Intervallo di porte	Origine	Consenti/Rifiuta	
*	Tutto il traffico	Tutto	Tutto	0.0.0.0/0	<input checked="" type="checkbox"/> Deny	

- Ogni sottorete nel VPC deve essere associata a una lista di controllo accessi di rete. Se non associ in maniera esplicita una sottorete a una lista di controllo accessi di rete, la sottorete viene associata automaticamente alla lista di controllo accessi di rete predefinita.
- Puoi associare una lista di controllo accessi di rete a più sottoreti. Tuttavia, una sottorete può essere associata a una sola lista di controllo accessi di rete alla volta. Quando associ una lista di controllo accessi di rete a una sottorete, l'associazione precedente viene rimossa.
- Una lista di controllo accessi di rete contiene un elenco numerato di regole. Si valutano le regole in ordine, partendo dalla regola con numerazione più bassa, per determinare se il traffico è consentito in entrata o in uscita da qualsiasi sottorete associata alla lista di controllo accessi di rete. Il numero più alto che puoi utilizzare per un regola è 32766. Ti consigliamo di iniziare creando regole in incrementi (ad esempio, incrementi di 10 o 100) in modo da poter inserire nuove regole se richiesto in seguito.
- Una lista di controllo accessi di rete dispone di regole in entrata e in uscita separate, e ciascuna regola può consentire o rifiutare traffico.
- Le liste di controllo accessi di rete sono stateless, il che significa che le risposte al traffico in entrata consentito sono soggette alle regole per il traffico in uscita (e viceversa).

#### 5.4.1 Regole di liste di controllo accessi di rete, [1]

Puoi aggiungere o rimuovere regole dalla lista di controllo accessi di rete predefinita o creare liste di controllo accessi di rete aggiuntive per il VPC. Quando aggiungi o rimuovi regole da una lista di controllo accessi di rete, le modifiche vengono applicate automaticamente alle sottoreti cui è associata.

Di seguito sono riportate le parti di una regola della lista di controllo accessi di rete:

- **Numero regola.** Le regole sono valutate a partire da quella con numerazione più bassa. Non appena una regola corrisponde al traffico, viene applicata a prescindere da qualsiasi altra regola con numerazione più alta che potrebbe contraddirla.
- **Tipo.** Il tipo di traffico; ad esempio, SSH. Puoi anche specificare tutto il traffico o un intervallo personalizzato.
- **Protocol (Protocollo).** Puoi specificare qualsiasi protocollo che dispone di un numero di protocollo standard.
- **Intervallo porte.** La porta di ascolto o l'intervallo di porte per il traffico. Ad esempio, 80 per il traffico HTTP.
- **Source (Origine).** [Solo regole in entrata] L'origine del traffico (intervallo CIDR).
- **Destination (Destinazione).** [Solo regole in uscita] La destinazione per il traffico (intervallo CIDR).
- **Consenti/Nega.** Scelta tra le opzioni *allow* o *deny* per il traffico specificato.

Un esempio:

In entrata						
Rule #	Tipo	Protocollo	Intervallo porte	Crea	Consenti/Nega	Commenti
100	HTTP	TCP	80	0.0.0.0/0	PERMETTI	Consente traffico HTTP in entrata da qualunque indirizzo IPv4.
110	HTTPS	TCP	443	0.0.0.0/0	PERMETTI	Consente traffico HTTPS in entrata da qualsiasi indirizzo IPv4.
120	SSH	TCP	22	192.0.2.0/24	PERMETTI	Consente traffico SSH in entrata dall'intervallo di indirizzi IPv4 pubblici della rete privata (su Internet gateway).
130	RDP	TCP	3389	192.0.2.0/24	PERMETTI	Consente traffico RDP in entrata ai server Web dall'intervallo di indirizzi IPv4 pubblici della rete privata (su Internet gateway).
140	Custom TCP	TCP	32768-65535	0.0.0.0/0	PERMETTI	Consente traffico IPv4 di ritorno in entrata da Internet (ovvero, per richieste che originano nella sottorete).  Questo intervallo è solo un esempio. Per ulteriori informazioni su come selezionare l'intervallo di porte temporanee appropriato, consulta <a href="#">Porte Effimere</a> .
*	All traffic	Tutti	Tutti	0.0.0.0/0	RIFIUTA	Rifiuta tutto il traffico IPv4 in entrata che non è già gestito da una regola precedente (non modificabile).
In uscita						
Rule #	Tipo	Protocollo	Intervallo porte	Destinazione	Consenti/Nega	Commenti
100	HTTP	TCP	80	0.0.0.0/0	PERMETTI	Permette traffico HTTP IPv4 in uscita dalla sottorete a Internet.
110	HTTPS	TCP	443	0.0.0.0/0	PERMETTI	Permette traffico HTTPS IPv4 in uscita dalla sottorete a Internet.
120	SSH	TCP	1024-65535	192.0.2.0/24	PERMETTI	Consente traffico SSH in uscita dall'intervallo di indirizzi IPv4 pubblici della rete privata (su Internet gateway).
140	Custom TCP	TCP	32768-65535	0.0.0.0/0	PERMETTI	Permette risposte IPv4 in uscita a client su Internet (ad esempio, distribuzione di pagine Web a persone che visitano i server Web nella sottorete).  Questo intervallo è solo un esempio. Per ulteriori informazioni su come selezionare l'intervallo di porte temporanee appropriato, consulta <a href="#">Porte Effimere</a> .
*	All traffic	Tutti	Tutti	0.0.0.0/0	RIFIUTA	Rifiuta tutto il traffico IPv4 in uscita che non è già gestito da una regola precedente (non modificabile).

#### 5.4.2 Porte Effimere, [1]

La lista di controllo accessi di rete di esempio nella sezione precedente utilizza un intervallo di porte Effimere di 32768-65535. Tuttavia, potrebbe essere necessario utilizzare un intervallo diverso per le liste di controllo accessi di rete a seconda del tipo di client in uso o con cui si comunica.

Il client che avvia la richiesta sceglie l'intervallo di porte Effimere. L'intervallo varia a seconda del sistema operativo del client.

- Molti kernel Linux (incluso il kernel Amazon Linux) usano le porte 32768-61000.
- Le richieste provenienti da Elastic Load Balancing utilizzano le porte 1024-65535.
- I sistemi operativi Windows tramite Windows Server 2003 utilizzano porte 1025-5000.
- Windows Server 2008 e versioni successive utilizzano porte 49152-65535.
- Un gateway NAT utilizza le porte 1024-65535.
- AWS Lambda funzioni utilizzano le porte 1024-65535.

Ad esempio, se una richiesta arriva in un server Web nel VPC da un client Windows 10 su Internet, la lista di controllo degli accessi di rete deve disporre di una regola in uscita per abilitare il traffico destinato alle porte 49152-65535.

Se un'istanza nel VPC è il client che avvia una richiesta, la lista di controllo accessi di rete deve disporre di una regola in entrata per abilitare il traffico destinato alle porte temporanee specifiche per il tipo di istanza (Amazon Linux, Windows Server 2008 e così via).

In pratica, per coprire i diversi tipi di client che possono avviare il traffico su istanze rivolte al pubblico nel VPC, puoi aprire porte Effimere 1024-65535. Tuttavia, puoi anche aggiungere regole alla lista di controllo accessi per rifiutare il traffico su porte dannose all'interno di tale intervallo. Accertati di posizionare le regole deny il prima possibile nella tabella rispetto alle regole allow che aprono l'ampio intervallo di porte temporanee.

## 5.4.3 Utilizzo di ACL di rete, [1]

### 5.4.3.1 Determinazione delle associazioni della lista di controllo accessi di rete, [1]

Puoi utilizzare la console Amazon VPC per determinare la lista di controllo accessi di rete che è associata a una sottorete. Le liste di controllo accessi di rete possono essere associate a più sottoreti, pertanto puoi anche determinare quali sottoreti sono associate a una lista di controllo accessi di rete.

**Per determinare quale lista di controllo accessi di rete è associata a una sottorete:**

1. Accedere alla console Amazon VPC:

The screenshot shows the AWS Management Console navigation bar. The 'Servizi' tab is selected. A search bar contains the text 'Ricerca di servizi, caratteristiche, blog, documenti [Opzione+S]'. To the right are icons for notifications, help, and account information, followed by 'Virginia settentrionale ▾'. Below the navigation bar, a list of services is displayed in two columns. The first column includes: Elaborazione degli utenti finali, Gestione costi AWS, Gestione e governance, Integrazione di applicazioni, Internet of Things, Machine Learning, Migrazione e trasferimento, Quantum Technologies, Realtà aumentata e realtà virtuale. The second column includes: AWS Cloud Map (description: Crea una mappa dinamica del cloud), CloudFront (description: Rete globale di distribuzione dei contenuti), Direct Connect (description: Connessione di rete dedicata ad AWS), Global Accelerator (description: Migliora la disponibilità e le prestazioni della tua applicazione utilizzando la rete globale AWS), Route 53 (description: DNS scalabile e registrazione dei nomi dei domini), and Amazon VPC IP Address Manager (description: Servizio di gestione degli indirizzi IP gestiti). A red box highlights the 'Reti e distribuzione di contenuti' service, which is circled with a red number 1. Another red box highlights the 'VPC' service, which is circled with a red number 2.

Elaborazione degli utenti finali	AWS Cloud Map Crea una mappa dinamica del cloud
Gestione costi AWS	CloudFront Rete globale di distribuzione dei contenuti
Gestione e governance	Direct Connect Connessione di rete dedicata ad AWS
Integrazione di applicazioni	Global Accelerator Migliora la disponibilità e le prestazioni della tua applicazione utilizzando la rete globale AWS
Internet of Things	Route 53 DNS scalabile e registrazione dei nomi dei domini
Machine Learning	
Migrazione e trasferimento	
Quantum Technologies	
Realtà aumentata e realtà virtuale	
<b>Reti e distribuzione di contenuti</b>	
Robotica	
Satellite	
Servizi multimediali	
Sicurezza, identità, conformità	
Storage	
<b>VPC</b> Risorse cloud isolate	Amazon VPC IP Address Manager Servizio di gestione degli indirizzi IP gestiti

2. Nel riquadro di navigazione, scegliere **Subnets (Sottoreti)** e selezionare la sottorete. La lista di controllo accessi di rete associata alla sottorete è inclusa nella scheda Network ACL (lista di controllo accessi di rete), insieme alle regole della lista di controllo accessi di rete.

**Sottoreti (1/6) Informazioni**

<input type="checkbox"/>	public-C	subnet-[REDACTED]	<input checked="" type="checkbox"/> Available	10.2.2.0/28
<input type="checkbox"/>	-	subnet-[REDACTED]	<input checked="" type="checkbox"/> Available	172.31.0.0/20
<input type="checkbox"/>	-	subnet-[REDACTED]	<input checked="" type="checkbox"/> Available	172.31.16.0/20
<input checked="" type="checkbox"/>	public-A	subnet-[REDACTED]	<input checked="" type="checkbox"/> Available	10.2.0.0/24
<input type="checkbox"/>	public-B	subnet-[REDACTED]	<input checked="" type="checkbox"/> Available	10.2.1.0/24
<input type="checkbox"/>	-	subnet-[REDACTED]	<input checked="" type="checkbox"/> Available	172.31.32.0/20

**Detailed Subnet Information for public-A:**

- Indirizzi IPv4 disponibili: 251
- VPC: [REDACTED]
- Assegna automaticamente Indirizzo IPv4 pubblico: Sì
- Indirizzi IPv6 disponibili: -
- VPC: [REDACTED]
- Assegna automaticamente Indirizzo IPv6: -

**Zona di disponibilità:** us-east-2a

**ACL di rete:** acl-[REDACTED] (highlighted with a red box)

**Other settings:**

- Assegna automaticamente l'indirizzo IPv4 di proprietà del cliente: No
- Pool IPv4 di proprietà del cliente: -
- Solo IPv6: No

(viceversa) Per determinare quale sottoreti sono associate a una lista di controllo accessi di rete:

- Nel riquadro di navigazione, selezionare **Network ACL (lista di controllo accessi di rete)**. Nella colonna **Associated To (Associato a)** è indicato il numero di sottoreti associate per ogni lista di controllo accessi di rete.

**ACL di rete (6) Informazioni**

	Name	ID lista di controllo...	Associato a	di default	ID VPC
<input type="checkbox"/>	-	acl-[REDACTED]	3 Sottoreti	Sì	vpc-[REDACTED]
<input type="checkbox"/>	web-traffic	acl-[REDACTED]	-	No	vpc-[REDACTED]
<input type="checkbox"/>	-	acl-[REDACTED]	-	Sì	vpc-[REDACTED]
<input type="checkbox"/>	-	acl-[REDACTED]	3 Sottoreti	Sì	vpc-[REDACTED]
<input type="checkbox"/>	NACL-2019	acl-[REDACTED]	-	No	vpc-[REDACTED]
<input type="checkbox"/>	web-trafficc	acl-[REDACTED]	-	No	vpc-[REDACTED]

- Selezionare una lista di controllo accessi di rete.
- Nel riquadro dei dettagli, scegliere **Subnet Associations (Associazioni sottorete)** per visualizzare le sottoreti che sono associate alla lista di controllo accessi di rete.

The screenshot shows the AWS VPC ACL management interface. On the left, the navigation sidebar includes links for 'Virtual Private Cloud', 'Sicurezza', 'Liste di controllo degli accessi di rete' (highlighted with a red circle), 'Gruppi di sicurezza', 'Network Analysis', 'Firewall DNS', and 'Network Firewall'. The main area displays an 'ACL di rete (1/6)' table with three rows. The second row, which has a checked checkbox, is highlighted with a red box. A red circle with the number '1' is on the 'Liste di controllo degli accessi di rete' link. A red circle with the number '2' is on the 'Associazioni sottorete' tab of the detailed view for 'acl-0b64fd6z'.

#### 5.4.3.2 Creazione di una lista di controllo degli accessi di rete, [1]

Puoi creare una lista di controllo accessi di rete personalizzata dal VPC. Per impostazione predefinita, una lista di controllo accessi di rete creata dall'utente blocca tutto il traffico in entrata e in uscita finché non si aggiungono regole. E non è associata a una sottorete finché una non viene associata in maniera esplicita.

**Per creare una lista di controllo accessi di rete:**

1. Nel riquadro di navigazione, selezionare **Network ACL (lista di controllo accessi di rete)**.
2. Selezionare **Create Network ACL (Crea una lista di controllo accessi della rete)**.

The screenshot shows the AWS VPC ACL management interface. The navigation sidebar is identical to the previous one. The main area displays an 'ACL di rete (6)' table. A red circle with the number '1' is on the 'Liste di controllo degli accessi di rete' link. A red circle with the number '2' is on the 'Crea una lista di controllo accessi della rete' button in the top right corner of the table header.

3. Nella finestra di dialogo **Create Network ACL (Crea lista di controllo accessi di rete)**, assegnare facoltativamente un nome alla lista di controllo accessi di rete e selezionare l'ID del VPC dall'elenco **VPC**. Quindi selezionare **Crea una lista di controllo accessi della rete**.

**Impostazioni ACL di rete**

**Nome - facoltativo**  
Crea un tag con una chiave del 'nome' e un valore specificato.

my-acl-01

1

**VPC**  
VPC da utilizzare per questa ACL di rete.

Seleziona un VPC

2

**Tag**  
Un tag è un'etichetta assegnata a una risorsa AWS. Ogni tag è composto da una chiave e da un valore facoltativo. È possibile usare i tag per ricercare e filtrare le risorse o monitorare i tuoi costi di AWS.

Nessun tag associato alla risorsa.

**Aggiungi nuovo tag**  
È possibile aggiungere 50 altri tag.

3

[Annulla](#) Crea una lista di controllo accessi della rete

#### 5.4.3.3 Aggiunta ed eliminazione di regole, [1]

Quando aggiungi o elimini una regola da una lista di controllo accessi, le eventuali sottoreti associate alla lista di controllo accessi sono influenzate dalla modifica. Non occorre terminare e avviare nuovamente le istanze nella sottorete. Le modifiche diventano effettive dopo un breve periodo di tempo.

**Attenzione:** *È necessario prestare molta attenzione se si aggiungono ed eliminano regole contemporaneamente. Le regole della lista di controllo degli accessi di rete definiscono quali tipi di traffico di rete possono entrare o uscire dai VPC. Se si eliminano regole in entrata o in uscita e quindi si aggiungono nuove voci rispetto a quelle consentite in Quote Amazon VPC, le voci selezionate per l'eliminazione verranno rimosse e le nuove voci non verranno aggiunte. Ciò potrebbe causare problemi di connettività imprevisti e impedire involontariamente l'accesso da e verso i VPC.*

Se stai utilizzando l'API Amazon EC2 o uno strumento a riga di comando, non puoi modificare le regole. Puoi solo aggiungere ed eliminare regole. Se stai utilizzando la console Amazon VPC, puoi modificare le voci relative alle regole esistenti. La console rimuove la regola esistente e aggiunge una nuova regola automaticamente. Se occorre modificare l'ordine di una regola nella lista di controllo accessi, devi aggiungere una nuova regola con il nuovo numero regola , quindi eliminare la regola originale.

**Per aggiungere regole a una lista di controllo accessi di rete:**

1. Nel riquadro di navigazione, selezionare **Network ACL (lista di controllo accessi di rete)**.
2. Nel riquadro dei dettagli, scegliere la scheda **Inbound Rules (Regole in entrata)** o **Outbound Rules (Regole in uscita)**, in base al tipo di regola che occorre aggiungere, quindi selezionare **Modifica le regole in entrata**.

The screenshot shows the AWS VPC ACLs interface. On the left sidebar, under 'SICUREZZA', the 'Liste di controllo degli accessi di rete' option is selected (marked with a red box and number 1). In the main area, the 'Regole in entrata' tab is active (marked with a red box and number 3). At the top right of this tab, there is a button labeled 'Modifica le regole in entrata' (marked with a red box and number 4).

3. In **Rule # (N. regola)**, immettere un numero regola (ad esempio, 100). Il numero regola non deve già essere in uso nella lista di controllo accessi di rete. AWS elabora le regole nell'ordine, partendo da quella con il numero più basso.  
È consigliato lasciare degli spazi vuoti tra i numeri regola (ad esempio 100, 200, 300), anziché utilizzare numeri in sequenziali (101, 102, 103). Questo semplifica l'aggiunta di una nuova regola senza la necessità di numerare le regole Esistenti.
4. Selezionare una regola dall'elenco **Type (Tipo)**. Ad esempio, per aggiungere una regola per HTTP, scegliere **HTTP**. Per aggiungere una regola per consentire tutto il traffico TCP, scegliere **Tutti i protocolli TCP**. Per alcune di queste opzioni (ad esempio, HTTP), la porta viene compilata automaticamente. Per utilizzare un protocollo non elencato, scegliere **Custom Protocol (Protocollo personalizzato)**.
5. (Facoltativo) Se si sta creando una regola protocollo personalizzata, selezionare il numero e il nome del protocollo dall'elenco **Protocol (Protocollo)**.
6. (Facoltativo) Se il protocollo selezionato richiede un numero di porta, immettere il numero di porta o l'intervallo di porte separato da un trattino (ad esempio, 49152-65535).
7. Nel campo **Source (Origine)** o **Destination (Destinazione)** (a seconda che si tratti di una regola in entrata o in uscita), immettere l'intervallo CIDR cui si applica la regola.
8. Dall'elenco **Allow/Deny (Consenti/Rifiuta)**, selezionare **ALLOW** per consentire il traffico specificato o **DENY** per rifiutare il traffico specificato.
9. (Facoltativo) Per aggiungere un'altra regola, selezionare **Add new rule (Aggiungi nuova regola)** e ripetere le fasi da 3 a 8 come richiesto.
10. Al termine, scegliere **Save changes (Salva modifiche)**.

## Modifica le regole in entrata Informazioni

Le regole per il traffico in entrata controllano il traffico in ingresso a cui è consentito raggiungere il VPC.

Numero di regola <small>Informazioni</small>	Tipo <small>Informazioni</small>	Protocollo <small>Informazioni</small>	Intervallo di porte <small>Informazioni</small>	Origine <small>Informazioni</small>	Consenti/Rifiuta <small>Informazioni</small>
<input type="text"/>	HTTP (80) ▾	TCP (6) ▾	80	0.0.0.0/0	Consenti ▾
*	Tutto il traffico ▾	Tutto ▾	Tutto	0.0.0.0/0	Rifiuta ▾
<input type="button" value="Aggiungi nuova regola"/> <input type="button" value="Ordina per numero di regola"/>					
<input type="button" value="Annulla"/> <input type="button" value="Anteprima delle modifiche"/> <input style="background-color: orange; color: white; border: none;" type="button" value="Salva modifiche"/>					

### Per eliminare una regola da una lista di controllo accessi di rete:

1. Nel riquadro di navigazione, scegliere **Network ACLs (Liste di controllo degli accessi di rete)**, quindi selezionare la lista di controllo accessi di rete.
2. Nel riquadro dei dettagli, selezionare la scheda **Inbound Rules (Regole in entrata)** o **Outbound Rules (Regole in uscita)**, quindi selezionare **Modifica le regole in entrata/uscita**.

The screenshot shows the AWS Network ACLs Inbound Rules page. On the left sidebar, under 'SICUREZZA', the 'Liste di controllo degli accessi di rete' option is highlighted with a red box and numbered 1. In the main content area, the 'Regole in entrata' tab is selected (highlighted with a red box and numbered 3). A specific rule, 'acl-6', is selected and highlighted with a red box and numbered 2. At the bottom right of the rules list, there is a red box around the 'Modifica le regole in entrata' button, which is also highlighted with a red circle and numbered 4.

3. Selezionare **Remove (Rimuovi)** per la regola da eliminare, quindi selezionare **Save change (Salva modifiche)**.

## Modifica le regole in uscita Informazioni

Le regole in uscita controllano il traffico in uscita che può lasciare il VPC.

Numero di regola <small>Informazioni</small>	Tipo <small>Informazioni</small>	Protocollo <small>Informazioni</small>	Intervallo di porte <small>Informazioni</small>	Destinazione <small>Informazioni</small>	Consenti/Rifiuta <small>Informazioni</small>
101	Tutti i protocol... ▾	TCP (6) ▾	Tutto	37.102.26.170/32	Consenti ▾
*	Tutto il traffico ▾	Tutto ▾	Tutto	0.0.0.0/0	Rifiuta ▾
<input type="button" value="Aggiungi nuova regola"/> <input type="button" value="Ordina per numero di regola"/>					
<input type="button" value="Annulla"/> <input type="button" value="Anteprima delle modifiche"/> <input style="background-color: orange; color: white; border: none;" type="button" value="Salva modifiche"/>					

#### 5.4.3.4 Associazione di una sottorete a una lista di controllo accessi di rete, [1]

Per applicare le regole di una lista di controllo accessi di rete a una particolare sottorete, occorre associare la sottorete alla lista di controllo accessi di rete. Puoi associare una lista di controllo accessi di rete a più sottoreti. Tuttavia, una sottorete può essere associata a una sola lista di controllo accessi di rete. Eventuali sottoreti non associate a una particolare lista di controllo accessi vengono associate per impostazione predefinita alla lista di controllo accessi di rete predefinita.

##### Per associare una sottorete a una lista di controllo accessi di rete:

- Nel riquadro di navigazione, scegliere **Network ACLs (liste di controllo degli accessi di rete)**, quindi selezionare la lista di controllo accessi di rete. Nel riquadro dei dettagli, nella scheda **Subnet Associations (Associazioni sottorete)** scegliere **Modifica associazioni sottorete**.

Name	ID lista di controllo...	Associato a
<input checked="" type="checkbox"/> [REDACTED]	acl-[REDACTED]	3 Sottoreti
<input type="checkbox"/> -	acl-[REDACTED]	subnet-[REDACTED] / kcluster-subnet

- Selezionare la casella di controllo per la sottorete da associare alla lista di controllo accessi di rete, quindi selezionare **Save change (Salva modifiche)**.

#### Modifica associazioni sottorete Informazioni

Modifica le sottoreti associate a questa ACL di rete.

Nome	ID sottorete	Associato a	Zona di disponibilità	CIDR IPv4
<input checked="" type="checkbox"/> sottoreteB	subnet-[REDACTED]	[REDACTED]	us-east-1b	10.5.2.0/24
<input checked="" type="checkbox"/> sottoreteC	subnet-[REDACTED]	[REDACTED]	us-east-1c	10.5.3.0/24
<input checked="" type="checkbox"/> sottoreteA	subnet-[REDACTED]	[REDACTED]	us-east-1a	10.5.1.0/24

**Sottoreti selezionate**

subnet-[REDACTED] / sottoreteB X    subnet-[REDACTED] / sottoreteC X    subnet-[REDACTED] / sottoreteA X

**Salva modifiche**

#### 5.4.3.5 Annullamento dell'associazione di una lista di controllo accessi di rete a una sottorete, [1]

È possibile annullare l'associazione di una lista di controllo accessi di rete personalizzata da una sottorete. Quando viene annullata l'associazione della sottorete dalla lista di controllo accessi di rete personalizzata, la sottorete viene quindi associata automaticamente alla lista di controllo accessi di rete predefinita.

**Per annullare l'associazione di una sottorete a una lista di controllo accessi di rete:**

- Nel riquadro di navigazione, scegliere **Network ACLs (liste di controllo degli accessi di rete)**, quindi selezionare la lista di controllo accessi di rete. Nel riquadro dei dettagli, selezionare la scheda **Subnet Associations (Associazioni sottorete)** scegliere **Modifica associazioni sottorete**.

The screenshot shows the AWS VPC ACLs Subnet Associations page. The left sidebar has sections like VIRTUAL PRIVATE CLOUD, SICUREZZA (with 'Lista di controllo degli accessi di rete' highlighted), NETWORK ANALYSIS, and FIREWALL DNS. The main area shows an ACL named 'acl-[REDACTED]' with three subnet associations. A red box highlights the first association ('3 Sottoreti'). Step 1 is indicated by a red circle on the 'SICUREZZA' section. Step 2 is indicated by a red circle on the first row of the table. Step 3 is indicated by a red circle on the 'Associazioni sottorete' tab. Step 4 is indicated by a red circle on the 'Modifica associazioni sottorete' button.

Name	ID lista di controllo...	Associato a
[REDACTED]	acl-[REDACTED]	3 Sottoreti
-	acl-[REDACTED]	subnet-[REDACTED] / kcluster-subnet

- Deselezionare la casella di controllo per la sottorete. Selezionare **Save change (Salva modifiche)**.

The screenshot shows the 'Modifica associazioni sottorete' dialog. It lists available subnets and selected ones. Step 1 is indicated by a red circle on the checkbox for 'sottoreteB'. Step 2 is indicated by a red circle on the 'Salva modifiche' button.

Nome	ID sottorete	Associato a	Zona di disponibili
sottoreteB	subnet-[REDACTED]	[REDACTED]	us-east-1b
sottoreteC	subnet-[REDACTED]	[REDACTED]	us-east-1c
sottoreteA	subnet-[REDACTED]	[REDACTED]	us-east-1a

**Sottoreti selezionate:**  
subnet-[REDACTED] / sottoreteB X

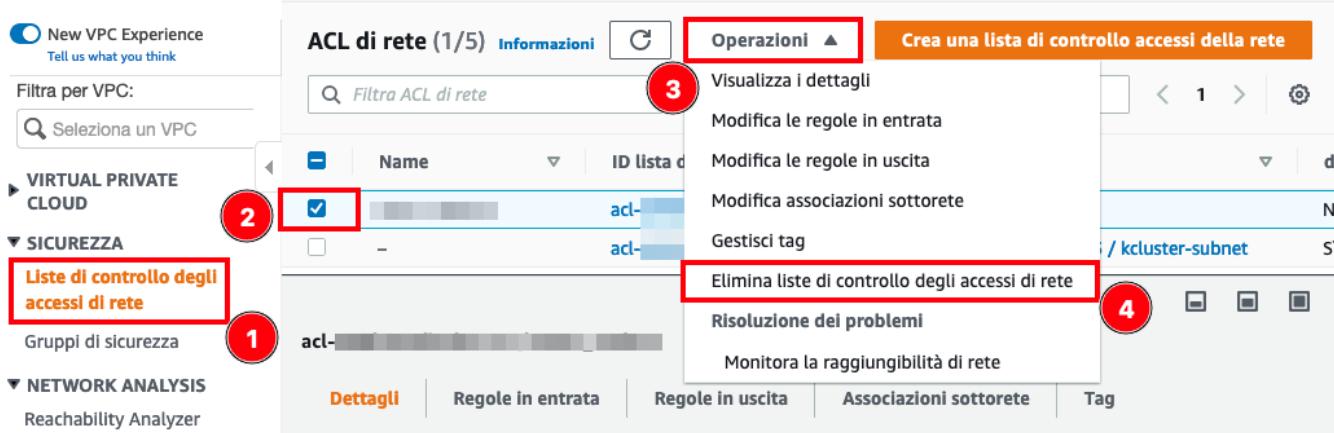
Buttons: Annulla, Salva modifiche

#### 5.4.3.6 Eliminazione di una lista di controllo accessi di rete, [1]

Puoi eliminare una lista di controllo accessi di rete solo se a essa non sono associate sottoreti. Non puoi eliminare la lista di controllo accessi di rete predefinita.

Per eliminare una lista di controllo accessi di rete:

- Nel riquadro di navigazione, selezionare **Network ACL (lista di controllo degli accessi di rete)**, selezionare la lista di controllo accessi di rete, quindi selezionare **Operazioni** infine **Elimina liste di controllo degli accessi di rete**.



- Nella finestra di dialogo di conferma, scegliere **Yes, Delete (Sì, elimina)**.

#### 5.5 Gruppi di sicurezza, [1]

Un gruppo di sicurezza funge da firewall virtuale e controlla il traffico consentito per raggiungere e lasciare le risorse a cui è associato. Ad esempio, dopo aver associato un gruppo di sicurezza a un'istanza EC2, controlla il traffico in entrata e in uscita per l'istanza.

Al momento della creazione di un VPC, questo include un gruppo di sicurezza di default. È possibile creare gruppi di sicurezza aggiuntivi per ogni VPC. È possibile associare un gruppo di sicurezza solo alle risorse nel VPC per il quale viene creato.

Per ogni gruppo di sicurezza, aggiungere regole per controllare il traffico in base ai protocolli e ai numeri di porta. Esistono insiemi separati di regole per il traffico in entrata e il traffico in uscita.

Si possono impostare liste di controllo accessi di rete con regole simili a quelle del gruppo di sicurezza, in modo tale da aggiungere un ulteriore livello di sicurezza al VPC.

Di seguito sono riportate le caratteristiche dei gruppi di sicurezza:

- I gruppi di sicurezza sono *stateful*. Ad esempio, inviando una richiesta da un'istanza, il traffico in risposta alla richiesta è autorizzato a raggiungerla, indipendentemente dalle regole in entrata del gruppo di sicurezza. Le risposte al traffico in entrata autorizzato possono lasciare l'istanza indipendentemente dalle regole in uscita.
- Esistono dei limiti (detti anche quote) per il numero di gruppi di sicurezza che si possono creare per ogni VPC, al numero di regole che si possono aggiungere a ciascun gruppo di sicurezza e al numero di gruppi di sicurezza che si possono associare a un'interfaccia di rete.

Di seguito sono riportate le caratteristiche delle regole dei gruppi di sicurezza:

- Puoi specificare regole che autorizzano, non regole che negano.
- Al momento della sua creazione, un gruppo di sicurezza è privo di regole in entrata. Di conseguenza, non è consentito alcun traffico in entrata fino a quando al gruppo di sicurezza non vengono aggiunte regole in entrata.

- Quando si crea per la prima volta un gruppo di sicurezza, questo include una regola in uscita che consente tutto il traffico in uscita dalla risorsa. Puoi rimuovere la regola e aggiungere regole in uscita che autorizzano l'uscita solo di un determinato tipo di traffico. Se un gruppo di sicurezza è privo di regole in uscita, non viene autorizzato alcun traffico in uscita.
- Se si associano a una risorsa molteplici gruppi di sicurezza, le regole di ciascun gruppo di sicurezza vengono aggregate efficacemente per creare un unico set di regole utilizzate per determinare se consentire l'accesso o meno.
- Quando si aggiunge, aggiorna o rimuove delle regole, queste si applicano automaticamente a tutte le risorse associate al gruppo di sicurezza.
- Quando si crea una regola del gruppo di sicurezza, AWS assegna un ID univoco alla regola. È possibile utilizzare l'ID di una regola quando si utilizza l'API o la CLI per modificare o eliminare la regola.

### Gruppi di sicurezza di default per VPC:

I VPC predefiniti e tutti i VPC creati includono un gruppo di sicurezza di default. Con alcune risorse, se non si associa un gruppo di sicurezza quando si crea la risorsa, questa viene associata al gruppo di sicurezza di default. Ad esempio, non specificando un gruppo di sicurezza all'avvio di un'istanza EC2, questa viene associata al gruppo di sicurezza di default. È possibile modificare le regole di un gruppo di sicurezza di default. Non è possibile eliminare un gruppo di sicurezza predefinito. Se provi a eliminare il gruppo di sicurezza predefinito, visualizzi il seguente errore: *Client.CannotDelete*.

La tabella seguente descrive le regole predefinite di un gruppo di sicurezza predefinito.

Inbound			
Crea	Protocollo	Intervallo porte	Descrizione
L'ID del gruppo di sicurezza (ID della risorsa)	Tutti	Tutti	Consente il traffico in ingresso dalle risorse assegnate allo stesso gruppo di sicurezza.
Outbound			
Destinazione	Protocollo	Intervallo porte	Descrizione
0.0.0.0/0	Tutti	Tutti	Autorizza tutto il traffico IPv4 in uscita.
::/0	All	All	Allows all outbound IPv6 traffic. This rule is added only if your VPC has an associated IPv6 CIDR block.

#### 5.5.1 Regole del gruppo di sicurezza, [1]

Le regole di un gruppo di sicurezza controllano il traffico in entrata autorizzato a raggiungere le risorse associate al gruppo di sicurezza, e il traffico in uscita autorizzato a lasciarle.

Puoi aggiungere o rimuovere le regole di un gruppo di sicurezza. Una regola si applica al traffico in entrata o al traffico in uscita. Puoi autorizzare l'accesso a un intervallo CIDR specifico o a un altro gruppo di sicurezza nel VPC o in un VPC in peering.

Per ogni regola, occorre specificare quanto segue:

- **Tipo:** Protocollo da aprire al traffico di rete. Per consentire al traffico di Internet di raggiungere la tua istanza, puoi scegliere un protocollo comune come SSH (per

un'istanza Linux), RDP (per un'istanza Windows), HTTP e HTTPS. Puoi anche inserire manualmente una porta personalizzata o intervalli di porte.

- **Protocollo:** il protocollo da autorizzare. I protocolli più comuni sono 6 (TCP) 17 (UDP) e 1 (ICMP).
- **Intervallo di porte:** per un protocollo personalizzato o per TCP e UDP, l'intervallo di porte da autorizzare. Puoi specificare un solo numero di porta (ad esempio 22) o un intervallo dei numeri di porta (ad esempio 7000-8000).
- **Destinazione/Origine:** Stabilisce il traffico che può raggiungere la tua istanza. Specifica un singolo indirizzo IP o un intervallo di indirizzi IP nella notazione CIDR (ad esempio, 203.0.113.5/32). Se ti stai connettendo con la protezione di un firewall, devi specificare l'intervallo di indirizzi IP utilizzato dai computer client. Puoi specificare il nome o l'ID di un altro gruppo di sicurezza nella stessa regione. Per specificare un gruppo di sicurezza in un altro account AWS (solo EC2-Classic), inserisci l'ID account come prefisso e una barra, ad esempio: 111122223333/OtherSecurityGroup.
- **(Opzionale) Descrizione:** puoi aggiungere una descrizione della regola, per semplificare l'identificazione in un secondo momento.

Di seguito riportate delle regole di esempio:

Inbound			
Crea	Protocollo	Intervallo porte	Descrizione
0.0.0.0/0	TCP	80	Permette l'accesso HTTP in entrata da tutti gli indirizzi IPv4
::/0	TCP	80	Allows inbound HTTP access from all IPv6 addresses
0.0.0.0/0	TCP	443	Autorizza l'accesso HTTPS in entrata da tutti gli indirizzi IPv4
::/0	TCP	443	Allows inbound HTTPS access from all IPv6 addresses
Intervallo di indirizzi IPv4 pubblici della tua rete	TCP	22	Autorizza l'accesso SSH in entrata dagli indirizzi IP IPv4 nella rete
Intervallo di indirizzi IPv4 pubblici della tua rete	TCP	3389	Autorizza l'accesso RDP in entrata dagli indirizzi IP IPv4 nella rete
Outbound			
Destinazione	Protocollo	Intervallo porte	Descrizione
L'ID del gruppo di sicurezza dei server di database Microsoft SQL Server	TCP	1433	Autorizzare l'accesso Microsoft SQL Server in uscita
L'ID del gruppo di sicurezza dei server di database MySQL	TCP	3306	Autorizzare l'accesso MySQL in uscita

### 5.5.2 Creazione di un gruppo di sicurezza, [1]

Di default, i nuovi gruppi di sicurezza hanno solo una regola in uscita che autorizza tutto il traffico a lasciare la risorsa. Devi aggiungere le regole per autorizzare qualsiasi tipo di traffico in

entra o per limitare quello in uscita. Un gruppo di sicurezza può essere utilizzato solo nel VPC per cui viene creato.

1. Fare clic su **Security Groups (Gruppi di sicurezza)** nel riquadro di navigazione.

Scegliere **Create Security Group (Crea gruppo di sicurezza)**.

**Gruppi di sicurezza (78) Informazioni**

**C** Operazioni ▾ Esporta gruppi di sicurezza in CSV ▾ **Crea gruppo di sicurezza**

Filtra i gruppi di sicurezza

<input type="checkbox"/>	Name	ID gruppo di sicurezza	Nome del gruppo d...	ID VPC
<input type="checkbox"/>	-	sg-1 [REDACTED]	launch-wizard-16	vpc-[REDACTED]
<input type="checkbox"/>	-	sg-1 [REDACTED]	launch-wizard-23	vpc-[REDACTED]
<input type="checkbox"/>	-	sg-1 [REDACTED]	launch-wizard-20	vpc-[REDACTED]
<input type="checkbox"/>	-	sg-1 [REDACTED]	launch-wizard-8	vpc-[REDACTED]

2. Nella scheda **Dettagli di base** immettere un **Nome** e una **Descrizione** per il gruppo di sicurezza. Non è possibile modificare il nome e la descrizione di un gruppo di sicurezza dopo averlo creato. Da VPC, seleziona il **VPC**.

**Dettagli di base**

Nome del gruppo di sicurezza Informazioni **1**  
MyWebServerGroup

Il nome non può essere modificato a creazione ultimata.

Descrizione Informazioni **2**  
Consente l'accesso SSH agli sviluppatori

VPC Informazioni **3**  
vpc-[REDACTED]

3. È possibile aggiungere le regole del gruppo di sicurezza a questo punto oppure in un secondo momento.

**Regole in entrata Informazioni**

Questo gruppo di sicurezza non ha regole in entrata.

**Aggiungi regola**

**Regole in uscita Informazioni**

Tipo <small>Informazioni</small>	Protocollo <small>Informazioni</small>	Intervallo porte <small>Informazioni</small>	Destinazione <small>Informazioni</small>	Descrizione - facoltativa <small>Informazioni</small>
Tutto il traffico	Tutti	Tutti	Person... ▾	<input type="text"/> 0.0.0.0 <input type="button" value="X"/>

**Aggiungi regola**

4. È possibile aggiungere un tag a questo punto oppure in un secondo momento. Per aggiungere un tag, scegliere **Aggiungi nuovo tag**, quindi specifica la **Chiave** e il **Valore** (opzionale) del tag. Scegliere infine **Create Security Group (Crea gruppo di sicurezza)**.

### Tag opzionale

Un tag è un'etichetta assegnata a una risorsa AWS. Ogni tag è formato da una coppia comprendente una chiave e un valore facoltativo. È possibile utilizzare i tag per cercare e filtrare le risorse o monitorare i costi AWS.

Chiave  
Inserisci chiave

Valore - opzionale  
Inserisci valore

Rimuovi

⚠ Devi specificare una chiave di tag

Aggiungi nuovo tag 1

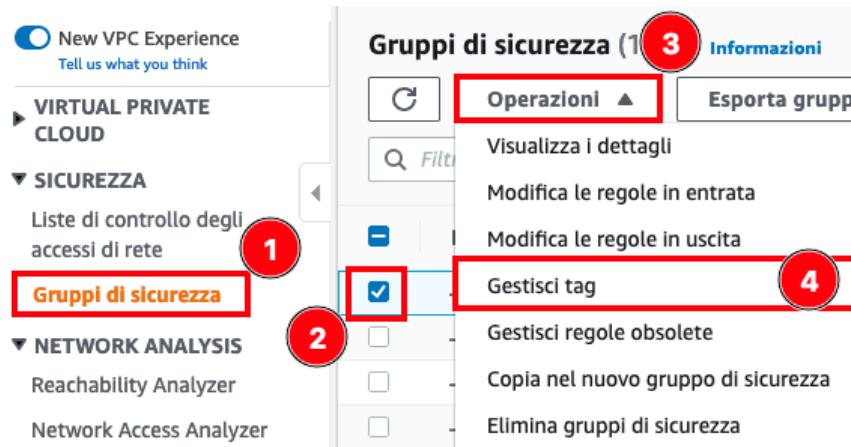
Puoi aggiungere fino a 49 tag in più

Annulla Crea gruppo di sicurezza 3

### 5.5.3 Tag dei gruppi di sicurezza, [1]

Aggiungi tag alle risorse per aiutare a organizzare e identificare, differenziando ad esempio per scopo, proprietario o ambiente. Puoi aggiungere i tag anche ai gruppi di sicurezza. Le chiavi dei tag devono essere univoche per ogni gruppo di sicurezza. Se aggiungi un tag con una chiave già associata alla regola, il valore del tag viene aggiornato.

1. **Fai clic su **Security Groups (Gruppi di sicurezza)** nel riquadro di navigazione.**  
Selezionare la casella accanto al gruppo di sicurezza. Scegliere **Actions (Operazioni)**, **Manage tags (Gestisci tag)**.



2. Nella sezione **Gestisci tag** vengono visualizzati tutti i tag assegnati al gruppo di sicurezza. Per aggiungere un tag, selezionare **Add new tag (Aggiungi nuovo tag)**, quindi specifica la chiave del tag e il suo valore. Per eliminare un tag, scegliere **Remove (Rimuovi)** accanto al tag che desideri eliminare. Selezionare infine **Save changes (Salva modifiche)**.

## Gestisci tag

Un tag è un'etichetta assegnata a una risorsa AWS. Ogni tag è formato da una coppia comprendente una chiave e un valore facoltativo. È possibile utilizzare i tag per cercare e filtrare le risorse o monitorare i costi AWS.

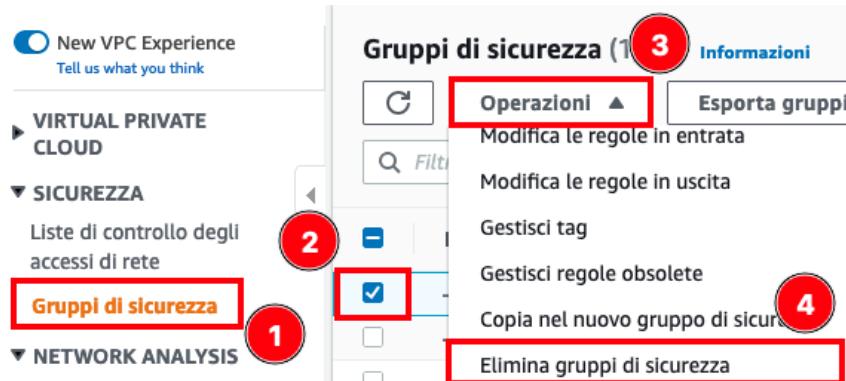
Chiave	Valore - opzionale
<input type="text" value="Inserisci chiave"/> 1	<input type="text" value="Inserisci valore"/> 2
<p><b>⚠ Devi specificare una chiave di tag</b></p>	
<b>Aggiungi nuovo tag</b> 3	
Puoi aggiungere fino a 49 tag in più	
<b>Annulla</b>	<b>Salva modifiche</b>

#### **5.5.4 Eliminare un gruppo di sicurezza, [1]**

È possibile eliminare un gruppo di sicurezza solo se non è associato a una risorsa. Non è possibile eliminare un gruppo di sicurezza predefinito.

Se utilizzi la console, puoi eliminare più di un gruppo di sicurezza alla volta. Se utilizzi la riga di comando o l'API, puoi eliminare solo un gruppo di sicurezza alla volta.

1. Fai clic su **Security Groups (Gruppi di sicurezza)** nel riquadro di navigazione. Seleziona uno o più gruppi di sicurezza e scegli **Operazioni** → **Elimina gruppo di sicurezza**.



2. Quando viene richiesta la conferma seleziona **Elimina**.



### 5.5.5 Aggiunta di regole a un gruppo di sicurezza, [1]

Quando si aggiunge una regola a un gruppo di sicurezza, la nuova regola viene applicata automaticamente a tutte le risorse associate al gruppo di sicurezza.

1. Fai clic su **Security Groups (Gruppi di sicurezza)** nel riquadro di navigazione. Selezionare il gruppo di sicurezza. Scegliere **Actions (Operazioni)** → **Edit inbound rules (Modifica le regole in entrata)** o **Actions (Operazioni)** → **Edit outbound rules (Modifica le regole in uscita)**.

2. Per ogni regola, seleziona **Aggiungi regola** e completa le attività riportate di seguito.

- Per **Type (Tipo)**, scegliere il tipo di protocollo consentito.
  - Per TCP o UDP, è necessario immettere l'intervallo di porte consentito.
  - Se si sceglie un protocollo ICMP personalizzato, occorre scegliere il nome del tipo ICMP da **Protocollo** e, se applicabile, il nome del codice da **Intervallo di porte**.
  - Se si sceglie qualsiasi altro tipo, il protocollo e l'intervallo di porte vengono configurati automaticamente.
- Per **Origine (regole in entrata)** o **Destinazione (regole in uscita)**, effettua una delle seguenti operazioni per consentire il traffico:
  - Scegli **Personalizzato**, quindi immetti un indirizzo IP in notazione CIDR, un blocco CIDR, un altro gruppo di sicurezza o un elenco di prefissi.
  - Scegliere **Anywhere (Ovunque)** per consentire l'accesso al traffico proveniente da qualsiasi indirizzo IP (regole in entrata) o per consentire al traffico di raggiungere tutti gli indirizzi IP (regole in uscita). Ciò aggiunge automaticamente una regola per il blocco CIDR IPv4 0.0.0.0/0. Se il gruppo di sicurezza si trova in un VPC abilitato per IPv6, ciò aggiunge automaticamente una regola per il blocco CIDR IPv6 ::/0. Per le regole in entrata, l'opzione è accettabile per un breve periodo di tempo in un ambiente di test, ma non è sicura per gli ambienti di produzione. In produzione, sarà possibile autorizzare l'accesso solo a un determinato indirizzo IP o a un intervallo di indirizzi IP.
  - Seleziona **Il mio IP** per permettere il traffico solo da (regole in entrata) o verso (regole in uscita) l'indirizzo IPv4 pubblico del computer locale.

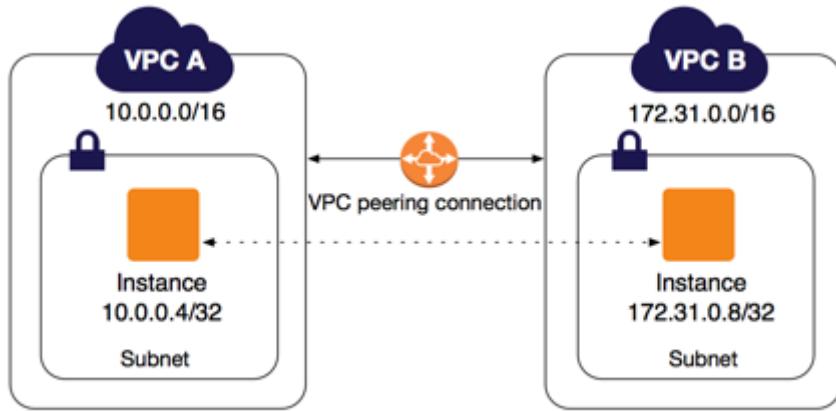
c. (Facoltativo) Per **Descrizione**, specifica una breve descrizione della regola.

3. Scegliere **Save rules (Salva regole)**.

I passaggi per aggiornare, taggare ed eliminare delle regole di un gruppo di sicurezza sono simili, vedi i paragrafi successivi nella documentazione ufficiale.

## 5.6 VPC Peering, [1]

Una connessione peering VPC è una connessione di rete tra due VPC che ti consente di instradare il traffico tra gli stessi utilizzando indirizzi IPv4 o IPv6 privati. Le istanze in uno qualsiasi dei VPC possono comunicare tra loro come se fossero nella stessa rete. È possibile creare una connessione peering VPC tra i VPC oppure con un VPC in un altro account AWS. I VPC possono essere in regioni differenti e in tal caso si parla di connessione peering VPC interregionale.



AWS utilizza l'infrastruttura esistente di un VPC per creare una connessione peering VPC; non si tratta di un gateway o di una connessione VPN e non dipende da un elemento hardware fisico distinto. Non prevede alcun singolo punto di errore né colli di bottiglia.

Una connessione peering VPC facilita il trasferimento di dati. Ad esempio, se sono presenti più account AWS, è possibile collegare in peering i VPC di tali account per creare una rete di condivisione di file. Puoi anche utilizzare una connessione peering VPC per consentire ad altri VPC di accedere alle risorse disponibili in uno dei VPC.

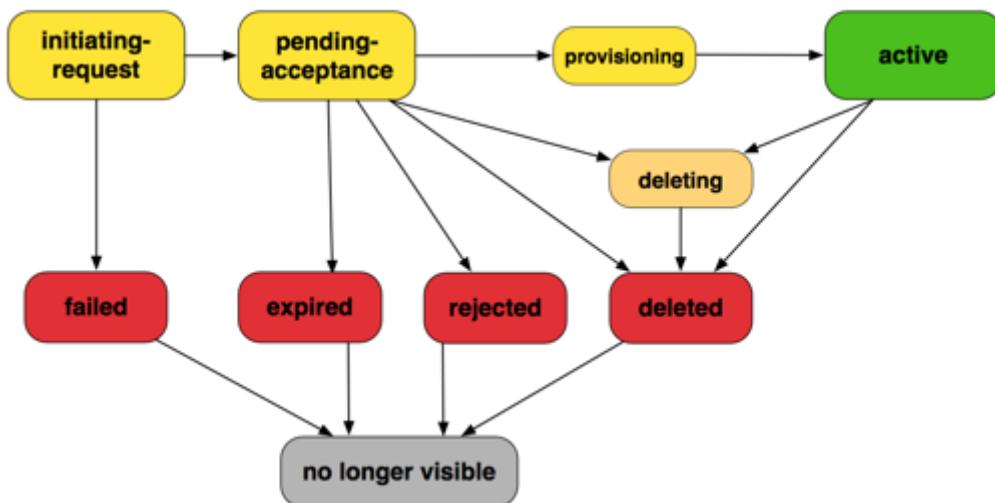
È possibile stabilire relazioni di peering tra VPC in regioni AWS diverse (operazione chiamata anche peering VPC tra regioni). Ciò consente alle risorse VPC, incluse le istanze EC2, i database Amazon RDS e le funzioni Lambda eseguite in diverse regioni AWS, di comunicare tra loro utilizzando indirizzi IP privati, senza richiedere gateway, connessioni VPN o appliance di rete separate. Il traffico rimane nello spazio degli IP privati. Tutto il traffico interregionale viene crittografato senza alcun singolo punto di errore o colli di bottiglia della larghezza di banda. Il traffico rimane sempre sulla dorsale AWS globale e non attraversa mai la rete Internet pubblica, riducendo così le minacce, come gli exploit comuni e gli attacchi DDoS. Il peering di VPC interregionale fornisce un modo semplice ed economico di condividere le risorse tra le regioni o di replicare i dati per la ridondanza geografica.

**Di seguito viene descritta la procedura per stabilire una connessione peering VPC:**

1. Il proprietario del VPC richiedente invia una richiesta al proprietario del VPC accettante per creare la connessione peering VPC. Il VPC accettante può essere di proprietà tua o di un altro account AWS e non può avere un blocco CIDR che si sovrappone al blocco CIDR del VPC richiedente.
2. Il proprietario del VPC accettante accetta la richiesta di connessione peering VPC per attivare tale connessione.
3. Per abilitare il flusso di traffico tra i VPC utilizzando gli indirizzi IP privati, il proprietario di ogni VPC nella connessione peering VPC deve aggiungere manualmente una route a una o più delle relative tabelle di routing VPC che punta all'intervallo di indirizzi IP dell'altro VPC (il VPC in peering).

4. Se necessario, aggiorna le regole di gruppo di sicurezza associate alla tua istanza per assicurarti che il traffico da e verso il VPC in peering non sia limitato. Se entrambi i VPC sono nella stessa regione, puoi fare riferimento a un gruppo di sicurezza del VPC in peering come origine o destinazione delle regole in ingresso o in uscita nelle regole di gruppo di sicurezza.
5. Con le opzioni di connessione in peering VPC di default, se le istanze EC2 su entrambi i lati di un indirizzo di connessione in peering VPC comunicano tra loro utilizzando un nome host DNS pubblico, il nome host si risolve nell'indirizzo IP pubblico dell'istanza. Per modificare questo comportamento, abilita la risoluzione del nome host DNS per la tua connessione VPC. Dopo l'abilitazione della risoluzione del nome host DNS, se le istanze su entrambi i lati di una connessione peering VPC comunicano tra loro utilizzando un nome host DNS pubblico, il nome host si risolve nell'indirizzo IP privato dell'istanza.

#### Ciclo di vita delle connessioni peering VPC:



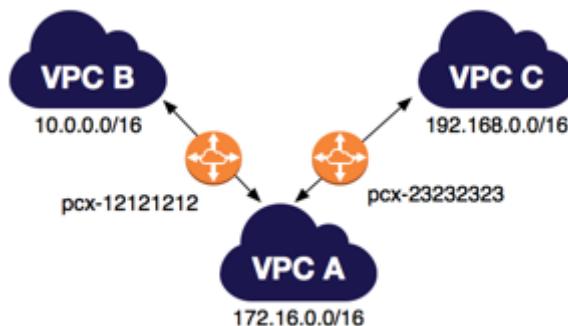
1. **Initiating-request:** una richiesta di connessione peering VPC è stata avviata. In questa fase, la connessione peering può non riuscire o passare allo stato *pending-acceptance*.
2. **Failed:** la richiesta di connessione peering VPC non è riuscita. Quando è in questo stato, non può essere accettata, rifiutata o eliminata. La connessione peering VPC non riuscita rimane visibile al richiedente per 2 ore.
3. **Pending-acceptance:** la richiesta di connessione peering VPC è in attesa di essere accettata dal proprietario del VPC accettante. Quando la richiesta è in questo stato, il proprietario del VPC richiedente può eliminarla e il proprietario del VPC accettante può accettarla o rifiutarla. Se non viene eseguita alcuna operazione, la richiesta scade dopo 7 giorni.
4. **Expired:** la richiesta di connessione peering VPC è scaduta e nessuna operazione può essere eseguita dai due proprietari di VPC. La connessione peering VPC scaduta rimane visibile a entrambi i proprietari per 2 giorni.
5. **Rejected:** il proprietario del VPC accettante ha rifiutato una richiesta di connessione peering VPC pending-acceptance. Durante tale stato, la richiesta non può essere accettata. La connessione peering VPC rifiutata rimane visibile al proprietario del VPC richiedente per 2 giorni e al proprietario del VPC accettante per 2 ore. Se la richiesta è stata creata nello stesso account AWS, la richiesta rifiutata rimarrà visibile per 2 ore.
6. **Provisioning:** la richiesta di connessione peering VPC è stata accettata e a breve il suo stato sarà *active*.

7. **Active:** la connessione peering VPC è attiva e il traffico può circolare tra i VPC (purché i gruppi di sicurezza e le tabelle di routing lo consentano). Durante questo stato, entrambi i proprietari di VPC possono eliminare la connessione peering VPC, ma non rifiutarla.
8. **Deleting:** si applica a una connessione peering VPC interregionale che sta per essere eliminata. Il proprietario di uno dei VPC ha inviato una richiesta di eliminazione di una connessione peering VPC **active** oppure il proprietario del VPC richiedente ha inviato una richiesta di eliminazione di una richiesta di connessione peering VPC **pending-acceptance**.
9. **Deleted:** una connessione peering VPC **active** è stata eliminata da uno dei proprietari, oppure una richiesta di connessione peering VPC **pending-acceptance** è stata eliminata dal proprietario del VPC richiedente. Durante questo stato la connessione peering VPC non può essere accettata o rifiutata. La connessione peering VPC rimane visibile al proprietario che l'ha eliminata per 2 ore e all'altro proprietario per 2 giorni. Se la connessione peering VPC è stata creata nello stesso account AWS, la richiesta eliminata rimarrà visibile per 2 ore.

### Molteplici connessioni peering VPC:

Una connessione peering VPC è una relazione uno a uno tra due VPC. Puoi creare molteplici connessioni peering VPC per ogni tuo VPC, ma le relazioni peering transitive non sono supportate. Non disponi di alcuna relazione peering con i VPC ai quali il tuo VPC non è direttamente collegato in peering.

Il diagramma seguente è un esempio di VPC collegato in peering a due differenti VPC. Si hanno due connessioni peering VPC: VPC A è collegato in peering a VPC B e VPC C. VPC B e VPC C non sono collegati in peering e non puoi utilizzare VPC A come punto di transito per il peering tra VPC B e VPC C. Se vuoi abilitare il routing del traffico tra VPC B e VPC C, devi creare una connessione peering VPC univoca tra gli stessi.



### Limitazioni (più importanti) relative al peering VPC, [1]:

- Non puoi creare una connessione peering VPC tra VPC che hanno blocchi CIDR IPv4 o IPv6 corrispondenti o sovrapposti.
- Hai un limite per il numero di connessioni peering VPC attive e in attesa per VPC.
- Il peering di VPC non supporta relazioni di peering transitive. In una connessione peering VPC, il tuo VPC non ha accesso ad altri VPC ai quali il VPC peer non può essere collegato in peering.
- Non puoi avere più di una connessione peering VPC tra due stessi VPC nello stesso momento.

## 5.6.1 Creazione di una connessione peering VPC a un altro VPC nel proprio account, [1]

Per richiedere una connessione peering VPC a un VPC nell'account, assicurati di disporre degli ID dei VPC con i quali stai creando la connessione peering VPC. Devi creare e accettare personalmente la richiesta di connessione peering VPC per attivarla.

Puoi creare una connessione peering VPC a un VPC nella stessa Regione o in una Regione differente.

**Importante:** Assicurati che i VPC non abbiano blocchi CIDR IPv4 che si sovrappongono. In caso contrario, lo stato della connessione peering VPC diventa immediatamente failed.

1. Nel riquadro di navigazione, scegliere **Peer Connections (Connessioni Peer)**, **Create Peering Connection (Crea connessione peering)**.

The screenshot shows the AWS VPC Peering Connections interface. On the left, a sidebar lists various VPC-related options, with 'Connessioni Peer' highlighted by a red box and the number '1'. The main content area is titled 'Connessioni peering' and contains a table with the following columns: 'Name', 'ID connessione peering', 'Stato', and 'VPC richiedente'. A search bar at the top of the table area is labeled 'Filtro connessioni peering'. A red box and the number '2' highlight the 'Crea connessione peering' button in the top right corner of the main area.

2. Nella scheda **Impostazioni di connessione peering** scegliere:

- a. **Nome** (facoltativo): è eventualmente possibile assegnare un nome alla connessione peering VPC.
- b. **ID VPC (Requester) (ID VPC (richiedente))**: selezionare il VPC nell'account con il quale si intende richiedere la connessione peering VPC.
- c. In **Select another VPC to peer with (Seleziona un altro VPC da collegare in peering)**, assicurarsi che **My account (Il mio account)** sia selezionato e selezionare un altro VPC.
- d. Nel selettore della **Regione**, selezionare la Regione del VPC accettante.

## Impostazioni di connessione peering

### Nome - facoltativo

Crea un tag con una chiave del 'nome' e un valore specificato.

 1

Selezione un VPC locale da collegare in peering con

ID VPC (richiedente)

 2

Selezione un altro VPC da collegare in peering con

Account

- Il mio account  
 Un altro account

3

Regione

- Questa regione (us-east-1)  
 Un'altra regione

ID VPC (accettatore)

 4

3. (Facoltativo) Nella scheda **Tag** premi **Aggiungi nuovo tag** e procedere come segue:

a. In **Key (Chiave)**, immettere il nome della chiave.

b. (facoltativo) In **Value (Valore)**, immettere il valore della chiave.

Per rimuovere un tag selezionare il pulsante **Rimuovi**, alla destra della chiave e del valore del tag.

4. Premi infine su **Crea connessione peering**.

## Tag

Un tag è un'etichetta assegnata a una risorsa AWS. Ogni tag è composto da una chiave e da un valore facoltativo. È possibile usare i tag per ricercare e filtrare le risorse o monitorare i tuoi costi di AWS.

Chiave

Valore - facoltativo

Rimuovi

⚠ È necessario specificare una chiave di tag.

**Aggiungi nuovo tag**

È possibile aggiungere 49 altri tag.

Annulla

**Crea connessione peering**

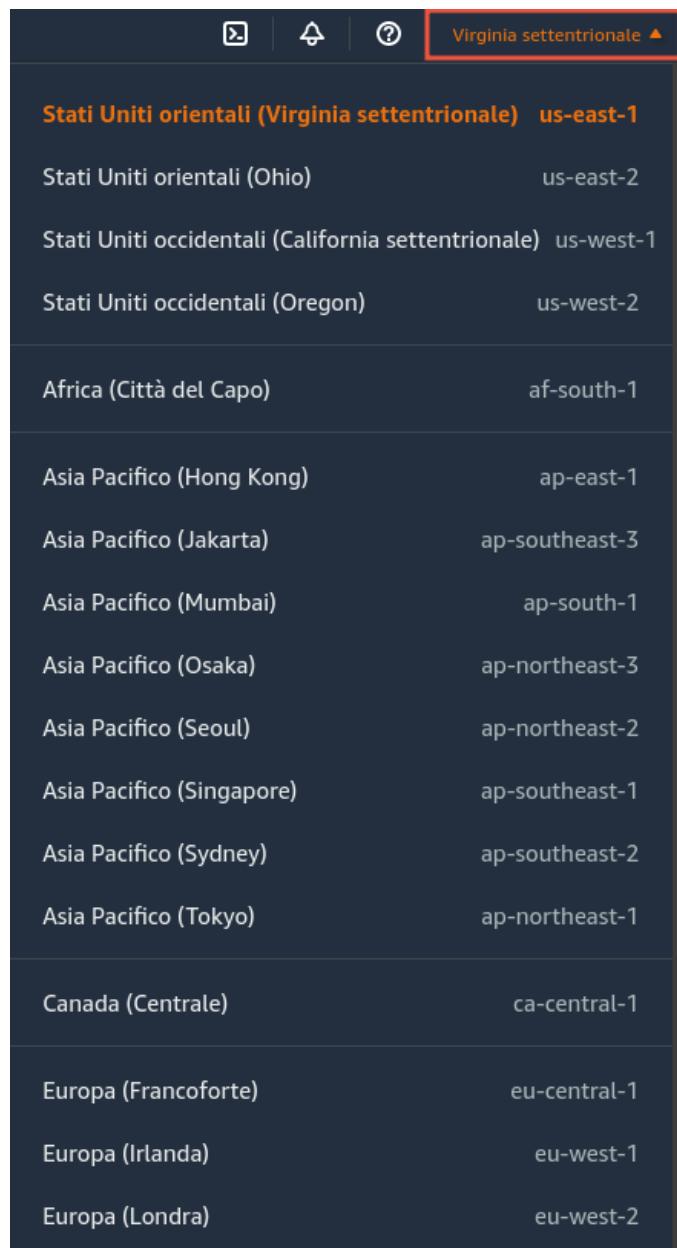
5. Nella finestra di dialogo di conferma, scegliere **OK**.

6. Selezionare la connessione peering VPC creata e scegliere **Actions (Operazioni)**, **Accept Request (Accetta richiesta)**.

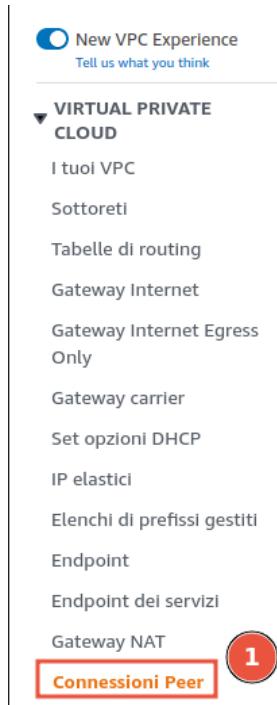
7. Nella finestra di dialogo di conferma, scegliere **Yes, Accept (Sì, accetta)**. Viene visualizzata una seconda finestra di dialogo di conferma; scegliere **Modify my route tables now (Modifica le tabelle di routing ora)** per accedere direttamente alla pagina delle tabelle di routing oppure scegliere **Close (Chiudi)** per farlo in seguito.

**Per accettare una connessione peering VPC:**

1. Utilizzare il selettore della regione per scegliere la regione del VPC accettante.



2. Nel riquadro di navigazione, scegliere **Peer Connections (Conessioni peer)**,



- Selezionare la connessione peering VPC in attesa (lo stato è *pending-acceptance*) e scegliere **Actions (Operazioni)**, **Accept Request (Accetta richiesta)**.

**Nota:** Se la connessione peering VPC in attesa non è visibile, controllare la Regione. Una richiesta di peering interregionale deve essere accettata nella Regione del VPC accettante.

- Nella finestra di dialogo di conferma, scegliere **Yes, Accept (Sì, accetta)**. Viene visualizzata una seconda finestra di dialogo di conferma; scegliere **Modify my route tables now (Modifica le tabelle di routing ora)** per accedere direttamente alla pagina delle tabelle di routing oppure scegliere **Close (Chiudi)** per farlo in seguito.

A questo punto, la connessione peering VPC è attiva e devi quindi aggiungere una voce alla tabella di routing VPC per consentire l'indirizzamento del traffico al VPC in peering.

### 5.6.2 Aggiornamento delle tabelle di routing per una connessione peering VPC, [1]

- Nel riquadro di navigazione, selezionare **Route Tables (Tabelle di routing)**. Seleziona la casella di controllo accanto alla tabella di routing associata alla sottorete in cui si trova la tua istanza. Selezionare **Actions (Operazioni)** → **Edit routes (Modifica route)**.

- Selezionare **Add route (Aggiungi route)**.

- Nel primo campo **Destination (Destinazione)**, immettere l'intervallo di indirizzi IPv4 verso il quale il traffico di rete nella connessione peering VPC deve essere diretto. È possibile specificare l'intero blocco CIDR IPv4 del VPC in peering, uno specifico intervallo o un singolo indirizzo IPv4, come l'indirizzo IP dell'istanza con la quale comunicare. Ad esempio, se il blocco CIDR del VPC in peering è **10.0.0.0/16**, è possibile specificare una parte **10.0.0.0/24** o uno specifico indirizzo IP **10.0.0.7/32**.
- Nel secondo campo **Destination (Destinazione)** seleziona la connessione peering VPC, quindi seleziona **Salva modifiche**.

## Modifica route

Destinazione	Destinazione	Stato	Propagata
10.5.0.0/16	<input type="text" value="Q local"/> <input type="button" value="X"/>	<input checked="" type="checkbox"/> Attivo	No
<input type="text" value="Q 0.0.0.0"/> <input type="button" value="X"/>	<input type="text" value="Q igw- [REDACTED]"/> <input type="button" value="X"/>	<input checked="" type="checkbox"/> Attivo	No <input type="button" value="Rimuovi"/>
<input type="text" value="Q 0.0.0.0/16"/> <input type="button" value="X"/>	<input type="text" value="Q"/> <input type="button" value="X"/>	-	No <input type="button" value="Rimuovi"/>
<input type="button" value="Aggiungi route"/>			
		<input type="button" value="Annulla"/>	<input type="button" value="Anteprima"/> <input type="button" value="Salva modifiche"/>

## 5.7 Gateway Internet, [1]

Un gateway Internet è un componente VPC scalato orizzontalmente, ridondante e ad alta disponibilità che consente la comunicazione tra il VPC e Internet. Un gateway Internet consente alle risorse (come le istanze EC2) nelle sottoreti pubbliche di connettersi a Internet se la risorsa dispone di un indirizzo IPv4 pubblico o un indirizzo IPv6. Analogamente, le risorse su Internet possono avviare una connessione alle risorse nella sottorete utilizzando l'indirizzo IPv4 pubblico o IPv6 pubblico. Ad esempio, un gateway Internet consente di connettersi a un'istanza EC2 in AWS utilizzando il computer locale.

Un Internet Gateway svolge due funzioni: fornisce un target nelle tabelle di routing VPC per il traffico instradabile su Internet ed esegue la conversione Network Address Translation (NAT) per le istanze a cui sono stati assegnati indirizzi IPv4 pubblici.

Un Internet Gateway supporta il traffico IPv4 e IPv6. Non causa rischi di disponibilità o vincoli di larghezza di banda nel traffico di rete. Non ci sono costi aggiuntivi per avere un gateway internet nell'account AWS.

### Abilitazione dell'accesso a Internet:

Per abilitare l'accesso a o da Internet per le istanze in una sottorete in un VPC, devi completare le operazioni riportate di seguito.

- Creare un gateway Internet e collegarlo al VPC;
- Aggiungere una route alla tabella di routing della sottorete che indirizza il traffico Internet all'Internet Gateway;
- Verificare che le istanze in una sottorete abbiano un indirizzo IP globalmente univoco (indirizzo IPv4 pubblico, indirizzo IP elastico o indirizzo IPv6);
- Verificare che le regole di gruppo di sicurezza e le liste di controllo degli accessi di rete consentano il flusso di traffico pertinente verso e dall'istanza.

### 5.7.1 Creazione e collegamento di un gateway Internet, [1]

- Accedere alla console Amazon VPC all'indirizzo <https://console.aws.amazon.com/vpc/>

2. Nel riquadro di navigazione, scegliere **Internet Gateways (Internet Gateway)**, quindi selezionare **Create internet gateway (Crea Internet Gateway)**.

The screenshot shows the AWS VPC Gateway Internet page. On the left, there's a sidebar with a red box around 'Gateway Internet' under 'Cloud privato virtuale'. A red arrow labeled '1' points to this. At the top right, there's a red box around the 'Crea gateway Internet' button, with a red arrow labeled '2' pointing to it. The main area shows a table with two rows of gateway entries, both marked as 'Attached'.

3. Facoltativamente è possibile assegnare un nome al gateway Internet.  
 4. (Facoltativo) Aggiungere o rimuovere un tag.  
     a. [Aggiungere un tag] Scegliere **Add tag (Aggiungi tag)** e procedere come segue:  
         b. In **Key (Chiave)**, immettere il nome della chiave.  
         c. In **Value (Valore)**, immettere il valore della chiave.  
             [Rimuovi un tag] Scegli **Rimuovi** a destra della Chiave e del Valore del tag.  
 5. Scegliere **Crea gateway Internet**.

This screenshot shows the 'Impostazioni gateway Internet' (Gateway Internet Settings) page. It has two main sections: 'Tag del nome' and 'Tag - facoltativo'.  
 In the 'Tag del nome' section, there's a text input field containing 'my-Internet-gateway' with a red box and arrow labeled '1' pointing to it.  
 In the 'Tag - facoltativo' section, there are fields for 'Chiave' and 'Valore - facoltativo', both with red boxes and arrows labeled '2' and '3' pointing to them. To the right is a 'Rimuovi' (Remove) button with a red box and arrow labeled '4' pointing to it.  
 At the bottom right are 'Annulla' (Cancel) and 'Crea gateway Internet' (Create gateway Internet) buttons, with the latter being orange.

6. Selezionare l'Internet Gateway appena creato, quindi selezionare **Actions → Attach to VPC (Operazioni → Collega al VPC)**.

**Gateway Internet (1/3) Informazioni**

**Operazioni ▲**

**Crea gateway Internet**

**Visualizza i dettagli**

**Collega al VPC** (highlighted)

**Scollega dal VPC**

**Gestisci tag**

**Elimina gateway Internet**

**Name** **ID gateway Internet**

**Attached**

Name	ID gateway Internet	Attached
gateway-dlTest	igw-1	Attached
-	igw-	-

7. Selezionare il VPC dall'elenco, quindi selezionare **Collega gateway Internet**.

### Collega al VPC (igw-██████████) Informazioni

**VPC**

Collega un gateway Internet a un VPC per abilitare la comunicazione con Internet. Specifica il VPC da collegare di seguito.

**VPC disponibili**

Collega il gateway Internet a questo VPC.

**Seleziona un VPC** (highlighted)

▶ Comando dell'interfaccia a riga di comando di AWS

**Annula** **Collega gateway Internet** (highlighted)

#### 5.7.2 Creazione di una tabella di routing personalizzata, [1]

Quando crei una sottorete, l'associamo automaticamente alla tabella di routing principale per il VPC. Per impostazione predefinita, la tabella di routing principale non contiene una route all'Internet Gateway. La procedura seguente crea una tabella di routing personalizzata con una route che invia il traffico destinato all'esterno del VPC all'Internet Gateway e quindi lo associa alla sottorete. Per creare una tabella di routing personalizzata:

1. Accedere alla console Amazon VPC all'indirizzo <https://console.aws.amazon.com/vpc/>
2. Nel riquadro di navigazione, scegli **Tabelle di routing**, quindi seleziona **Crea tabella di routing**.

Pannello di controllo VPC  
Visualizzazione globale di EC2 Novità  
Filtro per VPC:  
Seleziona un VPC  
Cloud privato virtuale  
I tuoi VPC  
Sottoreti  
**Tabelle di routing** 1  
Gateway Internet

**Tabelle di routing (4)** Informazioni

Copia | Operazioni ▾ Crea tabella di routing

<input type="checkbox"/>	Name	ID tabella di routing	Splicita associazioni s...	Associazioni ed...
<input type="checkbox"/>	-	rtb-[REDACTED]	-	-
<input type="checkbox"/>	-	rtb-[REDACTED]	-	-
<input type="checkbox"/>	-	rtb-[REDACTED]	-	-
<input type="checkbox"/>	-	rtb-[REDACTED]	-	-

3. Nella finestra di dialogo **Crea tabella di routing**, assegna facoltativamente un **Nome** alla tabella di routing, quindi seleziona il **VPC** e scegli **Crea tabella di routing**.

### Impostazioni della tabella di instradamento

**Nome - facoltativo**  
Crea un tag con una chiave del 'nome' e un valore specificato.

1

**VPC**  
Il VPC da utilizzare per questa tabella di instradamento.

2

### Tag

Un tag è un'etichetta assegnata a una risorsa AWS. Ogni tag è composto da una chiave e da un valore facoltativo. È possibile usare i tag per ricercare e filtrare le risorse o monitorare i tuoi costi di AWS.

<b>Chiave</b>	<b>Valore - facoltativo</b>
<input style="border: 2px solid red; width: 150px; height: 30px; margin-bottom: 10px;" type="text" value="Name"/> <span style="color: red; font-size: 2em; position: absolute; left: 50%; top: 50%;">3</span>	<input style="border: 2px solid red; width: 150px; height: 30px; margin-bottom: 10px;" type="text" value="route_tableTest"/> <span style="color: red; font-size: 2em; position: absolute; left: 50%; top: 50%;">4</span>
<b>Rimuovi</b>	
<b>Aggiungi nuovo tag</b>	
È possibile aggiungere 49 altri tag.	

**Annula** **Crea tabella di routing**

4. Selezionare la tabella di routing personalizzata appena creata. Nel riquadro dei **Dettagli** sono visualizzate le schede per utilizzare la route, le associazioni e la propagazione della route.

**Tabelle di routing (1/5) [Informazioni](#)**

[Crea tabella di routing](#)

Name	ID tabella di routing	Esplicita associazioni s...	Associazioni edge	Prin...
<input checked="" type="checkbox"/> route_tableTest	rtb-[REDACTED]	-	-	No
<input type="checkbox"/> -	rtb-[REDACTED]	-	-	Sì
<input type="checkbox"/> -	rtb-[REDACTED]	-	-	Sì

rtb-[REDACTED] / route\_tableTest

[Dettagli](#) **Route** [Associazioni sottorete](#) [Associazioni edge](#) [Propagazione dell'instradamento](#) [Tag](#)

**Route (1)** 2

[Filtro instradamenti](#) 3 [Modifica route](#)

Destinazione	Destinazione	Stato	Propagata
172.31.0.0/16	local	<input checked="" type="checkbox"/> Attivo	No

5. Nella scheda **Route**, seleziona **Modifica route**, Aggiungi un'altra route e aggiungi le seguenti route come necessario. Al termine, scegli **Salva modifiche**.

- Per il traffico IPv4, specificare 0.0.0.0/0 nella casella **Destinazione (Destinazione)** e selezionare l'ID dell'Internet Gateway nell'elenco **Target (Destinazione)**.
- Per il traffico IPv6, specificare ::/0 nella casella **Destinazione (Destinazione)** e selezionare l'ID dell'Internet Gateway nell'elenco **Target**.

**Modifica route**

Destinazione	Destinazione	Stato
172.31.0.0/16	<input type="text" value="local"/> <input type="button" value="X"/>	<input checked="" type="checkbox"/> Attivo
Propagata		
No		

---

**Modifica route**

Destinazione	Destinazione	Stato
<input type="text" value="0.0.0.0/0"/> <input type="button" value="X"/>	<input type="text" value="igw-813136e8"/> <input type="button" value="X"/>	-
Propagata		
No		

1

Salva modifiche 3

6. Nella scheda **Associazioni sottorete**, scegli **Modifica associazioni sottorete**,

Tabelle di routing (1/5) [Informazioni](#)

[C](#) Operazioni ▾ [Crea tabella di routing](#)

Filtra tabelle di routing

Name	ID tabella di routing	Esplicita associazioni s...	Associazioni edge	Prin...
<input checked="" type="checkbox"/> route_tableTest	rtb-[REDACTED]	-	-	No
<input type="checkbox"/> -	rtb-[REDACTED]	-	-	Sì
<input type="checkbox"/> -	rtb-[REDACTED]	-	-	Sì
<input type="checkbox"/> -	rtb-[REDACTED]	-	-	Sì

rtb-[REDACTED] / route\_tableTest

Dettagli Route **Associazioni sottorete** Associazioni edge Propagazione dell'instradamento Tag

**Esplicita associazioni sottorete (0)** **Modifica associazioni sottorete**

Trova associazione sottorete

ID sottorete CIDR IPv4

Nessuna associazione di sottoreti  
Non disponi di associazioni di sottorete.

seleziona la casella di controllo per la sottorete e seleziona **Salva associazioni**.

Sottoreti disponibili (1/3)

Filtra associazioni sottorete

Nome	ID sottorete	CIDR IPv4	CIDR IPv6	ID tabella di routing
<input checked="" type="checkbox"/> subnet-[REDACTED]	subnet-[REDACTED]	172.31.0.0/20	-	Principale (rtb-[REDACTED])
<input type="checkbox"/> subnet-[REDACTED]	subnet-[REDACTED]	172.31.16.0/20	-	Principale (rtb-[REDACTED])
<input type="checkbox"/> subnet-[REDACTED]	subnet-[REDACTED]	172.31.32.0/20	-	Principale (rtb-[REDACTED])

Sottoreti selezionate

subnet-[REDACTED] X

2

Annula **Salva associazioni**

### 5.7.3 Creazione di un gruppo di sicurezza per l'accesso a Internet, [1]

Per impostazione predefinita, un gruppo di sicurezza VPC abilita tutto il traffico in uscita. È possibile creare un nuovo gruppo di sicurezza e aggiungere regole che abilitano il traffico in entrata da Internet. È quindi possibile associare il gruppo di sicurezza alle istanze nella sottorete pubblica. Per creare un gruppo di sicurezza e associarlo alle istanze:

1. Apri la console Amazon EC2 all'indirizzo <https://console.aws.amazon.com/ec2/>
2. Nel pannello di navigazione, scegli **Gruppi di sicurezza**, quindi **Crea gruppo di sicurezza**.

Pannello di controllo EC2

Visualizzazione di EC2

Global

Eventi

Tag

Limiti

Istanze

Immagini

Elastic Block Store

Rete e sicurezza

**Gruppi di sicurezza**

IP elasticici

Gruppi di collocamento

**Gruppi di sicurezza (5) Informazioni**

C Operazioni Esporta gruppi di sicurezza in CSV ▾ Crea gruppo di sicurezza

Filtra i gruppi di sicurezza

<input type="checkbox"/>	Name	ID gruppo di sicurezza	Nome del gruppo d...	ID VPC
<input type="checkbox"/>	-	sg-[REDACTED]	launch-wizard-1	vpc-[REDACTED]
<input type="checkbox"/>	-	sg-[REDACTED]	default	vpc-[REDACTED]
<input type="checkbox"/>	-	sg-[REDACTED]	default	vpc-[REDACTED]
<input type="checkbox"/>	-	sg-[REDACTED]	default	vpc-[REDACTED]
<input type="checkbox"/>	-	sg-[REDACTED]	default	vpc-[REDACTED]

3. Immettere un **Nome** e una **Descrizione** per il gruppo di sicurezza.
4. In **VPC**, seleziona il tuo VPC.

### Dettagli di base

Nome del gruppo di sicurezza [Informazioni](#)

Il nome non può essere modificato a creazione ultimata.

Descrizione [Informazioni](#)

VPC [Informazioni](#)

5. In **Regole in entrata**, seleziona **Aggiungi regola**, quindi completa le informazioni richieste. Ad esempio, seleziona **HTTP** o **HTTPS** dall'elenco **Type (Tipo)** e inserisci in **Source (Origine)** `0.0.0.0/0` per il traffico IPv4 oppure `::/0` per il traffico IPv6.

**Regole in entrata** [Informazioni](#)

**Regola in entrata 1**

Tipo	Informazioni	Protocollo	Informazioni	Intervallo porte	Informazioni
HTTPS	TCP	443			
Tipo di origine	Informazioni	Origine	Informazioni	Descrizione - facoltativa	Informazioni
Anywhere-IPv4		0.0.0.0/0	X		

[Aggiungi regola](#)

6. Scegliere **Create Security Group (Crea gruppo di sicurezza)**.
7. Nel riquadro di navigazione, seleziona **Instances (Istanze)**.
8. Seleziona l'istanza, quindi scegli **Operazioni → Sicurezza → Modifica i gruppi di sicurezza**.

**Istanze (1/6) Informazioni**

**Operazioni** ▲ **Avvia le istanze** ▼

Name	ID istanza	Stato dell'istanza	Tipo di istanza
[checkbox]	[ID]	Arrestato	t3.small
[checkbox]	[ID]	Arrestato	t3.medium
[checkbox]	[ID]	Arrestato	t3.micro
[checkbox]	[ID]	Arrestato	t3.small
[checkbox]	[ID]	Arrestato	t3.medium
[checkbox]	[ID]	Arrestato	t3.micro

**Sicurezza**

Connetti Visualizza i dettagli Gestisci lo stato dell'istanza Impostazioni istanza Reti Immagine e modelli Monitoraggio e risoluzione dei problemi

9. Per **Gruppi di sicurezza associati**, seleziona un gruppo di sicurezza esistente e scegli **Aggiungi gruppo di sicurezza**. Per rimuovere un gruppo di sicurezza già associato, scegli **Rimuovi**. Una volta completate le modifiche, scegli **Salva**.

## Dettagli istanza

The screenshot shows the 'Dettagli istanza' (Instance Details) page in the AWS VPC console. At the top, there are fields for 'ID istanza' and 'ID Interfaccia di rete'. Below this, a section titled 'Gruppi di sicurezza selezionati' (Selected security groups) allows you to add or remove security groups from the network interface. A search bar contains the prefix 'sg-' (1). To the right is a blue button labeled 'Aggiungi gruppo di sicurezza' (Add security group). Further down, a list shows 'Gruppi di sicurezza associati all'interfaccia di rete (eni-...)' (Security groups associated with the network interface). It lists a single entry: 'Nome del gruppo di sicurezza' (sg-) and 'ID gruppo di sicurezza' (sg-...). To the right of this entry are a 'Rimuovi' (Remove) button and an orange 'Salva' (Save) button. Red numbers 2 and 3 are overlaid on the 'Aggiungi' button and the 'Salva' button respectively.

### 5.7.4 Assegnazione di un indirizzo IP elastico a un'istanza, [1]

Dopo l'avvio di un'istanza nella sottorete, devi assegnarle un indirizzo IP elastico affinché sia accessibile da Internet via IPv4.

**Nota:**

Se hai assegnato un indirizzo IPv4 pubblico all'istanza durante l'avvio, l'istanza è accessibile da Internet e non devi assegnarle un indirizzo IP elastico.

**Per allocare un indirizzo IP elastico e assegnarlo a un'istanza tramite la console:**

1. Accedere alla console Amazon VPC all'indirizzo <https://console.aws.amazon.com/vpc/>
2. Nel riquadro di navigazione, selezionare **Elastic IPs (IP elastici)**.
3. Scegliere **Allocate new address (Alloca nuovo indirizzo)**.
4. Selezionare **Alloca**.

**Nota:**

Se l'account supporta EC2-Classic, in primo luogo scegliere VPC.

5. Selezionare l'indirizzo IP elastico dall'elenco, scegliere **Actions (Operazioni)**, quindi selezionare **Associate Address (Associa indirizzo)**.
6. Selezionare **Instance (Istanza)** o **Network interface (Interfaccia di rete)**, quindi selezionare l'ID dell'istanza o dell'interfaccia di rete. Selezionare l'indirizzo IP privato con cui associare l'indirizzo IP elastico, quindi selezionare **Associate (Associa)**.

### 5.7.5 Scollegamento di un gateway Internet dal VPC, [1]

Se non hai più bisogno dell'accesso a Internet per le istanze che avvii in un VPC non predefinito, puoi scollegare un Internet Gateway da un VPC. Non puoi scollegare un Internet Gateway se il VPC ha risorse con indirizzi IP pubblici o indirizzi IP elasticici associati. Per scollegare un Internet Gateway:

1. Accedere alla console Amazon VPC all'indirizzo <https://console.aws.amazon.com/vpc/>
2. Nel riquadro di navigazione, selezionare **Elastic IPs (IP elasticici)**, quindi selezionare l'indirizzo IP elastico.
3. Selezionare Actions (Operazioni), scegliere **Disassociate address (Disassocia indirizzo)**.

The screenshot shows the 'Indirizzi IP elasticici' (Elastic IPs) page. A specific IP address (18.102.7.19) is selected. A context menu is open over this row, with step 3 pointing to the 'Disassocia indirizzo IP elastico' (Disassociate address) option. Step 4 points to the 'Aggiorna DNS Inverso' (Update Reverse DNS) button at the bottom of the menu.

4. Nel riquadro di navigazione, scegliere **Internet Gateways**.
5. Selezionare l'Internet Gateway e scegliere **Actions → Detach from VPC (Operazioni → Scollega dal VPC)**.

The screenshot shows the 'Gateway Internet' (Internet Gateways) page. An Internet Gateway named 'gateway-diTest' is selected. A context menu is open over this row, with step 3 pointing to the 'Scollega dal VPC' (Detach from VPC) option. Step 4 points to the 'Elimina gateway Internet' (Delete Internet Gateway) button at the bottom of the menu.

6. Nella finestra di dialogo Scollega dal VPC scegliere **Scollega gateway Internet**.

### 5.7.6 Eliminazione di un Internet Gateway, [1]

Se non hai più bisogno di un Internet Gateway, puoi eliminarlo. Non puoi tuttavia svolgere questa operazione se un Internet Gateway è ancora collegato a un VPC. Per eliminare un Internet Gateway:

1. Accedere alla console Amazon VPC all'indirizzo <https://console.aws.amazon.com/vpc/>
2. Nel riquadro di navigazione, scegliere **Gateway Internet**.
3. Selezionare l'Internet Gateway e scegliere **Actions (Operazioni) → Delete Internet Gateway (Elimina Internet Gateway)**.

Gateway Internet (1/3) <a href="#">Informazioni</a>			
	Name	ID gateway Internet	Stato
<input type="checkbox"/>	igw-[REDACTED]	igw-[REDACTED]	Attached
<input checked="" type="checkbox"/>	gateway-diTest	igw-[REDACTED]	Detached
<input type="checkbox"/>	-	igw-[REDACTED]	Attached

4. Nella finestra di dialogo **Elimina gateway Internet** immettere **elimina**, e scegliere **Elimina gateway Internet**.



## 5.8 Tabelle di routing, [1]

Una tabella di routing contiene un insieme di regole, denominato route, che consente di determinare la direzione del traffico di rete dalla sottorete o dal gateway.

### Route:

Ogni route in una tabella specifica una destinazione e un target. Ad esempio, per consentire alla sottorete di accedere a Internet tramite un Internet gateway, aggiungi la seguente route alla tabella di routing della sottorete. La destinazione per la route è `0.0.0.0/0`, che rappresenta tutti gli indirizzi IPv4. Il target è l'Internet gateway collegato al VPC.

Destinazione	Target
<code>0.0.0.0/0</code>	<code>igw-id</code>

I blocchi CIDR per IPv4 e IPv6 sono trattati separatamente. Ad esempio, una route con un CIDR di destinazione `0.0.0.0/0` non include automaticamente tutti gli indirizzi IPv6. Devi creare una route con un CIDR di destinazione `::/0` per tutti gli indirizzi IPv6.

Ogni tabella di routing contiene una route locale per la comunicazione all'interno del VPC.

Questa route viene aggiunta per impostazione predefinita a tutte le tabelle di routing.

Le seguenti regole si applicano alla tabella di routing principale:

- Non puoi eliminare la tabella di routing principale.

- Non puoi impostare una tabella di routing del gateway come tabella di routing principale.
- È possibile sostituire la tabella di routing principale con una tabella di routing della sottorete personalizzata.
- Puoi aggiungere, rimuovere e modificare le route nella tabella di routing principale.
- Puoi associare in modo esplicito una sottorete alla tabella di routing principale, anche se è già implicitamente associata.

Questa operazione può essere utile quando cambi la tabella di routing principale. In questo caso, viene modificata anche la tabella predefinita per le nuove sottoreti o per qualsiasi sottorete non esplicitamente associata ad altre tabelle di routing. Per ulteriori informazioni, consulta Sostituzione della tabella di routing principale.

### **Tabelle di routing personalizzate:**

Per impostazione predefinita, una tabella di routing personalizzata è vuota e puoi aggiungerci route in base alle esigenze. Se crei un VPC, sottoreti e altre risorse VPC e scegli una sottorete pubblica, Amazon VPC crea una tabella di instradamento personalizzata e aggiunge una route che punta al gateway Internet. Un modo per proteggere il VPC è lasciare la tabella di routing principale nel suo stato predefinito originale. Quindi, associare esplicitamente tutte le nuove sottoreti a una delle tabelle di routing personalizzate che hai creato. Ciò consente di controllare esplicitamente il modo in cui ogni sottorete instrada il traffico.

Puoi aggiungere, rimuovere e modificare le route in una tabella di routing personalizzata. Puoi eliminare una tabella di routing personalizzata solo se non ha associazioni.

### **Associazione di tabelle di routing della sottorete:**

Ogni sottorete nel VPC deve essere associata a una tabella di routing. Una sottorete può essere associata esplicitamente alla tabella di routing personalizzata oppure, implicitamente o esplicitamente, alla tabella di routing principale.

#### **5.8.1 Tabelle di routing del gateway, [1]**

Puoi associare una tabella di routing a un Internet gateway o a un gateway virtuale privato. Quando una tabella di routing è associata a un gateway, viene chiamata tabella di routing del gateway. Puoi creare una tabella di routing del gateway per controllare dettagliatamente il percorso di routing del traffico che entra nel VPC. Ad esempio, puoi intercettare il traffico che entra nel VPC tramite un Internet gateway reindirizzandolo a un'appliance middlebox (come un'appliance di sicurezza) nel VPC. [\[Regole e considerazioni\]](#)

#### **5.8.1 Priorità della route, [1]**

In generale, indirizziamo il traffico utilizzando il routing più specifico che corrisponde al traffico stesso. Ciò è noto come **corrispondenza prefisso più lungo**. Se la tabella di routing presenta routing sovrapposti o corrispondenti, si applicano le seguenti regole aggiuntive.

### **Corrispondenza prefisso più lungo:**

Le route verso indirizzi IPv4 e IPv6 o blocchi CIDR sono indipendenti l'uno dall'altro. Per determinare come instradare il traffico, viene usato il routing più specifico che corrisponde al traffico IPv4 o IPv6.

Ad esempio, la tabella di routing della sottorete seguente include una route per il traffico Internet IPv4 (0.0.0.0/0) che punta a un Gateway Internet e una route per il traffico IPv4 172.31.0.0/16 che punta a una connessione di peering (pcx-11223344556677889). Il traffico dalla sottorete destinato all'intervallo di indirizzi IP 172.31.0.0/16 utilizza la connessione di peering perché questa route è più specifica rispetto a quella per l'Internet gateway. Il traffico destinato a un

target nel VPC (10.0.0.0/16) è coperto dalla route local ed è quindi instradato all'interno del VPC. Il resto del traffico dalla sottorete utilizza l'Internet gateway.

Destinazione	Target
10.0.0.0/16	locale
172.31.0.0/16	pcx-11223344556677889
0.0.0.0/0	igw-12345678901234567

### Priorità del routing e routing propagati:

Se hai collegato un gateway virtuale privato al VPC e abilitato la propagazione delle route sulla tabella di routing della sottorete, le route che rappresentano la connessione Site-to-Site VPN vengono automaticamente visualizzate come route propagate nella tua tabella di routing.

Se la destinazione di un routing propagato si sovrappone a un route statico, il secondo ha la priorità.

Se la destinazione di un percrouting orso propagato è identica alla destinazione di un routing statico, quello statico ha la priorità se la destinazione è una delle seguenti:

- gateway Internet
- Gateway NAT
- Interfaccia di rete
- ID istanza
- Endpoint VPC del gateway
- Gateway di transito
- Connessione di peering di VPC
- Endpoint Gateway Load Balancer

Ad esempio, la seguente tabella di routing dispone di un routing statico a un Gateway Internet e un routing propagato a un gateway virtuale privato. La destinazione di entrambe le regole è 172.31.0.0/24. Poiché il routing statico verso un Gateway Internet ha la priorità, tutto il traffico destinato a 172.31.0.0/24 viene indirizzato al Gateway Internet.

Destinazione	Target	Propagato
10.0.0.0/16	locale	No
172.31.0.0/24	vgw-11223344556677889	Sì
172.31.0.0/24	igw-12345678901234567	No

## 6. CLI (AWS Command Line Interface), [1]

AWS Command Line Interface (AWS CLI) è uno strumento [open source](#) che consente di interagire con i servizi AWS utilizzando i comandi nella shell a riga di comando. Con una configurazione minima, AWS CLI consente di iniziare a eseguire i comandi che implementano funzionalità equivalenti a quelle fornite dalla console web di AWS.

**Digita i seguenti comandi per abilitare l'autocompletamento di AWS CLI (Linux, Mac), [1]:**

```
[per zsh]
$ autoload bashcompinit && bashcompinit
$ autoload -Uz compinit && compinit
$ complete -C '/usr/local/bin/aws_completer' aws

[per bash]
$ complete -C '/usr/local/bin/aws_completer' aws

[per tcsh]
> complete aws 'p/*/`aws_completer`/'
```

## 6.1 Configurazione preliminare, [1]

### Fase 1: Creazione di un account utente IAM Administrator

Per creare un utente amministratore per se stessi e aggiungere l'utente a un gruppo di amministratori (console):

1. Accedi alla console IAM come proprietario dell'account scegliendo **Utente root** e inserendo l'indirizzo e-mail di Account AWS. Nella pagina successiva, inserisci la password.  
**Nota:** È vivamente consigliato di rispettare la best practice sull'utilizzo dell'utente IAM **Administrator** e conservare le credenziali dell'utente root in un luogo sicuro. Accedi come utente root solo per eseguire alcune attività di gestione dell'account e del servizio.
2. Nel pannello di navigazione seleziona **Utenti**, quindi seleziona **Aggiungi utenti**.



3. In **Nome utente**, inserisci **Administrator**.
4. Seleziona la casella di controllo accanto a **Password - Accesso alla Console di gestione AWS**. Quindi scegli **Password personalizzata** e inserisci la nuova password nella casella di testo.
5. (Facoltativo) Di default, AWS richiede al nuovo utente di creare una nuova password al primo accesso. Puoi disattivare la casella di controllo accanto a **User must create a new password at next sign-in (L'utente deve creare una nuova password al prossimo accesso)** per consentire al nuovo utente di reimpostare la propria password dopo aver effettuato l'accesso.
6. Seleziona **Successivo: Autorizzazioni**.

## Aggiungi utente

1 2 3 4 5

### Imposta dettagli dell'utente

Puoi aggiungere più utenti contemporaneamente con lo stesso tipo di accesso e autorizzazioni. [Ulteriori informazioni](#)

Nome utente\*  1

[+ Aggiungi un altro utente](#)

### Seleziona il tipo di accesso AWS

Seleziona il modo in cui questi utenti accederanno principalmente ad AWS. Se scegli solo l'accesso programmatico, ciò NON impedirà agli utenti di accedere alla console utilizzando un ruolo assunto. Le chiavi di accesso e le password generate automaticamente vengono fornite nell'ultima fase. [Ulteriori informazioni](#)

Selezione il tipo di credenziali  Chiave di accesso - Accesso programmatico  
AWS\* Abilita una ID chiave di accesso e una chiave di accesso segreta per le API di AWS, l'interfaccia a riga di comando, SDK e altri strumenti di sviluppo.

Password - Accesso alla Console di gestione AWS  
Abilita una password che consente agli utenti di effettuare l'accesso alla console di gestione AWS. 2

Password console\*  Password autogenerata  Password personalizzata 3

\*\*\*\*\*  
 Mostra la password

Richiesta reimpostazione della password  L'utente deve creare una nuova password al prossimo accesso  
Gli utenti ottengono automaticamente la policy [IAMUserChangePassword](#) per consentire loro di modificare la propria password. 4

\* Campo obbligatorio

Annulla

Successivo: Autorizzazioni 5

7. In **Imposta autorizzazioni**, seleziona **Add user to group (Aggiungi l'utente al gruppo)**.
8. Seleziona **Crea gruppo**.

## Aggiungi utente

1 2 3 4 5

### Imposta autorizzazioni

1

Aggiungi un utente a un gruppo esistente o creane uno nuovo. L'utilizzo dei gruppi costituisce una procedura consigliata per gestire le autorizzazioni dell'utente in base alle funzioni lavorative. [Ulteriori informazioni](#)

### Add user to group (Aggiungi utente al gruppo)

2

9. Nella finestra di dialogo **Crea gruppo**, per **Nome gruppo** inserisci **Administrators**.
10. Scegli **Filtra policy**, per filtrare i contenuti della tabella.
11. Nell'elenco delle policy, seleziona la casella di controllo accanto ad **AdministratorAccess**. Seleziona quindi **Crea gruppo**.

## Crea gruppo



Crea un gruppo e seleziona le policy da collegare al gruppo. L'utilizzo dei gruppi è una procedura consigliata per gestire le autorizzazioni degli utenti in base alle funzioni lavorative, all'accesso al servizio AWS o alle autorizzazioni personalizzate. [Ulteriori informazioni](#)

Nome del  
gruppo  1

[Crea policy](#) [Aggiorna](#)

Filtro policy		Visualizzazione di 4 risultati		
	Nome policy	Tipo	Utilizzata come	Descrizione
<input checked="" type="checkbox"/>	AdministratorAccess	Funzione lavorativa	Nessuna	Provides full access to AWS services and resources.
<input type="checkbox"/>	AdministratorAccess-Amplify	Gestita da AWS	Nessuna	Grants account administrative permissions while explicitly allowing ...
<input type="checkbox"/>	AdministratorAccess-AWSEL...	Gestita da AWS	Permissions policy (1)	Grants account administrative permissions. Explicitly allows develo...
<input type="checkbox"/>	AWSAuditManagerAdministr...	Gestita da AWS	Nessuna	Provides administrative access to enable or disable AWS Audit Man...

234

[Annulla](#)

[Crea gruppo](#)

12. Nell'elenco dei gruppi seleziona la casella di controllo per il tuo nuovo gruppo. Se necessario, seleziona **Aggiorna** per visualizzare il gruppo nell'elenco.
13. Seleziona **Successivo: Tag**.

## Aggiungi utente

1 2 3 4 5

### ▼ Imposta autorizzazioni

 Add user to group (Aggiungi utente al gruppo)	 Copia le autorizzazioni dall'utente esistente	 Collega direttamente le policy esistenti
--	---	--

Aggiungi un utente a un gruppo esistente o creane uno nuovo. L'utilizzo dei gruppi costituisce una procedura consigliata per gestire le autorizzazioni dell'utente in base alle funzioni lavorative. [Ulteriori informazioni](#)

#### Add user to group (Aggiungi utente al gruppo)

Crea gruppo  1

Gruppo ▾		Policy collegate
<input type="checkbox"/>	[REDACTED]	[REDACTED]

Cerca Visualizzazione di 4 risultati

### ► Imposta limite di autorizzazioni

2 Annulla Precedente Successivo: Tag

14. (Facoltativo) Aggiungi metadati all'utente collegando i tag come coppie chiave-valore.

## Aggiungi utente

1 2 3 4 5

### Aggiungi tag (opzionale)

I tag IAM sono coppie chiave-valore che puoi aggiungere a utente. I tag possono includere informazioni sull'utente come, ad esempio, un indirizzo e-mail, oppure possono essere descrittivi come, ad esempio, una posizione professionale. È possibile usare i tag per organizzare, monitorare o controllare l'accesso per questo utente. [Ulteriori informazioni](#)

Chiave	Valore (facoltativo)	Rimuovi
<input type="text"/> Aggiungi una nuova chiave 1		

Puoi aggiungere 50 altri tag.

2 Annulla Precedente Successivo: Verifica

15. Seleziona **Successivo: Verifica** per visualizzare l'elenco dei membri del gruppo da aggiungere al nuovo utente. Quando sei pronto per continuare, seleziona **Crea utente**.

## Fase 2: Creazione di un ID chiave di accesso e una chiave di accesso segreta

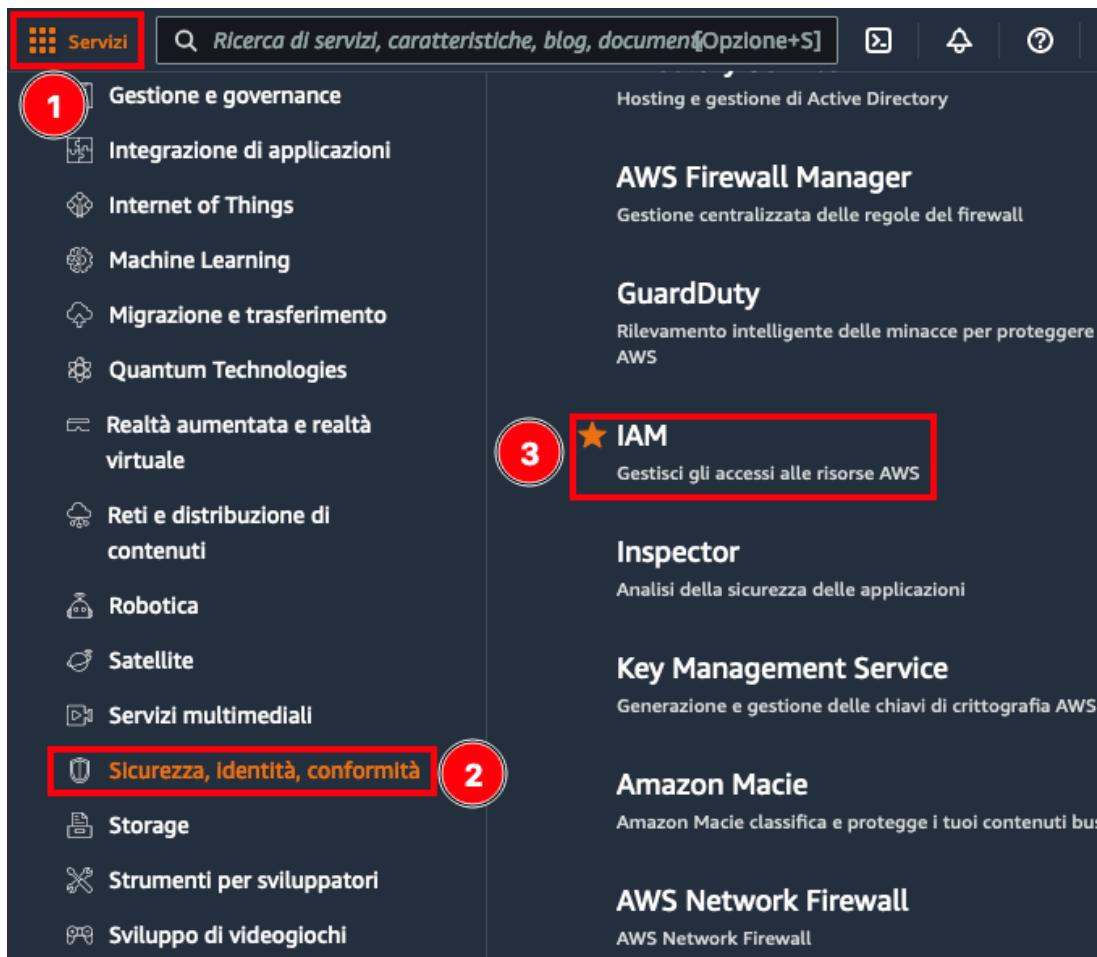
Per l'accesso alla CLI, sono necessari un ID chiave di accesso e una chiave di accesso segreta. Utilizza le chiavi di accesso utente IAM invece delle chiavi di accesso dell'utente root dell'Account AWS. IAM consente di controllare in modo sicuro l'accesso ai servizi e alle risorse Servizi AWS nell'Account AWS.

Le chiavi di accesso sono composte da un ID chiave di accesso e una chiave di accesso segreta che sono utilizzati per firmare le richieste programmatiche eseguite verso AWS. Se non si dispone di chiavi di accesso, sarà possibile crearle dalla AWS Management Console. Come best practice, non utilizzare le chiavi di accesso dell'utente root Account AWS per nessuna attività in cui non sia necessario. Invece, crea un nuovo utente IAM amministratore con le chiavi di accesso riservate a te (vedi fase precedente).

L'unica volta che è possibile visualizzare o scaricare la chiave di accesso segreta è durante la creazione delle chiavi. Non è possibile recuperarle successivamente. Tuttavia, è possibile creare nuove chiavi di accesso in qualsiasi momento. Occorre avere le autorizzazioni anche per effettuare le operazioni IAM richieste.

Come generare le chiavi di accesso per un utente IAM:

1. Accedi alla AWS Management Console e apri la console IAM.



2. Nel pannello di navigazione, seleziona **Utenti**.
3. Scegliere il nome dell'utente di cui si vogliono creare le chiavi di accesso.

The screenshot shows the AWS IAM 'Utenti' (Users) page. The left sidebar has 'Identity and Access Management (IAM)' at the top, followed by a search bar and a 'Pannello di controllo' button. Under 'Gestione degli accessi', the 'Utenti' tab is selected (1), indicated by a red circle. Below it are 'Ruoli', 'Policy', and 'Provider di identità'. The main area shows a table titled 'Utenti (20)'. The first column is 'Nome utente', which has a red box around it (2). The table includes columns for 'Gruppi', 'Ultima attività', 'MFA', and 'Valid'. A search bar at the top of the table area also has a red box around it.

4. Scegliere la scheda **Credenziali di sicurezza**.
5. Nella sezione **Chiavi di accesso**, scegliere **Crea chiave di accesso**.

## Riepilogo

The screenshot shows the AWS IAM User Details page for a specific user. At the top, there are fields for 'ARN utente' (arn:aws:iam:...), 'Percorso' (/), and 'Data di creazione' (2022-03-23 16:00 UTC+0200). Below these are tabs: 'Autorizzazioni', 'Gruppi (1)', 'Tag (1)', 'Credenziali di sicurezza' (selected, highlighted with a red box), and 'Consulente accessi'. The 'Credenziali di accesso' section contains a 'Riepilogo' table with items like 'Collegamento di accesso alla console' and a 'Password console' status. The 'Chiavi di accesso' section contains a note about using access keys and a 'Crea chiave di accesso' button (2), which is also highlighted with a red box.

6. Per visualizzare la nuova chiave di accesso, seleziona **Mostra**. Non sarà possibile accedere nuovamente alla chiave di accesso segreta dopo la chiusura di questa finestra di dialogo. Le credenziali saranno simili a quanto segue:
  - ID chiave di accesso: AKIAIOSFODNN7EXAMPLE
  - Chiave di accesso segreta: wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
7. Per fare il download della coppia di chiavi, scegliere **Scarica il file .csv**. Conserva le chiavi in un posto sicuro. Non sarà possibile accedere nuovamente alla chiave di accesso segreta dopo la chiusura di questa finestra di dialogo.
8. Mantieni la riservatezza delle chiavi in modo da proteggere l'account Account AWS e non inviarle mai via e-mail. Non condividerle all'esterno della tua organizzazione, anche se ricevi una richiesta che sembra provenire da AWS o Amazon.com. Nessuno che rappresenta legittimamente Amazon richiederà mai la tua chiave segreta.
9. Dopo aver scaricato il file .csv, seleziona **Chiudi**. Quando si crea una chiave di accesso, la coppia di chiavi è attiva di default e può essere utilizzata immediatamente.

**!** Avvertenza

Non pubblicare mai la chiave di accesso segreta su piattaforme pubbliche, ad esempio GitHub. Questa operazione può compromettere la sicurezza dell'account.

**✓ Operazione riuscita**

Questa è l' **soltanto** volta in cui le chiavi di accesso segrete possono essere visualizzate o scaricate. Non puoi recuperarle successivamente. Tuttavia, puoi creare nuove chiavi di accesso in qualsiasi momento.

 Scarica il file .csv

2

ID chiave di accesso	Chiave di accesso segreta
*****	 Mostra 

\*\*\*\*\*

Mostra

1

3

Chiudi

## 6.2 Configurazione rapida con aws configure, [1]

Per l'uso generale, il comando `aws configure` è il metodo più veloce per configurare l'installazione dell'AWS CLI. Quando digitri questo comando, nell'AWS CLI viene richiesto di immettere quattro informazioni:

1. ID chiave di accesso
2. Chiave di accesso segreta
3. Regione AWS
4. Formato di output

L'AWS CLI archivia queste informazioni in un profilo (una raccolta di impostazioni) denominato `default` nel file `credentials` (raggiungibile tramite il comando da CLI: `ls ~/.aws/credentials`).

L'esempio seguente illustra i valori di esempio:

```
$ aws configure
AWS Access Key ID [None]: AKIAIOSFODNN7EXAMPLE
AWS Secret Access Key [None]: wJalrXUtnFEMI/K7MDENG/bPxRfCYEXAMPLEKEY
Default region name [None]: us-west-2
Default output format [None]: json
```

### [1-2] ID chiave di accesso e Chiave di accesso segreta:

Le chiavi di accesso utilizzano un ID chiave di accesso e una chiave di accesso segreta che usi per firmare le richieste a livello di programmazione AWS. Il paragrafo precedente abbiamo illustrato come creare e scaricare il file .csv. Adesso vedremo come importare una key pair tramite file CSV.

Invece di utilizzare `aws configure` per inserire una key pair, è possibile importare il file .csv scaricato dopo aver creato la key pair.

Per importare il file .csv, usa il comando `aws configure import` seguito dal path del file .csv come segue:

```
$ aws configure import --csv file://credentials.csv
```

### [3] Regione AWS:

`Default region name` identifica la regione AWS ai cui server si desidera inviare le richieste per impostazione predefinita. Questa è tipicamente la regione più vicina per l'utente, ma può essere qualsiasi regione. Ad esempio, è possibile digitare `us-west-2` per utilizzare Stati Uniti occidentali (Oregon). Questa è la regione a cui verranno inviate tutte le richieste successive, se non specificato altrimenti in un singolo comando.

### [4] Formato di output:

`Default output format` specifica la modalità di formattazione dei risultati. Il valore può essere uno qualsiasi dei valori nell'elenco seguente. Se non si specifica un formato di output, `json` viene utilizzato come impostazione predefinita. Tutti i formati disponibili:

- json
- yaml
- yaml-stream
- text
- table

### Profili:

Una raccolta di impostazioni è chiamata profilo. Per impostazione predefinita, l'AWS CLI utilizza il profilo `default`. Puoi creare e utilizzare profili denominati aggiuntivi con credenziali e impostazioni variabili specificando l'opzione `--profile` e assegnando un nome.

Nell'esempio seguente viene creato un profilo denominato `produser`.

```
$ aws configure --profile produser
AWS Access Key ID [None]: AKIAI44QH8DHBEEXAMPLE
AWS Secret Access Key [None]: je7MtGbClwBF/2Zp9Utk/h3yCo8nvbEXAMPLEKEY
Default region name [None]: us-east-1
Default output format [None]: text
```

Puoi specificare un `--profile profilename` e utilizzare le credenziali e le impostazioni archiviate con quel nome.

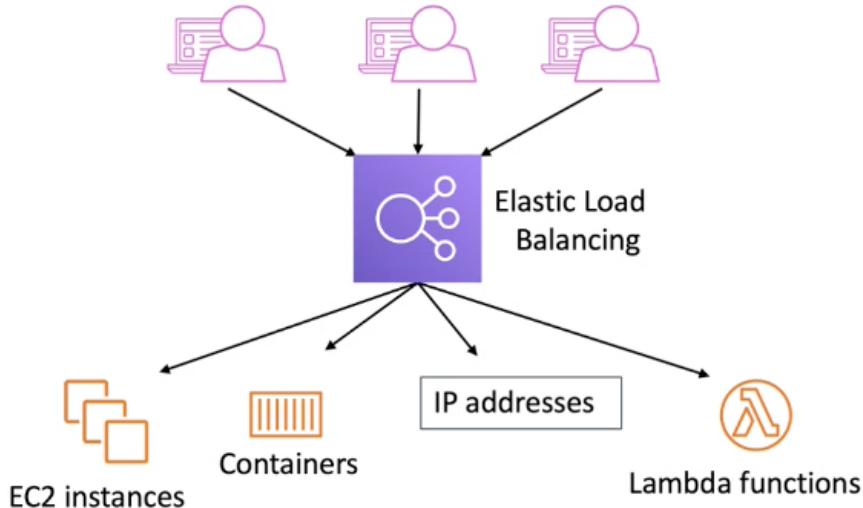
```
$ aws s3 ls --profile produser
```

Per aggiornare queste impostazioni, esegui nuovamente `aws configure` (con o senza il parametro `--profile`, a seconda del profilo che desideri aggiornare) e immetti i nuovi valori a seconda delle necessità.

[da terminare].....

## 7. Elastic Load Balancing (ELB), [1]

Elastic Load Balancing (ELB) distribuisce automaticamente il traffico in entrata su più target, come istanze EC2, container, indirizzi IP e funzioni lambda in una o più Availability Zone. Monitora lo stato di salute dei target registrati e instrada il traffico solo verso i target sani. Elastic Load Balancing scala automaticamente la capacità del bilanciatore di carico in risposta alle variazioni del traffico in entrata.

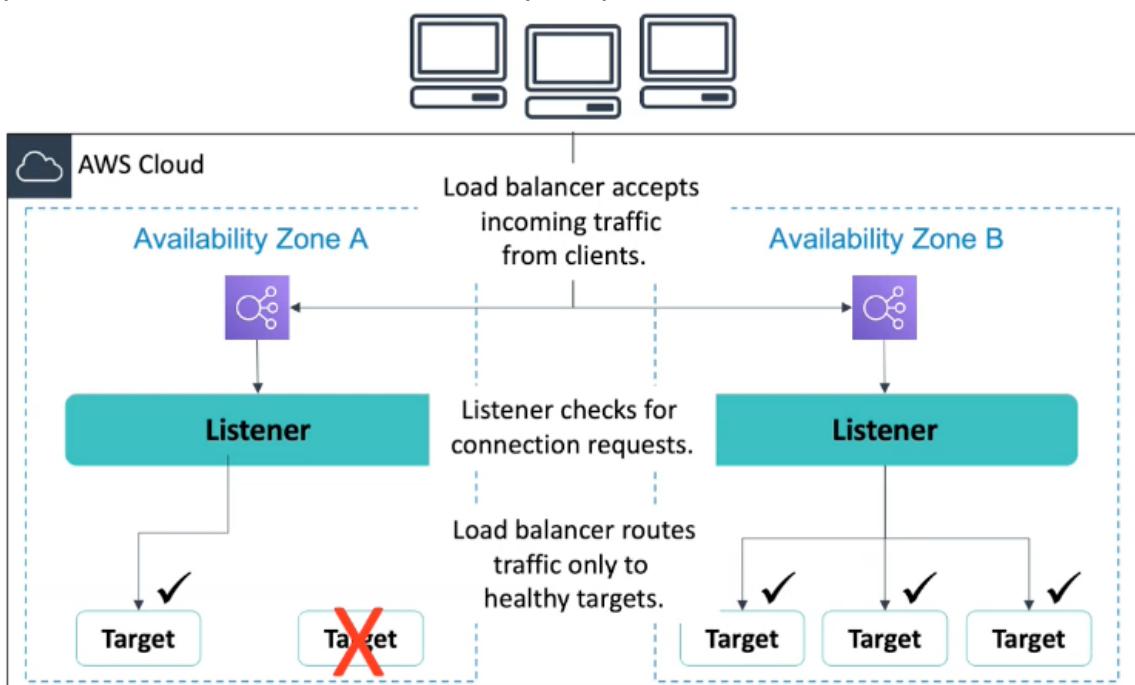


### Vantaggi del bilanciatore di carico:

Un bilanciatore di carico distribuisce i carichi di lavoro su più risorse di calcolo, come i server virtuali. L'uso di un bilanciatore di carico aumenta la disponibilità e la tolleranza agli errori delle applicazioni.

È possibile aggiungere e rimuovere risorse di calcolo dal load balancer in base alle proprie esigenze, senza interrompere il flusso complessivo delle richieste alle applicazioni.

È possibile configurare controlli di salute, che monitorano lo stato di salute delle risorse di calcolo, in modo che il load balancer invii le richieste solo a quelle sane. È anche possibile scaricare il lavoro di crittografia e decrittografia sul load balancer, in modo che le risorse di calcolo possano concentrarsi sul loro lavoro principale.



Si configura il bilanciatore di carico per accettare il traffico in entrata specificando uno o più ascoltatori. Un ascoltatore è un processo che controlla le richieste di connessione. È configurato con un protocollo e un numero di porta per le connessioni dai client al bilanciatore di carico. Allo stesso modo, è configurato con un protocollo e un numero di porta per le connessioni dal bilanciatore di carico ai target.

Elastic Load Balancing supporta i seguenti tipi di bilanciatori di carico:

- Application Load Balancers
- Network Load Balancers
- Gateway Load Balancers
- Classic Load Balancers

C'è una differenza fondamentale nel modo in cui vengono configurati i tipi di bilanciatori di carico. Con gli Application Load Balancer, i Network Load Balancer e i Gateway Load Balancer, si registrano i target in gruppi di target e si instrada il traffico verso i gruppi di target. Con i Classic Load Balancers, si registrano le istanze con il bilanciatore di carico.

### **Zone di disponibilità e nodi del bilanciatore di carico:**

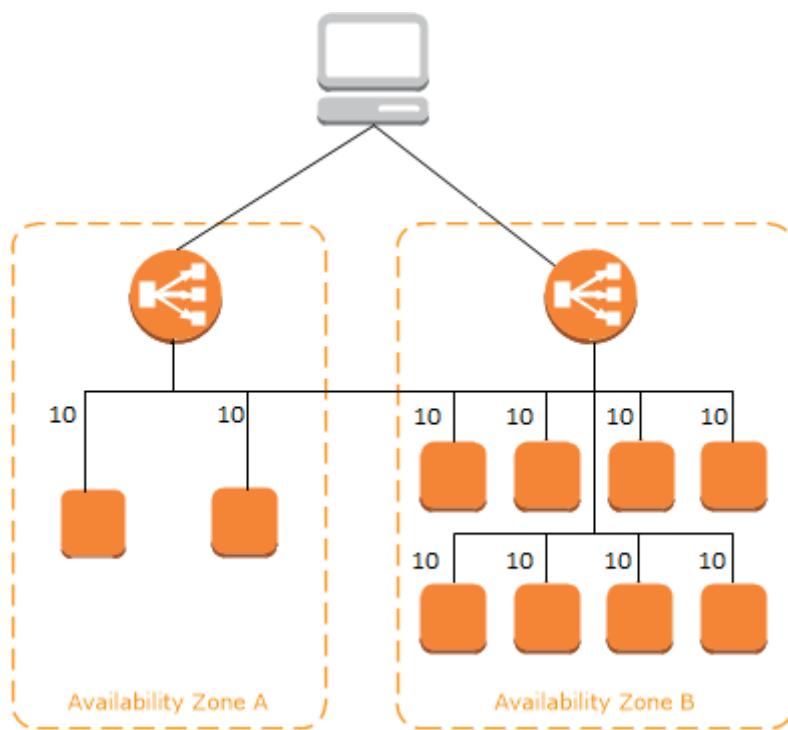
Quando si attiva una Availability Zone per il bilanciatore di carico, Elastic Load Balancing crea un nodo del bilanciatore di carico nella Availability Zone. Se si registrano target in una Availability Zone ma non si abilita la Availability Zone, questi target registrati non ricevono traffico. Il bilanciatore di carico è più efficace quando ci si assicura che ogni Availability Zone abilitata abbia almeno un target registrato.

Si consiglia di abilitare più Availability Zone per tutti i bilanciatori di carico. Tuttavia, per gli Application Load Balancer è necessario abilitare almeno due o più Availability Zone. Questa configurazione garantisce che il bilanciatore di carico possa continuare a instradare il traffico. Se una Availability Zone diventa non disponibile o non ha target sani, il bilanciatore di carico può instradare il traffico verso i target sani di un'altra Availability Zone.

Dopo aver disabilitato una Availability Zone, i target di quella Availability Zone rimangono registrati con il load balancer. Tuttavia, anche se rimangono registrati, il bilanciatore di carico non instrada il traffico verso di loro.

### **Bilanciamento del carico tra zone:**

I diagrammi seguenti dimostrano l'effetto del bilanciamento del carico tra zone con round robin come algoritmo di routing predefinito. Ci sono due Availability Zone abilitate, con due target nell'Availability Zone A e otto target nell'Availability Zone B. I client inviano richieste e Amazon Route 53 risponde a ciascuna richiesta con l'indirizzo IP di uno dei nodi di bilanciamento del carico. In base all'algoritmo di routing round robin, il traffico è distribuito in modo tale che ogni nodo di bilanciamento del carico riceva il 50% del traffico totale dei client. Ogni nodo di bilanciamento del carico distribuisce la sua quota di traffico tra gli obiettivi registrati nel suo ambito.

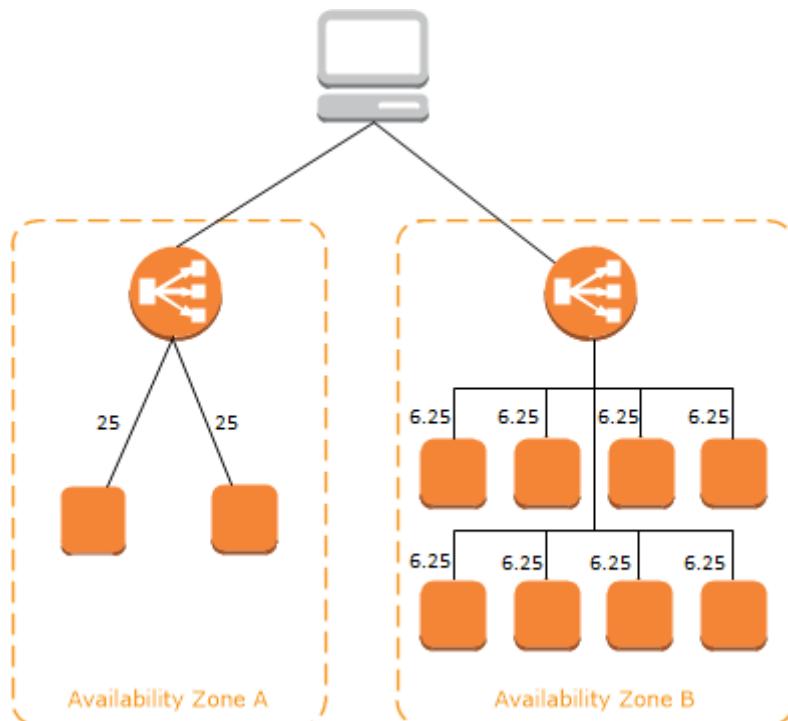


Se il bilanciamento del carico tra zone è abilitato, ciascuno dei 10 target riceve il 10% del traffico. Questo perché ogni nodo di bilanciamento del carico può instradare il 50% del traffico dei client verso tutti e 10 i target.

Se il bilanciamento del carico tra zone è disabilitato:

- Ciascuno dei due target della zona di disponibilità A riceve il 25% del traffico.
- Ciascuno degli otto target della zona di disponibilità B riceve il 6,25% del traffico.

Questo perché ogni nodo di bilanciamento del carico può instradare il 50% del traffico client solo verso i target della propria Availability Zone.



Con gli Application Load Balancer, il bilanciamento del carico tra zone è sempre abilitato.

Con i bilanciatori di carico di rete e i bilanciatori di carico gateway, il bilanciamento del carico tra zone è disabilitato per impostazione predefinita. Dopo aver creato il bilanciatore di carico, è possibile attivare o disattivare il bilanciamento del carico tra zone in qualsiasi momento.

Quando si crea un bilanciatore di carico classico, l'impostazione predefinita per il bilanciamento del carico tra zone dipende dalla modalità di creazione del bilanciatore di carico. Con l'API o la CLI, il bilanciamento del carico tra zone è disabilitato per impostazione predefinita. Con la console di gestione AWS, l'opzione per abilitare il bilanciamento del carico tra zone è selezionata per impostazione predefinita. Dopo aver creato un Classic Load Balancer, è possibile abilitare o disabilitare il bilanciamento del carico tra zone in qualsiasi momento. Per ulteriori informazioni, consultare Abilita il bilanciamento del carico tra zone nella Guida dell'utente per i Classic Load Balancer.

### Instradamento delle richieste:

Prima che un client invii una richiesta al bilanciatore di carico, il client risolve il nome di dominio del bilanciatore di carico utilizzando un server DNS (Domain Name System). La voce DNS è controllata da Amazon, perché i bilanciatori di carico si trovano nel dominio amazonaws.com. I server DNS di Amazon restituiscono uno o più indirizzi IP al client. Questi sono gli indirizzi IP dei nodi del bilanciatore di carico per il vostro bilanciatore di carico. Con i bilanciatori di carico di rete, Elastic Load Balancing crea un'interfaccia di rete per ogni Availability Zone attivata. Ogni nodo di bilanciamento del carico nella Availability Zone utilizza questa interfaccia di rete per ottenere un indirizzo IP statico. È possibile associare facoltativamente un indirizzo IP elastico a ciascuna interfaccia di rete quando si crea il bilanciatore di carico.

### Casi d'uso:

You must support traffic to a containerized application.	Application Load Balancer
You have extremely spiky and unpredictable TCP traffic.	Network Load Balancer
You need simple load balancing with multiple protocols.	Classic Load Balancer
You need to support a static or Elastic IP address, or an IP target outside a VPC.	Network Load Balancer
You need a load balancer that can handle millions of requests per second while maintaining low latencies.	Network Load Balancer
You must support HTTPS requests.	Application Load Balancer

## 7.1 Amazon CloudWatch, [1]

Amazon CloudWatch è un servizio di monitoraggio e gestione che fornisce dati e informazioni concrete per risorse di infrastruttura e per applicazioni on-premise, AWS e ibride. È possibile raccogliere e visualizzare tutti i dati relativi a prestazioni e operatività sotto forma di log e parametri, in un'unica piattaforma anziché monitorarli in silos (server, rete o database).

CloudWatch permette di monitorare lo stack completo (applicazioni, infrastruttura e servizi) e usare allarmi, log e dati relativi ad eventi per attivare operazioni automatizzate e ridurre i tempi medi di risoluzione dei problemi (MTTR). È così possibile liberare risorse e concentrare i propri sforzi sulla creazione di applicazioni e di valore aziendale.

CloudWatch fornisce informazioni concrete che aiutano a ottimizzare le prestazioni delle applicazioni, gestire l'utilizzo delle risorse e non perdere di vista lo stato di integrità operativa a

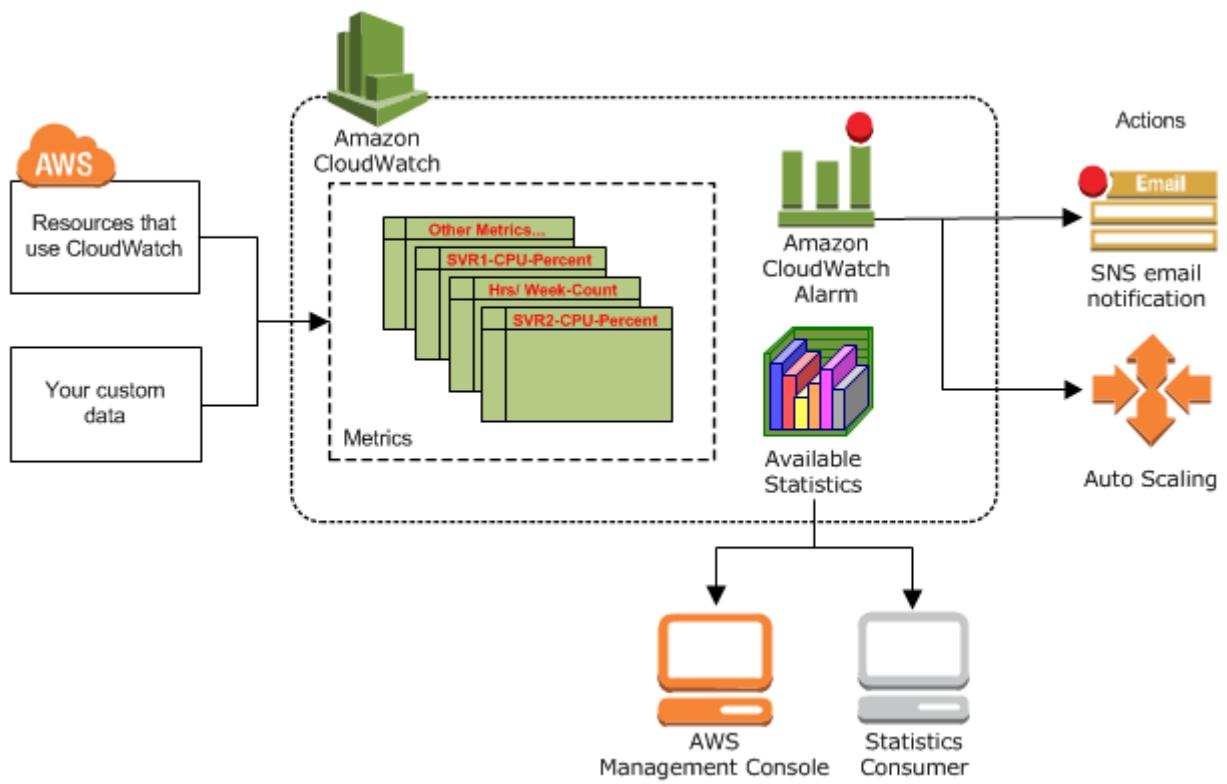
livello di sistema. CloudWatch fornisce visibilità al secondo di parametri e dati di log, 15 mesi di conservazione dei dati (parametri) e la possibilità di eseguire calcoli sui parametri. È così possibile eseguire analisi cronologiche per ottimizzare i costi e acquisire informazioni in tempo reale su come ottimizzare risorse di infrastruttura e applicazioni. È possibile usare CloudWatch Container Insights per monitorare, risolvere i problemi e creare allarmi per le tue applicazioni e i tuoi microservizi con container. CloudWatch raccoglie, aggrega e riepiloga le informazioni sull'utilizzo del calcolo come CPU, memoria, disco e dati di rete, nonché informazioni diagnostiche come errori di riavvio di container, per aiutare gli ingegneri DevOps a isolare i problemi e risolverli in modo rapido. Container Insights offre informazioni dettagliate dai servizi di gestione dei container come Amazon ECS per Kubernetes (EKS), Elastic Container Service (ECS) di Amazon, AWS Fargate e Kubernetes indipendenti (k8s).

Con Amazon CloudWatch vengono utilizzati i servizi seguenti:

- **Amazon Simple Notification Service (Amazon SNS)**, coordina e gestisce la consegna o l'invio di messaggi agli endpoint o ai clienti abbonati. È possibile utilizzare Amazon SNS con CloudWatch per l'invio di messaggi quando viene raggiunta una soglia di allarme.
- **Amazon EC2 Auto Scaling**, permette di avviare o terminare automaticamente istanze Amazon EC2 in base a policy definite dall'utente, controlli dello stato di integrità e pianificazioni. È possibile utilizzare un allarme CloudWatch con Amazon EC2 Auto Scaling per dimensionare le istanze EC2 on demand.
- **AWS CloudTrail**, permette di monitorare le chiamate effettuate all'API di Amazon CloudWatch per il proprio account, tra cui le chiamate effettuate dalla AWS Management Console, da AWS CLI e da altri servizi. Quando è attiva la registrazione in CloudTrail, CloudWatch scrive file di log sul bucket Amazon S3 specificato durante la configurazione di CloudTrail.
- **AWS Identity and Access Management (IAM)**, è un servizio Web che aiuta a controllare in modo sicuro l'accesso alle risorse AWS per gli utenti. È possibile utilizzare IAM per controllare chi può utilizzare le delle risorse AWS (autenticazione), quali risorse e in che modo (autorizzazione).

#### **Funzionamento di Amazon CloudWatch:**

Amazon CloudWatch è fondamentalmente un repository di parametri. Un servizio AWS, ad esempio Amazon EC2, salva i parametri in archivio. Sulla base di questi parametri puoi recuperare le statistiche. Puoi anche recuperare le statistiche basate su parametri personalizzati, se questi sono stati messi in archivio.



lezione 18 minuti 35 rimanenti

## 8. Kubernetes, [1], [killercoda], [play-with-k8s]

[https://linuxacademy.com/site-content/uploads/2019/04/Kubernetes-Cheat-Sheet\\_07182019.pdf](https://linuxacademy.com/site-content/uploads/2019/04/Kubernetes-Cheat-Sheet_07182019.pdf)

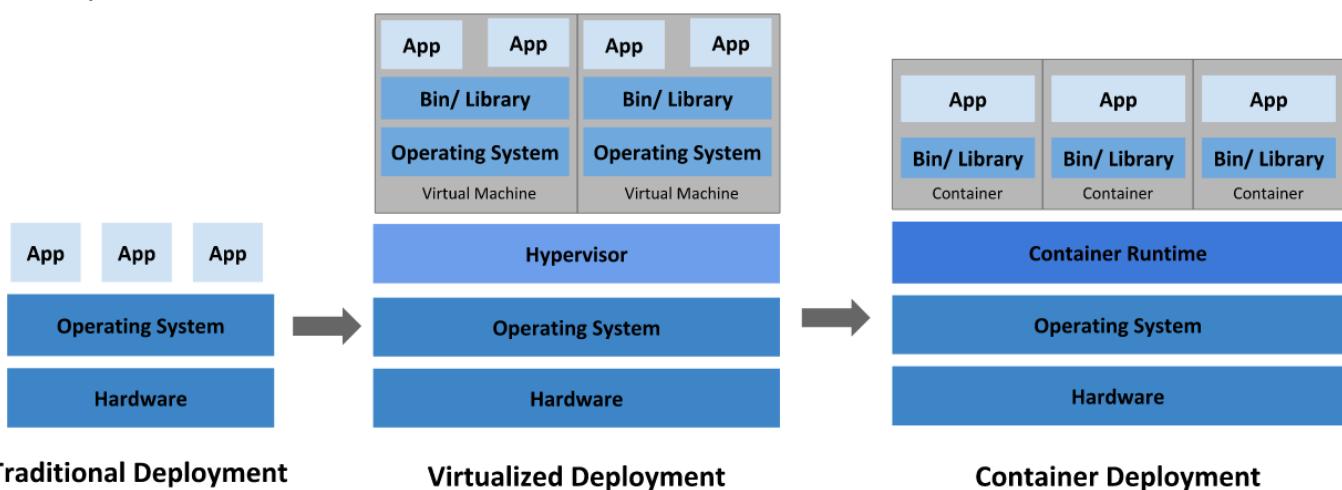
<https://phoenixnap.com/kb/wp-content/uploads/2021/11/kubectl-commands-cheat-sheet-by-pnap.pdf>

<https://intellipaat.com/mediaFiles/2019/03/Kubernetes-Cheat-Sheet.pdf>

Kubernetes è una piattaforma portatile, estensibile e open-source per la gestione di carichi di lavoro e servizi containerizzati, in grado di facilitare sia la configurazione dichiarativa che l'automazione. La piattaforma vanta un grande ecosistema in rapida crescita. Servizi, supporto e strumenti sono ampiamente disponibili nel mondo Kubernetes.

### 8.1 Facciamo un piccolo salto indietro, [1]

Diamo un'occhiata alla ragione per cui Kubernetes è così utile facendo un piccolo salto indietro nel tempo.



**L'era del deployment tradizionale:** All'inizio, le organizzazioni eseguivano applicazioni su server fisici. Non c'era modo di definire i limiti delle risorse per le applicazioni in un server fisico e questo ha causato non pochi problemi di allocazione delle risorse. Ad esempio, se più applicazioni vengono eseguite sullo stesso server fisico, si possono verificare casi in cui un'applicazione assorbe la maggior parte delle risorse e, di conseguenza, le altre applicazioni non hanno le prestazioni attese. Una soluzione per questo sarebbe di eseguire ogni applicazione su un server fisico diverso. Ma questa non è una soluzione ideale, dal momento che le risorse vengono sottoutilizzate, inoltre, questa pratica risulta essere costosa per le organizzazioni, le quali devono mantenere numerosi server fisici.

**L'era del deployment virtualizzato:** Come soluzione venne introdotta la virtualizzazione. Essa consente di eseguire più macchine virtuali (VM) su una singola CPU fisica. La virtualizzazione consente di isolare le applicazioni in più macchine virtuali e fornisce un livello di sicurezza superiore, dal momento che le informazioni di un'applicazione non sono liberamente accessibili da un'altra applicazione.

La virtualizzazione consente un migliore utilizzo delle risorse riducendo i costi per l'hardware, permette una migliore scalabilità, dato che un'applicazione può essere aggiunta o aggiornata facilmente, e ha molti altri vantaggi.

Ogni VM è una macchina completa che esegue tutti i componenti, compreso il proprio sistema operativo, sopra all'hardware virtualizzato.

**L'era del deployment in container:** I container sono simili alle macchine virtuali, ma presentano un modello di isolamento più leggero, condividendo il sistema operativo (OS) tra le

applicazioni. Pertanto, i container sono considerati più leggeri. Analogamente a una macchina virtuale, un container dispone di una segregazione di filesystem, CPU, memoria, PID e altro ancora. Poiché sono disaccoppiati dall'infrastruttura sottostante, risultano portabili tra differenti cloud e diverse distribuzioni.

I container sono diventati popolari dal momento che offrono molteplici vantaggi, ad esempio:

- Creazione e distribuzione di applicazioni in modalità Agile: maggiore facilità ed efficienza nella creazione di immagini container rispetto all'uso di immagini VM.
- Adozione di pratiche per lo sviluppo/test/rilascio continuativo: consente la frequente creazione e la distribuzione di container image affidabili, dando la possibilità di fare rollback rapidi e semplici (grazie all'immutabilità dell'immagine stessa).
- Separazione delle fasi di Dev e Ops: le container image vengono prodotte al momento della compilazione dell'applicativo piuttosto che nel momento del rilascio, permettendo così di disaccoppiare le applicazioni dall'infrastruttura sottostante.
- L'osservabilità non riguarda solo le informazioni e le metriche del sistema operativo, ma anche lo stato di salute e altri segnali dalle applicazioni.
- Coerenza di ambiente tra sviluppo, test e produzione: i container funzionano allo stesso modo su un computer portatile come nel cloud.
- Portabilità tra cloud e sistemi operativi differenti: lo stesso container funziona su Ubuntu, RHEL, CoreOS, on-premise, nei più grandi cloud pubblici e da qualsiasi altra parte.
- Gestione incentrata sulle applicazioni: Aumenta il livello di astrazione dall'esecuzione di un sistema operativo su hardware virtualizzato all'esecuzione di un'applicazione su un sistema operativo utilizzando risorse logiche.
- Microservizi liberamente combinabili, distribuiti, ad alta scalabilità: le applicazioni sono suddivise in pezzi più piccoli e indipendenti che possono essere distribuite e gestite dinamicamente - niente stack monolitici che girano su una singola grande macchina.
- Isolamento delle risorse: le prestazioni delle applicazioni sono prevedibili.
- Utilizzo delle risorse: alta efficienza e densità.

## 8.2 Perché necessito di Kubernetes e cosa posso farci?, [1]

I container sono un buon modo per distribuire ed eseguire le tue applicazioni. In un ambiente di produzione, è necessario gestire i container che eseguono le applicazioni e garantire che non si verifichino interruzioni dei servizi. Per esempio, se un container si interrompe, è necessario avviare un nuovo container. Non sarebbe più facile se questo comportamento fosse gestito direttamente da un sistema?

È proprio qui che Kubernetes viene in soccorso! Kubernetes ti fornisce un framework per far funzionare i sistemi distribuiti in modo resiliente. Kubernetes si occupa della scalabilità, failover, distribuzione delle tue applicazioni. Per esempio, Kubernetes può facilmente gestire i rilasci con modalità Canary deployment.

Kubernetes ti fornisce:

1. **Scoperta dei servizi e bilanciamento del carico.** Kubernetes può esporre un container usando un nome DNS o il suo indirizzo IP. Se il traffico verso un container è alto, Kubernetes è in grado di distribuire il traffico su più container in modo che il servizio rimanga stabile.

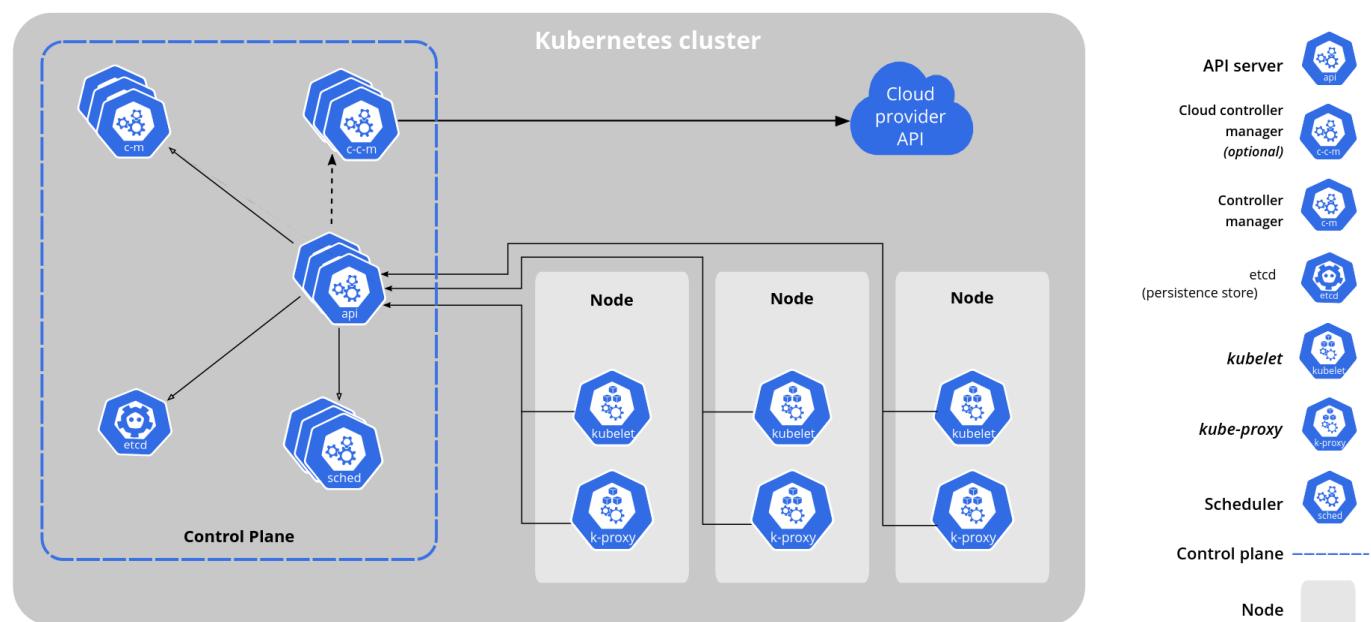
2. **Orchestrazione dello storage.** Kubernetes ti permette di montare automaticamente un sistema di archiviazione di vostra scelta, come per esempio storage locale, dischi forniti da cloud pubblici, e altro ancora.
3. **Rollout e rollback automatizzati.** Puoi utilizzare Kubernetes per descrivere lo stato desiderato per i propri container, e Kubernetes si occuperà di cambiare lo stato attuale per raggiungere quello desiderato ad una velocità controllata. Per esempio, puoi automatizzare Kubernetes per creare nuovi container per il tuo servizio, rimuovere i container esistenti e adattare le loro risorse a quelle richieste dal nuovo container.
4. **Ottimizzazione dei carichi.** Fornisci a Kubernetes un cluster di nodi per eseguire i container. Puoi istruire Kubernetes su quanta CPU e memoria (RAM) ha bisogno ogni singolo container. Kubernetes allocherà i container sui nodi per massimizzare l'uso delle risorse a disposizione.
5. **Self-healing.** Kubernetes riavvia i container che si bloccano, sostituisce container, termina i container che non rispondono agli health checks, e evita di far arrivare traffico ai container che non sono ancora pronti per rispondere correttamente.
6. **Gestione di informazioni sensibili e della configurazione.** Kubernetes consente di memorizzare e gestire informazioni sensibili, come le password, i token OAuth e le chiavi SSH. Puoi distribuire e aggiornare le informazioni sensibili e la configurazione dell'applicazione senza dover ricostruire le immagini dei container e senza rivelare le informazioni sensibili nella configurazione del tuo sistema.

### 8.3 I componenti di Kubernetes, [1]

Quando si distribuisce Kubernetes, si ottiene un **cluster**. Un cluster Kubernetes consiste in un insieme di macchine **worker**, chiamate **nodi**, che eseguono applicazioni containerizzate. Ogni cluster ha almeno un nodo worker.

I nodi worker ospitano i **Pod** che sono i componenti del carico di lavoro dell'applicazione. Il control plane gestisce i nodi worker e i Pod nel cluster. Negli ambienti di produzione, il control plane di solito viene eseguito su più computer, e un cluster di solito gestisce più nodi, fornendo tolleranza agli errori e alta disponibilità.

La seguente immagine illustra i vari componenti necessari per un cluster Kubernetes completo e funzionante.



### 8.3.1 Componenti del Control Plane, [1]

I componenti del Control Plane sono responsabili di tutte le decisioni globali sul cluster (ad esempio, lo scheduling) oltre che a rilevare e rispondere agli eventi del cluster (ad esempio, l'avvio di un nuovo pod quando il valore replicas di un deployment non è soddisfatto).

I componenti del control plane possono essere eseguiti su qualsiasi macchina del cluster.

Tuttavia, per semplicità, gli script di configurazione di solito avviano tutti i componenti del control plane sulla stessa macchina e non eseguono i container degli utenti su questa macchina.

- **kube-apiserver.** L'API server è un componente di Kubernetes control plane che espone le Kubernetes API. L'API server è il front end del control plane di Kubernetes. La principale implementazione di un server Kubernetes API è kube-apiserver.  
kube-apiserver è progettato per scalare orizzontalmente, cioè scala aumentando il numero di istanze. Puoi eseguire multiple istanze di kube-apiserver e bilanciare il traffico tra queste istanze.
- **etcd.** È un database key-value ridondato, che è usato da Kubernetes per salvare tutte le informazioni del cluster. Se il vostro cluster Kubernetes utilizza etcd come backing store, assicuratevi di avere un piano di backup per questi dati.
- **kube-scheduler.** Componente della Control Plane che controlla i pod appena creati che non hanno un nodo assegnato, e dopo averlo identificato glielo assegna. I fattori presi in considerazione nell'individuare un nodo a cui assegnare l'esecuzione di un Pod includono la richiesta di risorse del Pod stesso e degli altri workload presenti nel sistema, i vincoli delle hardware/software/policy, le indicazioni di affinity e di anti-affinity, requisiti relativi alla disponibilità di dati/Volumes, le interferenze tra diversi workload e le scadenze.
- **kube-controller-manager.** Componente della Control Plane che gestisce controllers. Da un punto di vista logico, ogni controller è un processo separato, ma per ridurre la complessità, tutti i principali controller di Kubernetes vengono raggruppati in un unico container ed eseguiti in un singolo processo.
- **cloud-controller-manager.** Un componente della control plane di Kubernetes che aggiunge logiche di controllo specifiche per il cloud. Il cloud-controller-manager permette di collegare il cluster con le API del cloud provider e separa le componenti che interagiscono con la piattaforma cloud dai componenti che interagiscono solamente col cluster. Il cloud-controller-manager esegue dei controller specifici del cloud provider. Come nel kube-controller-manager, il cloud-controller-manager combina diversi control loop logicamente indipendenti in un singolo binario che puoi eseguire come un singolo processo. Tu puoi scalare orizzontalmente (eseguire più di una copia) per migliorare la responsività o per migliorare la tolleranza ai fallimenti.

### 8.3.2 Componenti dei Nodi, [1]

I componenti del nodo vengono eseguiti su ogni nodo, mantenendo i pod in esecuzione e forniscono l'ambiente di runtime Kubernetes.

- **kubelet.** Si tratta di un agente che viene eseguito su ogni nodo del cluster. Si assicura che i container siano eseguiti in un pod. La kubelet riceve un set di *PodSpecs* che vengono forniti attraverso vari meccanismi, e si assicura che i container descritti in questi PodSpecs funzionino correttamente e siano sani. La kubelet non gestisce i container che non sono stati creati da Kubernetes.
- **kube-proxy.** Kube-proxy è un proxy eseguito su ogni nodo del cluster, responsabile della gestione dei Kubernetes Service. I kube-proxy mantengono le regole di networking sui nodi. Queste regole permettono la comunicazione verso gli altri nodi del cluster o

l'esterno. Il kube-proxy usa le librerie del sistema operativo quando possibile; in caso contrario il kube-proxy gestisce direttamente il traffico.

- **Container Runtime.** Il container runtime è il software che è responsabile per l'esecuzione dei container. Kubernetes supporta diversi container runtimes: Docker, containerd, cri-o, rktlet e tutte le implementazioni di Kubernetes CRI (Container Runtime Interface).

È possibile inserire anche degli [addons](#).

## 8.4 Lavorare con gli oggetti Kubernetes, [1]

Gli oggetti Kubernetes sono entità persistenti nel sistema Kubernetes. Kubernetes utilizza queste entità per rappresentare lo stato del cluster. In particolare, possono descrivere:

- Quali applicazioni containerizzate sono in esecuzione (e su quali nodi);
- Le risorse disponibili per tali applicazioni;
- Le politiche di comportamento delle applicazioni, come le politiche di riavvio, gli aggiornamenti e la tolleranza ai guasti.

### 8.4.1 Specifica e stato dell'oggetto, [1]

Quasi tutti gli oggetti di Kubernetes includono due campi annidati che regolano la configurazione dell'oggetto: la specifica dell'oggetto e lo stato dell'oggetto. Per gli oggetti che hanno una specifica, è necessario impostarla quando si crea l'oggetto, fornendo una descrizione delle caratteristiche che si desidera che la risorsa abbia: il suo stato desiderato. Lo stato descrive lo stato attuale dell'oggetto, fornito e aggiornato dal sistema Kubernetes e dai suoi componenti. Il piano di controllo di Kubernetes gestisce continuamente e attivamente lo stato effettivo di ogni oggetto per farlo corrispondere allo stato desiderato dall'utente.

Ad esempio, in Kubernetes, un Deployment è un oggetto che può rappresentare un'applicazione in esecuzione sul cluster. Quando si crea il Deployment, si può impostare la specifica del Deployment per specificare che si vogliono tre repliche dell'applicazione in esecuzione. Il sistema Kubernetes legge le specifiche del Deployment e avvia tre istanze dell'applicazione desiderata, aggiornando lo stato in base alle specifiche. Se una di queste istanze dovesse guastarsi (un cambiamento di stato), il sistema Kubernetes risponde alla differenza tra le specifiche e lo stato apportando una correzione, in questo caso avviando un'istanza sostitutiva.

### Descrivere un oggetto Kubernetes, [1]

Quando si crea un oggetto in Kubernetes, è necessario fornire la specifica dell'oggetto che descrive lo stato desiderato, oltre ad alcune informazioni di base sull'oggetto (come il nome). Quando si utilizza l'API di Kubernetes per creare l'oggetto (direttamente o tramite kubectl), la richiesta API deve includere queste informazioni come JSON nel corpo della richiesta. Nella maggior parte dei casi, si forniscono le informazioni a kubectl in un file .yaml. kubectl converte le informazioni in JSON quando effettua la richiesta API.

Ecco un esempio di file .yaml che mostra i campi obbligatori e le specifiche degli oggetti per una distribuzione Kubernetes:

```

apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment
spec:
  selector:
    matchLabels:
      app: nginx
  replicas: 2 # tells deployment to run 2 pods matching the template
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
        - name: nginx
          image: nginx:1.14.2
          ports:
            - containerPort: 80

```

Un modo per creare un deployment usando un file .yaml come quello sopra è usare il comando **kubectl apply** nell'interfaccia a riga di comando di kubectl, passando il file .yaml come argomento. Ecco un esempio:

```
$ kubectl apply -f https://k8s.io/examples/application/deployment.yaml
```

L'output è simile a questo:

```
$ deployment.apps/nginx-deployment created
```

## Campi richiesti, [1]

Nel file .yaml dell'oggetto Kubernetes che si desidera creare, è necessario impostare i valori dei seguenti campi:

- **apiVersion** - La versione dell'API di Kubernetes utilizzata per creare l'oggetto.
- **kind** - Il tipo di oggetto che si vuole creare.
- **metadata** - Dati che aiutano a identificare in modo univoco l'oggetto, tra cui una stringa *name*, un *UID* e un *namespace* opzionale.
- **spec** - Lo stato che si desidera per l'oggetto

Il formato preciso della specifica dell'oggetto è diverso per ogni oggetto Kubernetes e contiene campi annidati specifici per quell'oggetto. La [Kubernetes API Reference](#) può aiutare a trovare il formato delle specifiche per tutti gli oggetti che si possono creare con Kubernetes.

## 8.5 Nodi, [1]

Kubernetes esegue il carico di lavoro inserendo i container in Pod da eseguire sui nodi. Un nodo può essere una macchina virtuale o fisica, a seconda del cluster. Ogni nodo è gestito dal control plane e contiene i servizi necessari per l'esecuzione dei Pod.

In genere si hanno diversi nodi in un cluster; in un ambiente di apprendimento o con risorse limitate, si potrebbe avere un solo nodo.

I componenti di un nodo includono kubelet, un runtime di container e kube-proxy (visti precedentemente).

### 8.5.1 Gestione, [1]

Esistono due modi principali per aggiungere i nodi al server API:

- Il kubelet su un nodo si autoregistra al control plane.
- L'utente (o un altro utente umano) aggiunge manualmente un oggetto Node.

Dopo aver creato un oggetto Node o dopo che il kubelet su un nodo si è autoregistrato, il control plane controlla se il nuovo oggetto Node è valido. Ad esempio, se si tenta di creare un nodo dal seguente manifest JSON:

```
{  
  "kind": "Node",  
  "apiVersion": "v1",  
  "metadata": {  
    "name": "10.240.79.157",  
    "labels": {  
      "name": "my-first-k8s-node"  
    }  
  }  
}
```

Kubernetes crea internamente un oggetto Node (la rappresentazione). Kubernetes verifica se al server API si è registrato un kubelet che corrisponde al campo *metadata.name* del Node. Se il nodo è sano (cioè tutti i servizi necessari sono in esecuzione), allora è idoneo a eseguire un Pod. Altrimenti, il nodo viene ignorato per qualsiasi attività del cluster finché non diventa sano.

## **8.6 Concetti, [1]**

Una volta impostato lo stato desiderato, il Kubernetes Control Plane funziona per fare in modo che lo stato corrente del cluster corrisponda allo stato desiderato. Per fare ciò, Kubernetes esegue automaticamente una serie di attività, come l'avvio o il riavvio dei contenitori, il ridimensionamento del numero di repliche di una determinata applicazione e altro ancora. Il control plane di Kubernetes è costituito da una serie di processi in esecuzione sul cluster:

- Il Kubernetes Master è una raccolta di tre processi che vengono eseguiti su un singolo nodo nel cluster, che è designato come nodo principale. Questi processi sono: kube-apiserver, kube-controller-manager e kube-scheduler.
- Ogni singolo nodo non principale nel cluster esegue due processi:
  - kubelet, che comunica con il master di Kubernetes.
  - kube-proxy, un proxy di rete che riflette i servizi di rete di Kubernetes su ciascun nodo.

### **Kubernetes Objects:**

Kubernetes contiene una serie di astrazioni che rappresentano lo stato del tuo sistema: applicazioni e carichi di lavoro distribuiti in container, le loro risorse di rete e disco associate e altre informazioni su ciò che sta facendo il tuo cluster. Queste astrazioni sono rappresentate da oggetti nell'API di Kubernetes.

Gli oggetti di base di Kubernetes includono:

- Pod. Istanze di container all'interno della singola applicazione.
- Service. Microservizi di k8s che fungono da endpoint, quindi non fanno altro che esportare, mediante aperture di porte di rete, i pods.
- Volume.
- Namespace.

## **Esempio di deployment, [1]**

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: tomcat-deployment
spec:
  selector:
    matchLabels:
      app: tomcat
  replicas: 1
  template:
    metadata:
      labels:
        app: tomcat
    spec:
      containers:
        - name: tomcat
          image: tomcat:latest
          ports:
            - containerPort: 8080
```

## Scaling delle applicazioni, [1]

Uno degli aspetti più importanti di k8s è la replica dei microservizi. Una distinzione importante è tra le applicazioni stateful e stateless.

- Un'applicazione si dice **Stateful** quando conserva uno stato interno tramite variabili di sessione. Per esempio la compilazione di form ecc.. Per l'aspetto dei microservizi questo approccio non è ideale in quanto non permette una corretta replica dei microservizi.
- Un'applicazione si dice **Stateless** è l'opposto ovvero non conserva stati interni.

Per abilitare k8s ad utilizzare le repliche è possibile tramite l'inserimento della parola chiave "replica" nel file Deployment.yaml (metodo raccomandato). Oppure, senza modificare il file .yaml, è possibile utilizzare l'opzione "scale":

```
$ kubectl scale --replicas=4 deployment/tomcat-deployment
```

Per accedere adesso però disponiamo di 4 repliche, ragion per cui anziché usare Nodeport per la gestione delle porte si utilizza LoadBalancer:

```
$ kubectl expose deployment tomcat-deployment --type=LoadBalancer  
--port=8080 --target-port=8080 --name=tomcat-load-balancer
```

Durante una chiamata verremo reindirizzati su una delle 4 repliche.

```
$ kubectl describe services tomcat-load-balancer
```

## Deployments:

I deployments sono oggetti di alto livello che definiscono lo stato desiderato di un'applicazione. Noi descriviamo questo stato desiderato tramite il file deployment.yaml. Fondamentalmente il deploy si trasforma da un file di descrizione ad una serie di pod. Con l'oggetto deployment è possibile:

- Creare un nuovo deployment;
- Aggiornare un deployment esistente;
- Aggiornare i pod senza creare disservizi;
- Rollback a versioni precedenti dovuti per esempio a dei bug;
- È possibile mettere in pausa e riavviare l'applicativo.

Per verificare i deployments digitare:

```
$ kubectl get deployments
```

Per tornare alla versione precedente digitare il comando:

```
$ kubectl rollout status
```

Per aggiornare un'immagine utilizziamo il comando:

```
$ kubectl set image
```

Per visionare lo storico delle versioni digitare il comando:

```
$ kubectl rollout history
```

## Labels & Selectors, [1]

Poiché il deployment si occupa di tutto l'applicativo spesso assume notevoli complessità. Per non creare confusione esistono le *labels* e i *selectors*. Sono dei metodi con cui si vanno ad etichettare gli oggetti in k8s, come nodi, servizi, pods, immagini, ecc. L'etichettatura è formata

da due campi: chiave=valore. I selector sono dei criteri con cui andiamo a discriminare gli oggetti in base alle label assegnate.

**Esercizio:** Vogliamo creare una label denominata "storage" definendola con valore "SSD", inoltre vogliamo usare un selector nel deployment con il quale diciamo a k8s di fare il deploy soltanto in quelle macchine (o nodi) che sono stati etichettati con la label "SSD".

Verifichiamo i nodi disponibili con:

```
$ kubectl get nodes
```

Procediamo ad etichettare con:

```
$ kubectl label node minikube storageType=ssd
```

Utilizziamo questa etichetta con un selector:

```
1  apiVersion: apps/v1beta2
2  kind: Deployment
3  metadata:
4    name: tomcat-deployment
5  spec:
6    selector:
7      matchLabels:
8        app: tomcat
9    replicas: 4
10   template:
11     metadata:
12       labels:
13         app: tomcat
14     spec:
15       containers:
16         - name: tomcat
17           image: tomcat:9.0
18         ports:
19           - containerPort: 8080
20         nodeSelector:
21           storageType: ssd
```

Con queste due righe stiamo dicendo a k8s di fare il deploy soltanto nei nodi (con `nodeSelector`) che hanno come label chiave|valore `storageType: ssd`.

Applichiamo le modifiche con:

```
$ kubectl apply -f ./deployment.yaml
```

## Health checks/Controllo di operatività, [1]

Come fa K8s a capire se i servizi disponibili, ovvero le varie repliche, sono momentaneamente offline o online? Per farlo k8s utilizza degli strumenti che si chiamano health checks oppure probes. Ce ne sono di due tipologie:

- **Readiness probes.** Con questa tipologia K8s riesce a determinare se un pod è nello stato di *ready*.
- **Liveness probes.** Con questa tipologia K8s riesce a determinare se un pod è nello stato di *healthy* o *unhealthy*.

A prescindere dal tipo di probes utilizzato k8s esegue due tipi di controlli:

- Chiamate HTTP o TCP al pod per verificarne lo stato;
- Comandi di esecuzione al pod.

Grazie ai quali k8s riesce a capire se un servizio è offline o meno.

**Esercizio:** Dall'esercizio precedente inseriamo un Readiness probes e un Liveness probes per verificare quando il deployment di tomcat è disponibile e lo stato di salute. Andiamo a modificare il file .yaml precedente.

```

1  apiVersion: apps/v1beta2
2  kind: Deployment
3  metadata:
4    name: tomcat-deployment
5  spec:
6    selector:
7      matchLabels:
8        app: tomcat
9    replicas: 4
10   template:
11     metadata:
12       labels:
13         app: tomcat
14     spec:
15       containers:
16         - name: tomcat
17           image: tomcat:9.0
18           ports:
19             - containerPort: 8080
20           livenessProbe:
21             httpGet:
22               path: /
23               port: 8080
24             initialDelaySeconds: 30
25             periodSeconds: 30
26           readinessProbe:
27             httpGet:
28               path: /
29               port: 8080
30             initialDelaySeconds: 15
31             periodSeconds: 3

```

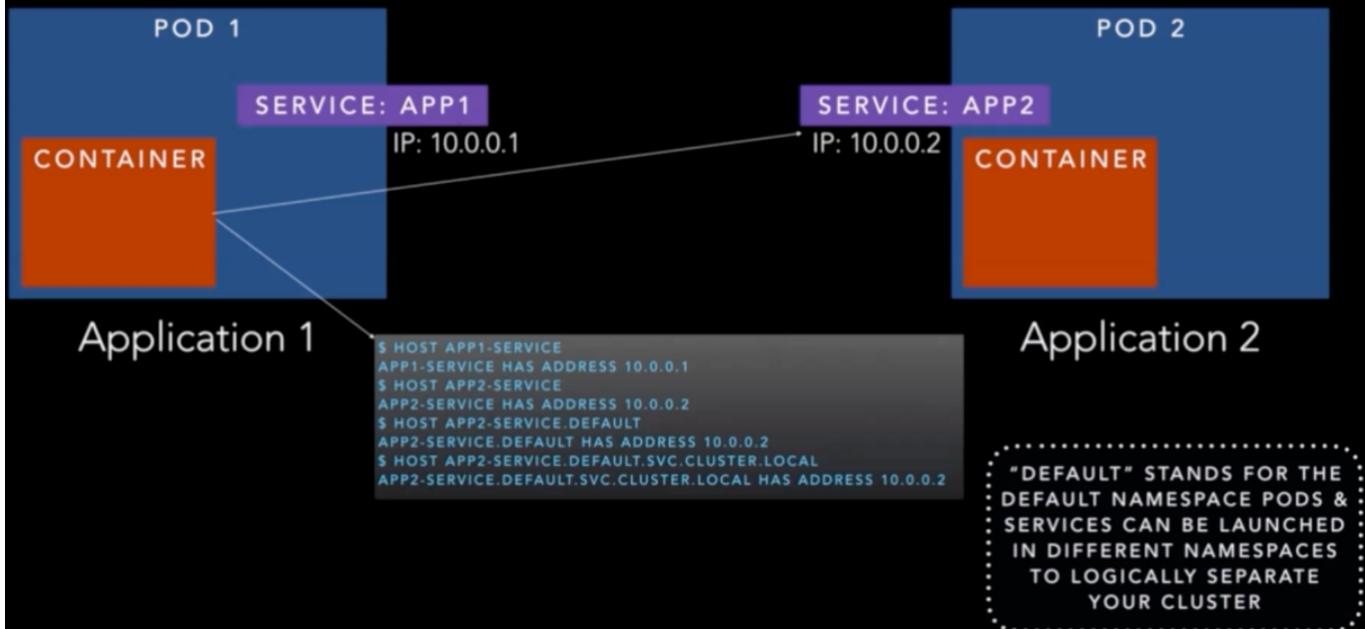
### DNS e Service Discovery:

Il modo in cui k8s realizza l'interconnessione tra i vari microservizi è tramite il proprio DNS. Il DNS ricordiamo che non fa altro che tradurre dei nomi in indirizzi ip. K8s utilizza una nomenclatura nota per decidere qual è il nome del dns:

<b>&lt;my-service-name&gt;.&lt;my-namespace&gt;.svc.cluster.local</b>
General Form of DNS Name for Non-Headless Services

K8s permette di definire i namespaces, e servono a separare un determinato cluster in dei cluster logici più piccoli. Per impostazione predefinita tutto ciò che si fa viene inserito nel namespace di default. Per esempio:

## HOW APP1 COULD REACH APP2 USING DNS



Nell'esempio di sopra abbiamo due applicazioni/microservizi in deploy in due differenti pod, ad ogni pod corrisponde un ip address diverso.

Che succede se il riferimento al pod 2 dovesse cambiare? Dobbiamo utilizzare al posto degli indirizzi ip dei nomi logici. Inseriamo quindi il nome logico definito dal dns del service: app2 e innestiamolo nel deployment del servizio nel pod 1. Di conseguenza avremo i due applicativi sempre connessi tra di loro indipendentemente dalla variazione del loro indirizzo ip.

### Volumes, [1]

K8s utilizza il concetto di volumes per memorizzare le informazioni persistenti dei microservizi, e possono essere dischi fisici o virtuali. I volumi possono essere considerati come delle directory, i cui pod possono avere l'accesso. Possono essere montati contemporaneamente in più pod così da condividere dati.

I pod devono specificare il tipo di volume che hanno bisogno e il path di montaggio:

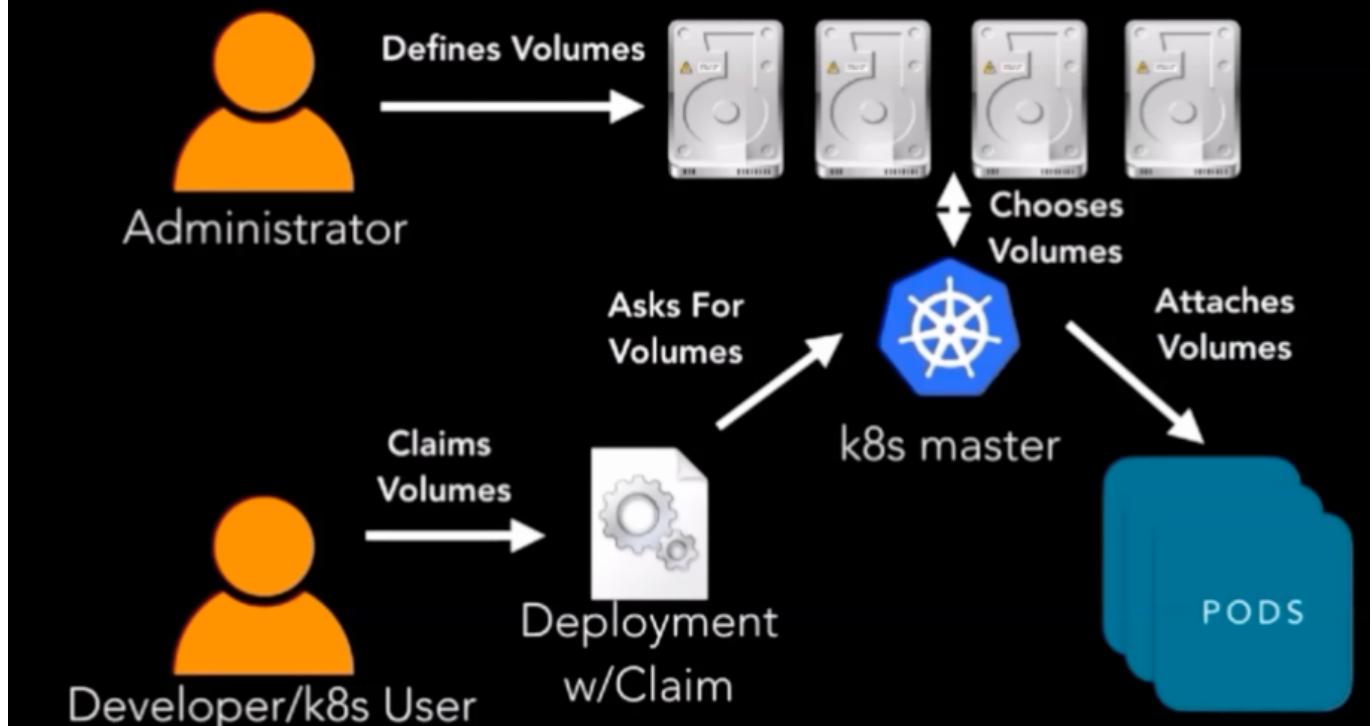
- Usare il campo `spec.volume` per specificare la tipologia di volume voluta;
- Usare il campo `spec.containers.volumeMounts` per specificare il path di montaggio.

In questo modo l'infrastruttura può essere scalata e può vivere separatamente dai dati.

La tipologia di volume maggiormente sviluppata sono i *PersistentVolumes*. Per usarli bisogna:

- Allocare gli spazi di memoria tramite uno o più *PersistentVolumes*.
- Stabilire un *PersistentVolumeClaim*, si tratta di una richiesta di storage da parte di un utente/pod.

# HOW CLAIMS & VOLUMES INTERACT



- Un amministratore definisce degli spazi di memoria/storage fisici o virtuali;
- Lo sviluppatore/utente k8s richiede uno spazio di memoria attraverso dei claim nel file di deployment del pod.
- Il master k8s elabora la richiesta e sceglie i volumi adatti. Monta quindi sui pod i volumi richiesti.

Esempio di creazione di un volume:

```

kind: PersistentVolume
apiVersion: v1
metadata:
  name: task-pv-volume
  labels:
    type: local
spec:
  storageClassName: manual
  capacity:
    storage: 10Gi
  accessModes:
    - ReadWriteOnce
  hostPath:
    path: "/mnt/data"
  
```

[1]

```

kind: PersistentVolumeClaim
apiVersion: v1
metadata:
  name: task-pv-claim
spec:
  storageClassName: manual
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: 3Gi
  
```

[2]

Nell'esempio [1] definiamo un file .yaml inserendo la tipologia in *kind: PersistentVolume*. In metadata definiamo il nome *task-pv-volume* e l'etichetta di tipo locale. Viene assegnata una classe una capacità di 10Gi. Si specifica la modalità di accesso in *accessModes*, in questo caso la lettura e la scrittura sono in [mutua esclusione](#), *ReadWriteOnce*. Infine la directory/path.

Nell'esempio [2] definiamo un file .yaml per i pod che definiscono un *kind: PersistentVolumeClaim*. I pod richiedono la memoria definita precedentemente. Se il master trova un match crea una connessione tra il claim e il volume definito precedentemente.

Eseguiamo il seguente comando per istanziare i volumi:

```
$ kubectl create -f local-volumes.yaml
```

Verifichiamo con:

```
$ kubectl get persistentvolumes
```

## Secrets, [1], [2]

Un Secret è un oggetto che contiene una piccola quantità di dati sensibili, come una password, un token o una chiave. Tali informazioni potrebbero altrimenti essere inserite in una specifica Pod o in un'immagine del contenitore. L'uso di un Secret consente di non includere dati riservati nel codice dell'applicazione.

Poiché i Secret possono essere creati indipendentemente dai Pod che li utilizzano, c'è meno rischio che il Secret (e i suoi dati) vengano esposti durante il flusso di lavoro di creazione, visualizzazione e modifica dei Pod. Kubernetes, e le applicazioni che girano nel cluster, possono anche prendere ulteriori precauzioni con i Secret, come ad esempio evitare di scrivere i dati segreti su uno storage non volatile.

Esistono tre modi principali in cui un Pod può utilizzare un Secret:

- Come file in un volume montato su uno o più dei suoi contenitori.
- Come variabile d'ambiente del contenitore.
- Da parte del kubelet durante l'estrazione delle immagini per il Pod.

### Creare un Secret, [1]

Un segreto può contenere le credenziali utente richieste dai pod per accedere a un database. Ad esempio, una stringa di connessione al database è composta da un nome utente e da una password. È possibile memorizzare il nome utente in un file `./username.txt` e la password in un file `./password.txt` sul computer locale.

```
$ echo -n 'admin' > ./username.txt  
$ echo -n '1f2d1e2e67df' > ./password.txt
```

In questi comandi, il flag `-n` assicura che i file generati non abbiano un carattere newline in più alla fine del testo. Questo è importante perché quando kubectl legge un file e ne codifica il contenuto in una stringa base64, anche il carattere newline extra viene codificato.

Il comando `kubectl create secret` impacchetta questi file in un Secret e crea l'oggetto sul server API.

```
$ kubectl create secret generic db-user-pass \  
--from-file=./username.txt \  
--from-file=./password.txt
```

L'output è simile a questo:

```
$ secret/db-user-pass created
```

Il nome predefinito della chiave è il nome del file. È possibile impostare facoltativamente il nome della chiave usando `--from-file=[key=]source`. Ad esempio:

```
$ kubectl create secret generic db-user-pass \  
--from-file=username=./username.txt \  
--from-file=password=./password.txt
```

È anche possibile fornire dati segreti utilizzando il tag `--from-literal=<key>=<value>`. Questo tag può essere specificato più di una volta per fornire più coppie chiave-valore. Si noti che i caratteri speciali come `$`, `\`, `*`, `=` e `!` saranno interpretati dalla shell e richiederanno un escape.

Nella maggior parte delle shell, il modo più semplice per escludere la password è circondarla con apici singoli (''). Ad esempio, se la password è `S!B\*d$zDsb=`, eseguire il seguente comando:

```
$ kubectl create secret generic db-user-pass \
--from-literal=username=devuser \
--from-literal=password='S!B\*d$zDsb='
```

Possiamo verificare quanto fatto con uno dei due seguenti comandi (sono analoghi):

```
$ kubectl get secrets
```

oppure:

```
$ kubectl describe secrets/db-user-pass
```

I comandi `kubectl get` e `kubectl describe` evitano di mostrare il contenuto di un segreto per impostazione predefinita. Questo per evitare che il segreto venga esposto accidentalmente o che venga memorizzato in un log del terminale.

## Decodifica i segreti, [1]

Per visualizzare il contenuto del Segreto creato, eseguire il seguente comando:

```
$ kubectl get secret db-user-pass -o jsonpath='{.data}'
```

L'output è simile a questo:

```
$ {"password": "MwYyZDF1MmU2N2Rm", "username": "YWRtaW4="}
```

Ora è possibile decodificare il valore della password:

```
$ echo 'MwYyZDF1MmU2N2Rm' | base64 --decode
```

L'output è simile a questo:

```
$ 1f2d1e2e67df
```

Per evitare di memorizzare un valore segreto codificato nella cronologia della shell, è possibile eseguire il seguente comando:

```
$ kubectl get secret db-user-pass -o jsonpath='{.data.password}' | base64 --decode
```

Infine per cancellare i Secret usare il comando:

```
$ kubectl delete secret db-user-pass
```

## Creare il file di configurazione, [1]

È possibile creare prima un segreto in un file, in formato JSON o YAML, e poi creare l'oggetto. La risorsa Secret contiene due mappe: `data` e `stringData`. Il campo `data` è usato per memorizzare dati arbitrari, codificati con base64. Il campo `stringData` è fornito per comodità e consente di fornire dati segreti come stringhe non codificate. Le chiavi di `data` e `stringData` devono essere costituite da caratteri alfanumerici, `-`, `_` o `..`.

Ad esempio, per memorizzare due stringhe in un segreto utilizzando il campo `data`, convertire le stringhe in base64 come segue:

```
$ echo -n 'admin' | base64
```

L'output è simile a questo:

```
$ YWRtaW4=
```

```
$ echo -n '1f2d1e2e67df' | base64
```

L'output è simile a questo:

```
$ MWYyZDFlMmU2N2Rm
```

Scrivere un file di configurazione Secret che assomigli a questo:

```
apiVersion: v1
kind: Secret
metadata:
  name: mysecret
type: Opaque
data:
  username: YWRtaW4=
  password: MWYyZDFlMmU2N2Rm
```

Ora creare il segreto usando `kubectl apply`:

```
$ kubectl apply -f ./secret.yaml
```

L'output è simile a questo:

```
$ secret/mysecret created
```

Per eliminare il segreto creato:

```
$ kubectl delete secret mysecret
```

### Usare i Secret come file da un Pod, [1]

Se si desidera accedere ai dati di un segreto in un Pod, un modo per farlo è fare in modo che Kubernetes renda disponibile il valore di quel segreto come file all'interno del filesystem di uno o più contenitori del Pod.

Per configuararlo, dovete:

1. Creare un secret o utilizzarne uno esistente. Più Pod possono fare riferimento allo stesso secret.
2. Modificare la definizione del Pod per aggiungere un volume sotto `.spec.volumes[]`. Dare al volume un nome qualsiasi e avere un campo `.spec.volumes[].secret.secretName` uguale al nome dell'oggetto Secret.
3. Aggiungere un `.spec.containers[].volumeMounts[]` a ogni contenitore che necessita del segreto. Specificare `.spec.containers[].volumeMounts[].readOnly = true` e `.spec.containers[].volumeMounts[].mountPath` con il nome di una directory inutilizzata in cui si desidera che vengano visualizzati i segreti.
4. Modificare l'immagine o la riga di comando in modo che il programma cerchi i file in quella directory. Ogni chiave nella mappa dei dati segreti diventa il nome del file sotto `mountPath`.

Questo è un esempio di Pod che monta un segreto chiamato *mysecret* in un volume:

```

apiVersion: v1
kind: Pod
metadata:
  name: mypod
spec:
  containers:
    - name: mypod
      image: redis
      volumeMounts:
        - name: foo
          mountPath: "/etc/foo"
          readOnly: true
  volumes:
    - name: foo
      secret:
        secretName: mysecret
        optional: false # default setting; "mysecret" must exist

```

Ogni Secret che si vuole usare deve essere citato in `.spec.volumes`.

Se ci sono più contenitori nel Pod, ogni contenitore ha bisogno del proprio blocco `volumeMounts`, ma è necessario un solo `.spec.volumes` per ogni Secret.

## Usare i Secrets come variabili d'ambiente, [1]

Per usare un Secret come variabile d'ambiente in un Pod:

1. Creare un Secret (o usarne uno esistente). Più Pod possono fare riferimento allo stesso segreto.
2. Modificare la definizione di Pod in ogni contenitore che si desidera consumare il valore di una chiave segreta per aggiungere una variabile d'ambiente per ogni chiave segreta che si desidera consumare. La variabile d'ambiente che consuma la chiave segreta deve inserire il nome e la chiave del segreto in `env[].valueFrom.secretKeyRef`.
3. Modificare l'immagine e/o la riga di comando in modo che il programma cerchi i valori nelle variabili d'ambiente specificate.

Questo è un esempio di Pod che utilizza un segreto tramite variabili d'ambiente:

```

apiVersion: v1
kind: Pod
metadata:
  name: secret-env-pod
spec:
  containers:
  - name: mycontainer
    image: redis
    env:
      - name: SECRET_USERNAME
        valueFrom:
          secretKeyRef:
            name: mysecret
            key: username
            optional: false # same as default; "mysecret" must exist
                          # and include a key named "username"
      - name: SECRET_PASSWORD
        valueFrom:
          secretKeyRef:
            name: mysecret
            key: password
            optional: false # same as default; "mysecret" must exist
                          # and include a key named "password"
  restartPolicy: Never

```

## Utilizzo e monitoraggio delle risorse, [1]

Per scalare un'applicazione e fornire un servizio affidabile, è necessario capire come si comporta l'applicazione quando viene distribuita. È possibile esaminare le prestazioni dell'applicazione in un cluster Kubernetes esaminando i container, i pod, i servizi e le caratteristiche del cluster complessivo. Kubernetes fornisce informazioni dettagliate sull'utilizzo delle risorse di un'applicazione a ciascuno di questi livelli. Queste informazioni consentono di valutare le prestazioni dell'applicazione e di individuare i colli di bottiglia che possono essere rimossi per migliorare le prestazioni complessive.

In Kubernetes, il monitoraggio delle applicazioni non dipende da un'unica soluzione di monitoraggio. Sui nuovi cluster, è possibile utilizzare le metriche delle risorse o le pipeline di metriche complete per raccogliere le statistiche di monitoraggio.

Pipeline di metriche delle risorse:

La pipeline delle metriche delle risorse fornisce un insieme limitato di metriche relative ai componenti del cluster, come il controller Horizontal Pod Autoscaler e l'utilità kubectl top. Queste metriche sono raccolte dal metricserver, leggero, a breve termine e in memoria, ed esposte tramite l'API metrics.k8s.io.

Metrics-server scopre tutti i nodi del cluster e interroga il kubelet di ciascun nodo per l'utilizzo della CPU e della memoria. Il kubelet funge da ponte tra il master Kubernetes e i nodi, gestendo i pod e i container in esecuzione su una macchina. Il kubelet traduce ogni pod nei contenitori che lo compongono e recupera le statistiche di utilizzo dei singoli contenitori dal runtime del contenitore attraverso l'interfaccia del runtime del contenitore. Se si usa un runtime di container che utilizza cgroup e namespace di Linux per implementare i container e il runtime di container non pubblica le statistiche di utilizzo, il kubelet può cercarle direttamente (usando il codice di cAdvisor). Indipendentemente dal modo in cui le statistiche arrivano, il kubelet espone le

statistiche aggregate sull'uso delle risorse dei pod attraverso l'API metrics-server Resource Metrics. Questa API è servita all'indirizzo /metrics/resource/v1beta1 sulle porte autenticate e di sola lettura del kubelet.

### Pipeline di metriche completa:

Una pipeline di metriche completa consente di accedere a metriche più ricche. Kubernetes può rispondere a queste metriche scalando o adattando automaticamente il cluster in base al suo stato attuale, utilizzando meccanismi come l'Horizontal Pod Autoscaler. La pipeline di monitoraggio recupera le metriche dal kubelet e poi le espone a Kubernetes tramite un adattatore, implementando l'API custom.metrics.k8s.io o external.metrics.k8s.io.

[Prometheus](#), un progetto CNCF, può monitorare nativamente Kubernetes, i nodi e Prometheus stesso. Vedi anche [Monitoraggio della salute del nodo](#).

## Namespaces, [1]

In Kubernetes, i namespaces forniscono un meccanismo per isolare gruppi di risorse all'interno di un singolo cluster. I nomi delle risorse devono essere unici all'interno di un namespace, ma non tra i namespaces. Lo scoping basato sui namespace è applicabile solo agli oggetti namespace (ad esempio, Deployments, Services, ecc.) e non agli oggetti a livello di cluster (ad esempio, StorageClass, Nodes, PersistentVolumes, ecc.).

### Quando usare più spazi di nomi:

I Namespace sono destinati all'uso in ambienti con molti utenti distribuiti su più team o progetti. Per i cluster con pochi o decine di utenti, non dovrebbe essere necessario creare o pensare ai namespace. Iniziate a usare gli spazi dei nomi quando avrete bisogno delle loro caratteristiche. I namespace forniscono un ambito per i nomi. I nomi delle risorse devono essere univoci all'interno di uno namespace, ma non in tutti i namespaces. I namespace non possono essere annidati l'uno nell'altro e ogni risorsa Kubernetes può essere presente in un solo namespace. I namespace sono un modo per dividere le risorse del cluster tra più utenti (tramite la [quota di risorse](#)).

Non è necessario utilizzare più spazi dei nomi per separare risorse leggermente diverse, come ad esempio versioni diverse dello stesso software: utilizzate le etichette per distinguere le risorse all'interno dello stesso namespace. Se non specifichiamo un namespace verrà inserito tutto nel namespace di default.

È possibile elencare i namespaces correnti in un cluster utilizzando il comando:

```
$ kubectl get namespace
```

Creare un nuovo file YAML chiamato my-namespace.yaml con i contenuti:

```
apiVersion: v1
kind: Namespace
metadata:
  name: <insert-namespace-name-here>
```

Quindi esegui:

```
$ kubectl create -f ./my-namespace.yaml
```

In alternativa, è possibile creare un namespace utilizzando il seguente comando:

```
$ kubectl create namespace <insert-namespace-name-here>
```

Eliminare un namespace con (Attenzione: Questo cancella tutto ciò che si trova nel namespace!):

```
$ kubectl delete namespaces <insert-some-namespace-name>
```

## Quote di risorse, [1]

Quando diversi utenti o team condividono un cluster con un numero fisso di nodi, si teme che un team possa utilizzare più della sua giusta quota di risorse.

Le quote di risorse sono uno strumento a disposizione degli amministratori per risolvere questo problema.

Una quota di risorse, definita da un oggetto ResourceQuota, fornisce vincoli che limitano il consumo aggregato di risorse per ogni namespace. Può limitare la quantità di oggetti che possono essere creati in un namespace per tipo, nonché la quantità totale di risorse di calcolo che possono essere consumate dalle risorse in quel namespace.

Le quote di risorse funzionano in questo modo:

- Team diversi lavorano in spazi dei nomi diversi. Questo può essere imposto con RBAC.
- L'amministratore crea una ResourceQuota per ogni spazio dei nomi.
- Gli utenti creano risorse (pod, servizi, ecc.) nello spazio dei nomi e il sistema di quote tiene traccia dell'utilizzo per garantire che non superi i limiti di risorse definiti in una ResourceQuota.
- Se la creazione o l'aggiornamento di una risorsa viola un vincolo di quota, la richiesta fallirà con il codice di stato HTTP 403 FORBIDDEN con un messaggio che spiega il vincolo che sarebbe stato violato.
- Se la quota è abilitata in uno spazio dei nomi per le risorse di calcolo come la cpu e la memoria, gli utenti devono specificare le richieste o i limiti per questi valori; altrimenti, il sistema di quote potrebbe rifiutare la creazione di pod. Suggerimento: utilizzare il controllore di ammissione LimitRanger per forzare i valori predefiniti per i pod che non richiedono risorse di calcolo.

Vedere la seguente [guida](#) per un esempio di come evitare questo problema.

Esempi di policy che potrebbero essere create utilizzando namespace e quote sono i seguenti:

- In un cluster con una capacità di 32 GiB di RAM e 16 core, lasciare che il team A usi 20 GiB e 10 core, e che B usi 10GiB e 4 core e che tenere 2GiB e 2 core come riserva per una futura allocazione.
- Limitare il namespace "testing" a 1 core e 1GiB di RAM e lasciare che il namespace "produzione" utilizzi qualsiasi quantità.

Nel caso in cui la capacità totale del cluster sia inferiore alla somma delle quote dei namespace, potrebbe verificarsi una contesa per le risorse. Questo viene gestito in base al principio "primo arrivato, primo servito". Né la contesa né le modifiche alla quota influiscono sulle risorse già create.

## Horizontal Pod Autoscaling, [1]

In Kubernetes, un HorizontalPodAutoscaler aggiorna automaticamente una risorsa del carico di lavoro (come un Deployment o uno StatefulSet), con l'obiettivo di scalare automaticamente il carico di lavoro per soddisfare la domanda.

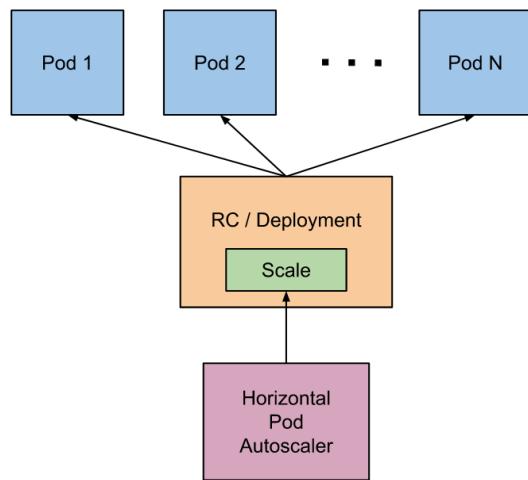
La scaling orizzontale significa che la risposta all'aumento del carico consiste nel deployment di un maggior numero di Pod. Questo è diverso dal vertical scaling, che per Kubernetes significa

assegnare più risorse (ad esempio, memoria o CPU) ai Pod già in esecuzione per il carico di lavoro.

Se il carico diminuisce e il numero di Pod è superiore al minimo configurato, HorizontalPodAutoscaler indica alla risorsa del carico di lavoro (Deployment, StatefulSet o altra risorsa simile) di ridimensionarsi.

L'autoscaling orizzontale dei pod non si applica agli oggetti che non possono essere scalati (ad esempio, un DaemonSet).

HorizontalPodAutoscaler è implementato come risorsa dell'API Kubernetes e come controllore. La risorsa determina il comportamento del controllore. Il controller di autoscaling dei pod orizzontali, in esecuzione all'interno del piano di controllo di Kubernetes, regola periodicamente la scala desiderata del suo target (ad esempio, un Deployment) per adattarsi alle metriche osservate, come l'utilizzo medio della CPU, l'utilizzo medio della memoria o qualsiasi altra metrica personalizzata specificata dall'utente. [\[ESEMPIO\]](#)



## Extra - Esercizio riepilogativo

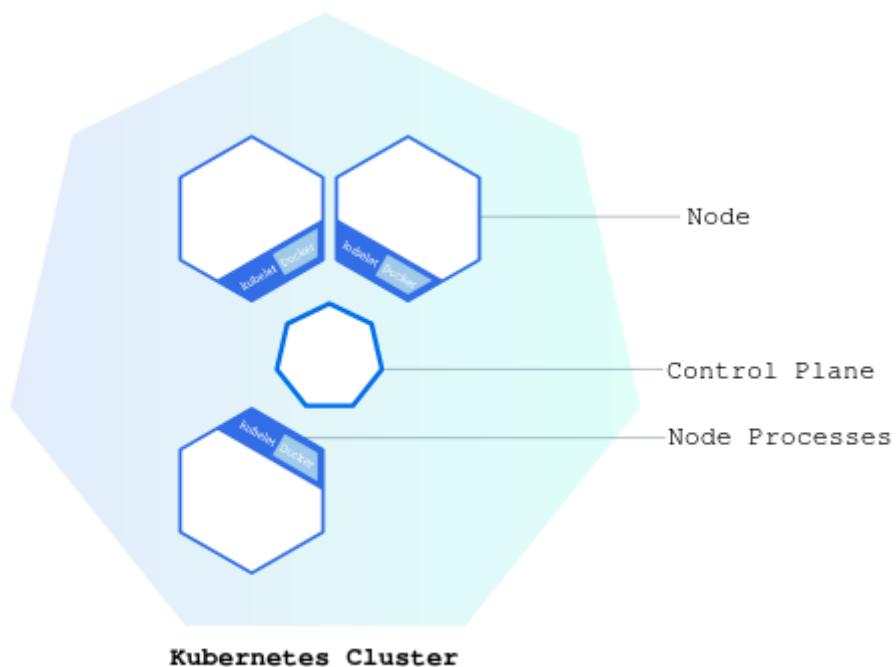
### Usare Minikube per creare un cluster, [1]

#### Cluster Kubernetes:

Kubernetes coordina un cluster ad alta disponibilità di computer collegati per lavorare come un'unica unità. Le astrazioni di Kubernetes consentono di distribuire applicazioni containerizzate in un cluster senza legarle specificamente a singole macchine. Per utilizzare questo nuovo modello di distribuzione, le applicazioni devono essere confezionate in modo da disaccoppiarle dai singoli host: devono essere containerizzate. Le applicazioni containerizzate sono più flessibili e disponibili rispetto ai modelli di distribuzione del passato, in cui le applicazioni venivano installate direttamente su macchine specifiche come pacchetti profondamente integrati nell'host. Kubernetes automatizza la distribuzione e la pianificazione dei container di applicazioni in un cluster in modo più efficiente.

Un cluster Kubernetes è composto da due tipi di risorse:

- Il Control Plane che coordina il cluster;
- I nodi sono i lavoratori che eseguono le applicazioni.



Il Control Plane è responsabile della gestione del cluster coordinando tutte le attività del cluster, come la pianificazione delle applicazioni, il mantenimento dello stato desiderato delle applicazioni, il ridimensionamento delle applicazioni e il roll-out di nuovi aggiornamenti.

Un nodo è una macchina virtuale o un computer fisico che serve come macchina operatrice in un cluster Kubernetes. Ogni nodo ha un Kubelet, che è un agente per gestire il nodo e comunicare con il control plane di Kubernetes. Il nodo deve anche disporre di strumenti per gestire le operazioni dei container, come containerd o Docker. Un cluster Kubernetes che gestisce il traffico di produzione dovrebbe avere un minimo di tre nodi, perché se un nodo si guasta, sia un membro etcd che un'istanza del control plane sarebbero persi e la ridondanza sarebbe compromessa. È possibile ridurre questo rischio aggiungendo altri nodi control plane.

Quando si distribuiscono le applicazioni su Kubernetes, si dice al control plane di avviare i container delle applicazioni. Il control plane pianifica l'esecuzione dei container sui nodi del

cluster. I nodi comunicano con il control plane utilizzando l'API di Kubernetes, che il control plane espone. Gli utenti finali possono anche utilizzare direttamente l'API Kubernetes per interagire con il cluster.

Un cluster Kubernetes può essere distribuito su macchine fisiche o virtuali. Per iniziare a sviluppare Kubernetes, si può usare Minikube. Minikube è un'implementazione leggera di Kubernetes che crea una macchina virtuale sulla vostra macchina locale e distribuisce un semplice cluster contenente un solo nodo. Minikube è disponibile per sistemi Linux, macOS e Windows. La CLI di Minikube fornisce le operazioni di avvio di base per lavorare con il cluster, tra cui avvio, arresto, stato e cancellazione. Per questa esercitazione, tuttavia, si utilizzerà un terminale online fornito con Minikube preinstallato.

### **Creare un cluster Kubernetes:**

Verificare prima se minikube sia correttamente installato tramite il comando:

```
$ minikube version
```

Verificato che sia correttamente installato possiamo avviare il cluster, eseguendo il comando:

```
$ minikube start
```

Ora abbiamo un cluster Kubernetes in esecuzione nel nostro terminale online. Minikube ha avviato una macchina virtuale per noi e un cluster Kubernetes è ora in esecuzione in quella macchina virtuale.

Per verificare se kubectl è installato, si può eseguire il comando:

```
$ kubectl version
```

Adesso kubectl è configurato e possiamo vedere sia la versione del client che quella del server. La versione del client è la versione di kubectl; la versione del server è la versione di Kubernetes installata sul master.

Vediamo i dettagli del cluster:

```
$ kubectl cluster-info
```

Per visualizzare i nodi del cluster, eseguire il comando:

```
$ kubectl get nodes
```

Questo comando mostra tutti i nodi che possono essere utilizzati per ospitare le nostre applicazioni. Ora abbiamo un solo nodo e possiamo vedere che il suo stato è pronto (è pronto ad accettare applicazioni per la distribuzione).

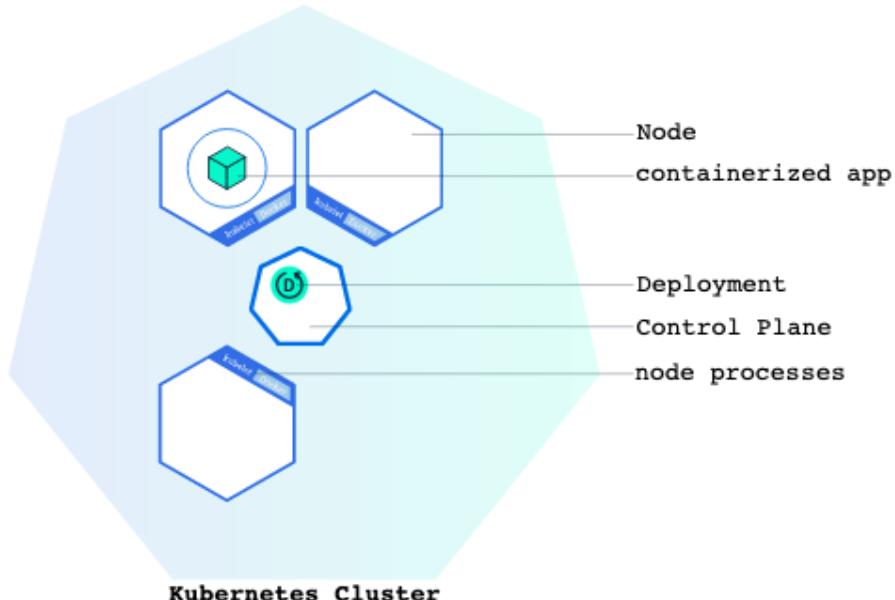
### **Usare di kubectl per creare un deployment, [1]**

#### **Distribuzioni Kubernetes:**

Una volta che si dispone di un cluster Kubernetes funzionante, è possibile distribuire le applicazioni containerizzate su di esso. Per farlo, si crea una configurazione di deployment Kubernetes. L'installazione client indica a Kubernetes come creare e aggiornare le istanze della vostra applicazione. Una volta creato un Deployment, il control plane di Kubernetes pianifica l'esecuzione delle istanze dell'applicazione incluse in quel Deployment sui singoli nodi del cluster.

Una volta create le istanze dell'applicazione, il controller di distribuzione Kubernetes monitora continuamente le istanze. Se il nodo che ospita un'istanza si spegne o viene eliminato, il controller di deployment sostituisce l'istanza con un'altra su un altro nodo del cluster. In questo

modo si ottiene un meccanismo di self-healing per affrontare i guasti o la manutenzione della macchina.



Quando si crea un Deployment, è necessario specificare l'immagine del contenitore per l'applicazione e il numero di repliche che si desidera eseguire. È possibile modificare queste informazioni in seguito aggiornando l'installazione client.

### Deploy di un'app:

Distribuiamo la nostra prima applicazione su Kubernetes con il comando `kubectl create deployment`. Dobbiamo fornire il nome del deployment e la posizione dell'immagine dell'app (includere l'url completo del repository per le immagini ospitate al di fuori dell'hub Docker).

```
$ kubectl create deployment kubernetes-bootcamp  
--image=gcr.io/google-samples/kubernetes-bootcamp:v1
```

Abbiamo appena distribuito la nostra prima applicazione creando un deployment. Questo ha eseguito alcune operazioni:

- ha cercato un nodo adatto per l'esecuzione di un'istanza dell'applicazione (abbiamo solo un nodo disponibile);
- ha pianificato l'esecuzione dell'applicazione su quel nodo;
- configurato il cluster per riprogrammare l'istanza su un nuovo nodo quando necessario.

Per elencare le distribuzioni, utilizzare il comando:

```
kubectl get deployments
```

Vediamo che c'è una distribuzione che esegue una singola istanza della nostra applicazione. L'istanza è in esecuzione all'interno di un contenitore Docker sul proprio nodo.

I pod in esecuzione all'interno di Kubernetes vengono eseguiti su una rete privata e isolata. Per impostazione predefinita, sono visibili da altri pod e servizi all'interno dello stesso cluster Kubernetes, ma non al di fuori della rete. Quando utilizziamo `kubectl`, interagiamo attraverso un endpoint API per comunicare con la nostra applicazione.

Il comando `kubectl` può creare un proxy che inoltra le comunicazioni nella rete privata del cluster. Il proxy può essere terminato premendo control-C e non mostrerà alcun output mentre è in esecuzione.

Apriamo una seconda finestra di terminale per eseguire il proxy:

```
$ echo -e "\n\n\n\033[92mStarting Proxy. After starting it will not output a response. Please click the first Terminal Tab\n";  
kubectl proxy
```

Ora abbiamo una connessione tra il nostro host (il terminale online) e il cluster Kubernetes. Il proxy consente l'accesso diretto alle API da questi terminali.

È possibile vedere tutte le API ospitate attraverso l'endpoint del proxy. Ad esempio, possiamo interrogare la versione direttamente attraverso l'API, utilizzando il comando curl:

```
$ curl http://localhost:8001/version
```

Il server API creerà automaticamente un endpoint per ogni pod, basato sul nome del pod, accessibile anche attraverso il proxy.

Per prima cosa dobbiamo ottenere il nome del pod, che memorizzeremo nella variabile d'ambiente POD\_NAME:

```
$ export POD_NAME=$(kubectl get pods -o go-template --template '{{range .items}}{{.metadata.name}}\n{{end}}')  
echo Name of the Pod: $POD_NAME
```

È possibile accedere al Pod attraverso l'API eseguendo:

```
$ curl http://localhost:8001/api/v1/namespaces/default/pods/$POD_NAME/
```

Affinché la nuova distribuzione sia accessibile senza usare il Proxy, è necessario un Servizio che verrà spiegato successivamente.

## Visualizzazione di pod e nodi, [1]

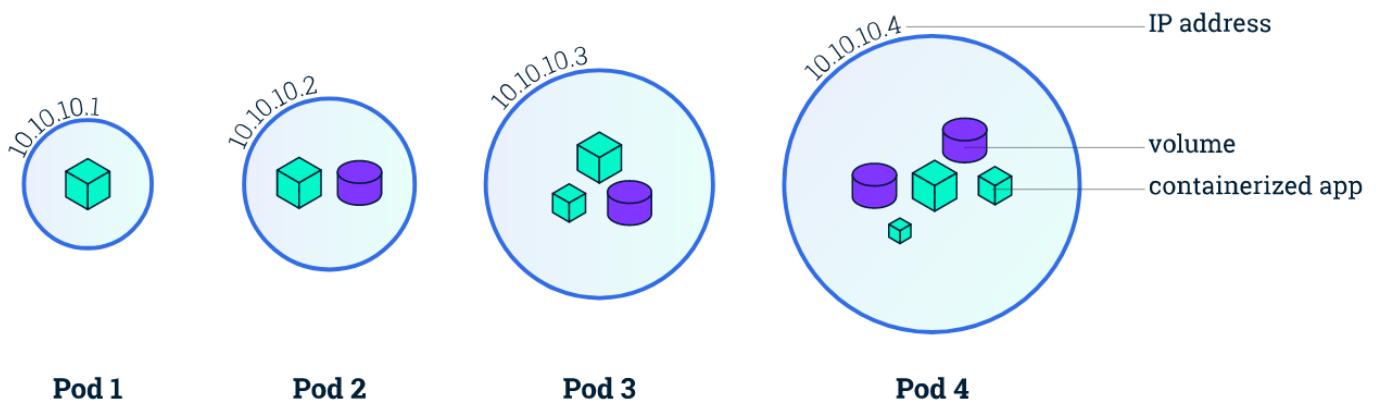
### Kubernetes Pods:

In precedenza quando si è creato un deployment, Kubernetes ha creato un Pod per ospitare l'istanza dell'applicazione. Un Pod è un'astrazione di Kubernetes che rappresenta un gruppo di uno o più container di applicazioni (come Docker) e alcune risorse condivise per questi container. Tali risorse includono:

- Storage condiviso, come volumi;
- Networking, come indirizzo IP unico del cluster;
- Informazioni su come eseguire ciascun container, come la versione dell'immagine del container o le porte specifiche da utilizzare.

Un Pod modella un "host logico" specifico di un'applicazione e può contenere diversi container di applicazioni che sono accoppiati in modo relativamente stretto. Per esempio, un Pod può includere sia il contenitore con l'applicazione Node.js, sia un altro contenitore che alimenta i dati che devono essere pubblicati dal server web Node.js. I contenitori in un Pod condividono un indirizzo IP e uno spazio di porta, sono sempre co-localizzati e co-schedulati ed eseguono in un contesto condiviso sullo stesso Nodo.

I Pod sono l'unità atomica della piattaforma Kubernetes. Quando creiamo una distribuzione su Kubernetes, questa crea Pod con container al loro interno (invece di creare direttamente i container). Ogni Pod è legato al Nodo in cui è pianificato e vi rimane fino alla sua terminazione (secondo la politica di riavvio) o cancellazione. In caso di guasto di un nodo, Pod identici vengono pianificati su altri nodi disponibili nel cluster.



**Pod 1**

**Pod 2**

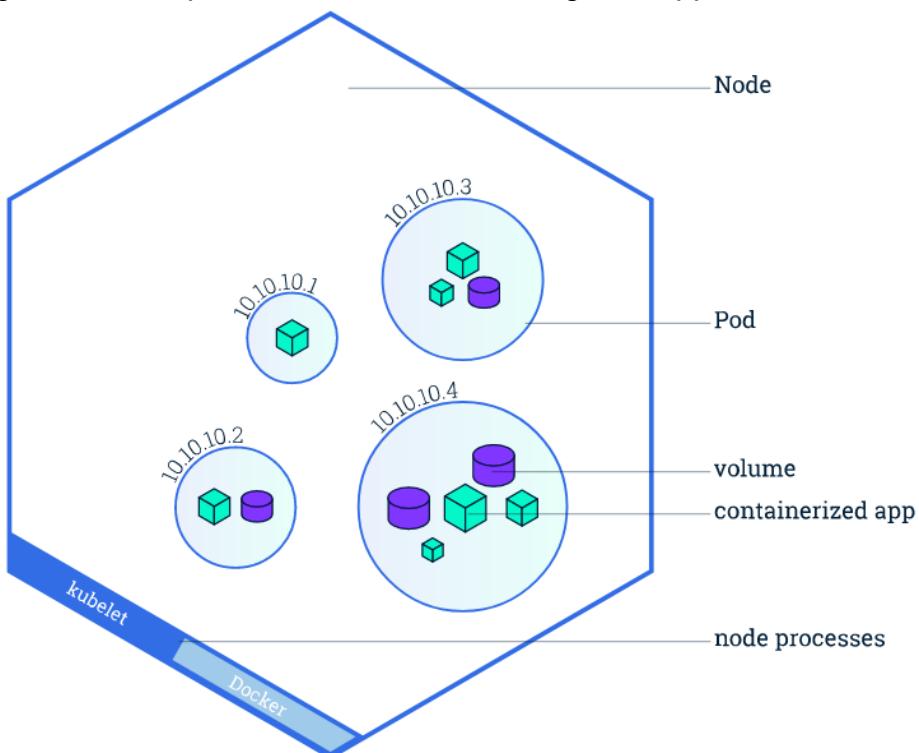
**Pod 3**

**Pod 4**

### Nodi:

Un Pod viene sempre eseguito su un Nodo. Un nodo è una macchina worker in Kubernetes e può essere una macchina virtuale o fisica, a seconda del cluster. Ogni nodo è gestito dal control plane. Un nodo può avere più pod e il control plane di Kubernetes gestisce automaticamente la pianificazione dei pod tra i nodi del cluster. La pianificazione automatica del control plane tiene conto delle risorse disponibili su ogni nodo. Ogni nodo Kubernetes esegue almeno:

- Kubelet, un processo responsabile della comunicazione tra il control plane di Kubernetes e il nodo; gestisce i pod e i container in esecuzione su una macchina.
- Un runtime di container (come Docker) responsabile di estrarre l'immagine del container da un registro, decomprimere il container ed eseguire l'applicazione.



### Step 1 - Controllare la configurazione dell'applicazione:

Verifichiamo che l'applicazione distribuita nello scenario precedente sia in esecuzione. Usiamo il comando kubectl get per cercare i Pod esistenti:

```
$ kubectl get pods
```

Poi, per vedere quali contenitori sono all'interno di quel Pod e quali immagini sono usate per costruire quei contenitori, si esegue il comando describe pods:

```
$ kubectl describe pods
```

Qui vengono visualizzati i dettagli sul contenitore del Pod: Indirizzo IP, porte utilizzate e un elenco di eventi relativi al ciclo di vita del Pod.

Nota: il comando describe può essere usato per ottenere informazioni dettagliate sulla maggior parte delle primitive di kubernetes: nodi, pod, distribuzioni.

### **Step 2 - Mostrare l'applicazione nel terminale:**

Ricordiamo che i Pod sono in esecuzione in una rete isolata e privata, per cui dobbiamo fornire un proxy per l'accesso ad essi, in modo da poter eseguire il debug e interagire con essi. Per farlo, useremo il comando kubectl proxy per eseguire un proxy in una seconda finestra di terminale. Fate clic sul comando sottostante per aprire automaticamente un nuovo terminale ed eseguire il proxy:

```
$ echo -e "\n\n\n\ne[92mStarting Proxy. After starting it will not output a response. Please click the first Terminal Tab\n"; kubectl proxy
```

Ora, di nuovo, otterremo il nome del Pod e lo interrogheremo direttamente attraverso il proxy. Per ottenere il nome del pod e memorizzarlo nella variabile d'ambiente POD\_NAME:

```
$ export POD_NAME=$(kubectl get pods -o go-template --template '{{range .items}}{{.metadata.name}}{{$"\n"}}{{end}}')
```

```
echo Name of the Pod: $POD_NAME
```

Per vedere l'output della nostra applicazione, eseguire una richiesta curl:

```
$ curl  
http://localhost:8001/api/v1/namespaces/default/pods/$POD_NAME/proxy/
```

L'url è il percorso dell'API del Pod.

### **Step 3 - Visualizzare i registri del container:**

Tutto ciò che l'applicazione normalmente invia a STDOUT diventa un log per il contenitore all'interno del Pod. È possibile recuperare questi log con il comando kubectl logs:

```
$ kubectl logs $POD_NAME
```

Nota: non è necessario specificare il nome del container, perché abbiamo un solo container all'interno del pod.

### **Step 4 - Esecuzione del comando sul container:**

Possiamo eseguire i comandi direttamente sul container, una volta che il Pod è attivo e funzionante. A tale scopo, si usa il comando exec e si utilizza il nome del Pod come parametro. Elenchiamo le variabili d'ambiente:

```
$ kubectl exec $POD_NAME -- env
```

Vale la pena ricordare che il nome del contenitore stesso può essere omesso, dato che abbiamo un solo contenitore nel Pod.

Avviamo quindi una sessione di bash nel contenitore del Pod:

```
$ kubectl exec -ti $POD_NAME -- bash
```

Ora abbiamo una console aperta sul container dove eseguiamo la nostra applicazione NodeJS. Il codice sorgente dell'applicazione si trova nel file server.js:

```
$ cat server.js
```

È possibile verificare che l'applicazione sia attiva eseguendo il comando curl:

```
curl localhost:8080
```

Nota: qui abbiamo usato localhost perché abbiamo eseguito il comando all'interno del Pod NodeJS. Se non si riesce a connettersi a localhost:8080, verificare di aver eseguito il comando kubectl exec e di aver lanciato il comando dall'interno del Pod.

Per chiudere la connessione al contenitore, digitare `exit`.

## Usare un servizio per esporre l'applicazione, [1]

### Panoramica dei servizi Kubernetes:

I Pod di Kubernetes sono mortali. I Pod hanno infatti un ciclo di vita. Quando un nodo worker muore, anche i Pod in esecuzione sul nodo vengono persi. Un ReplicaSet può quindi riportare dinamicamente il cluster allo stato desiderato attraverso la creazione di nuovi Pod per mantenere l'applicazione in esecuzione. Come altro esempio, si consideri un backend di elaborazione delle immagini con 3 repliche. Queste repliche sono scambiabili; il sistema front-end non dovrebbe preoccuparsi delle repliche del backend o anche se un Pod viene perso e ricreato. Detto questo, ogni Pod in un cluster Kubernetes ha un indirizzo IP univoco, anche i Pod sullo stesso Nodo, quindi è necessario un modo per riconciliare automaticamente le modifiche tra i Pod in modo che le applicazioni continuino a funzionare.

Un servizio in Kubernetes è un'astrazione che definisce un insieme logico di Pod e una politica di accesso ad essi. I servizi consentono un accoppiamento lasso tra Pod dipendenti. Un servizio viene definito utilizzando YAML (preferibile) o JSON, come tutti gli oggetti di Kubernetes. L'insieme dei Pod a cui si rivolge un Servizio è solitamente determinato da un LabelSelector (si veda sotto per capire perché si potrebbe volere un Servizio senza includere il selettore nelle specifiche).

Sebbene ogni Pod abbia un indirizzo IP unico, questi IP non sono esposti all'esterno del cluster senza un Servizio. I servizi consentono alle applicazioni di ricevere traffico. I servizi possono essere esposti in modi diversi, specificando un tipo nel ServiceSpec:

- ClusterIP (predefinito) - Espone il servizio su un IP interno al cluster. Questo tipo rende il servizio raggiungibile solo dall'interno del cluster.
- NodePort - Espone il servizio sulla stessa porta di ogni nodo selezionato nel cluster utilizzando NAT. Rende un servizio accessibile dall'esterno del cluster utilizzando `<NodeIP>:<NodePort>`. È un sottoinsieme di ClusterIP.
- LoadBalancer - Crea un bilanciatore di carico esterno nel cloud corrente (se supportato) e assegna un IP esterno fisso al servizio. Superset di NodePort.
- ExternalName - Mappa il servizio al contenuto del campo externalName (ad esempio, `foo.bar.example.com`), restituendo un record CNAME con il suo valore. Non viene impostato alcun tipo di proxy. Questo tipo richiede la versione v1.7 o superiore di kube-dns o CoreDNS versione 0.0.8 o superiore.

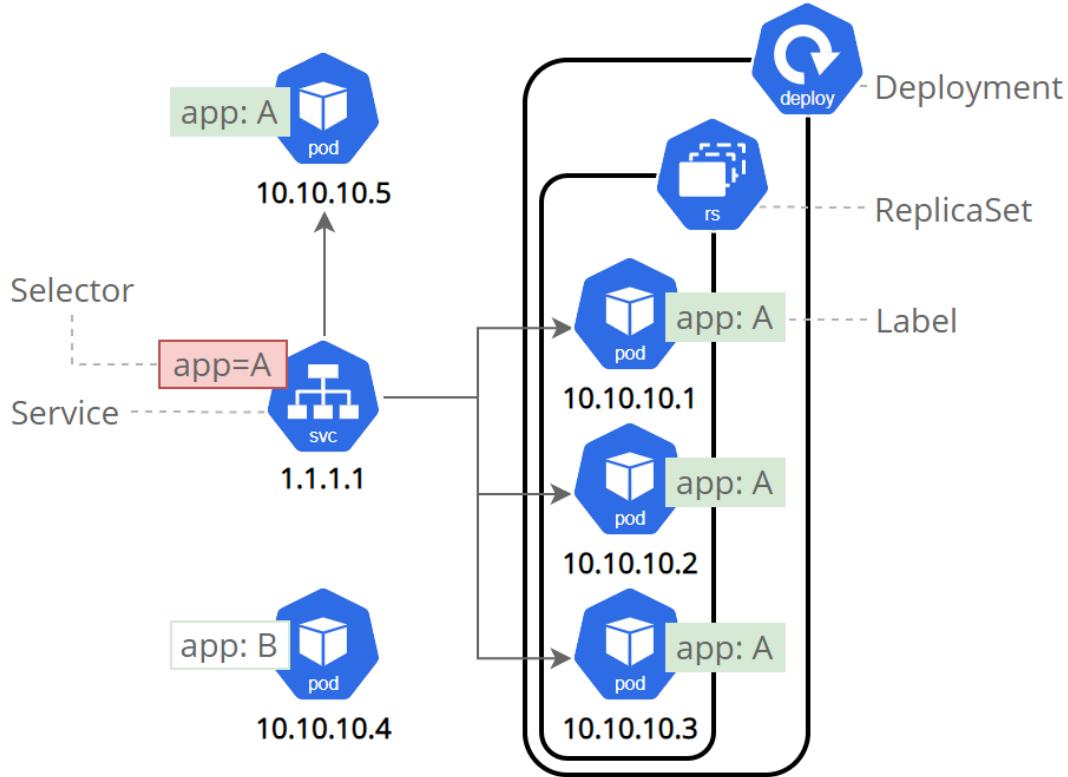
### Servizi ed etichette:

Un servizio instrada il traffico attraverso un insieme di pod. I servizi sono l'astrazione che consente ai pod di morire e replicarsi in Kubernetes senza impattare l'applicazione. Il rilevamento e l'instradamento tra Pod dipendenti (come i componenti frontend e backend di un'applicazione) sono gestiti dai servizi Kubernetes.

I servizi abbinano un insieme di Pod utilizzando etichette e selettori, una primitiva di raggruppamento che consente di effettuare operazioni logiche sugli oggetti in Kubernetes. Le etichette sono coppie chiave/valore collegate agli oggetti e possono essere utilizzate in vari

modi:

- Designare gli oggetti per lo sviluppo, il test e la produzione;
- Incorporare tag di versione;
- Classificare un oggetto utilizzando i tag.



Le etichette possono essere applicate agli oggetti al momento della creazione o in un secondo momento. Possono essere modificate in qualsiasi momento. Esponiamo ora la nostra applicazione utilizzando un servizio e applichiamo alcune etichette.

### Step 1 - Creare un nuovo servizio

Verifichiamo che la nostra applicazione sia in esecuzione. Useremo il comando kubectl get per cercare i Pod esistenti:

```
$ kubectl get pods
```

Quindi, elenchiamo i servizi correnti del nostro cluster:

```
$ kubectl get services
```

Abbiamo un servizio chiamato kubernetes che viene creato di default quando minikube avvia il cluster. Per creare un nuovo servizio ed esporlo al traffico esterno useremo il comando expose con NodePort come parametro.

```
$ kubectl expose deployment/kubernetes-bootcamp --type="NodePort" --port 8080
```

Eseguiamo nuovamente il comando get services:

```
kubectl get services
```

Ora abbiamo un servizio in esecuzione chiamato kubernetes-bootcamp. Qui vediamo che il servizio ha ricevuto un unico cluster-IP, una porta interna e un external-IP (l'IP del nodo).

Per scoprire quale porta è stata aperta esternamente (dall'opzione NodePort), eseguiremo il comando describe service:

```
$ kubectl describe services/kubernetes-bootcamp
```

Adesso creiamo una variabile d'ambiente chiamata NODE\_PORT con il valore della porta del nodo assegnato:

```
$ export NODE_PORT=$(kubectl get services/kubernetes-bootcamp -o go-template='{{(index .spec.ports 0).nodePort}}')  
echo NODE_PORT=$NODE_PORT
```

Ora possiamo verificare che l'applicazione sia esposta all'esterno del cluster usando curl, l'IP del nodo e la porta esposta all'esterno:

```
$ curl $(minikube ip):$NODE_PORT
```

E otteniamo una risposta dal server. Il servizio è esposto.

## Step 2 - Uso delle etichette

Il deployment ha creato automaticamente un'etichetta per il nostro Pod. Con il comando describe deployment si può vedere il nome dell'etichetta:

```
$ kubectl describe deployment
```

Usiamo questa etichetta per interrogare il nostro elenco di Pod. Useremo il comando kubectl get pods con -l come parametro, seguito dai valori dell'etichetta:

```
$ kubectl get pods -l app=kubernetes-bootcamp
```

È possibile fare lo stesso per elencare i servizi esistenti:

```
$ kubectl get services -l app=kubernetes-bootcamp
```

Si ottiene il nome del Pod e lo si memorizza nella variabile d'ambiente POD\_NAME:

```
$ export POD_NAME=$(kubectl get pods -o go-template --template '{{range .items}}{{.metadata.name}}\n{{end}}')  
echo Name of the Pod: $POD_NAME
```

Per applicare una nuova etichetta si usa il comando label seguito dal tipo di oggetto, dal nome dell'oggetto e dalla nuova etichetta:

```
$ kubectl label pods $POD_NAME version=v1
```

Questo applicherà una nuova etichetta al nostro Pod (abbiamo appuntato la versione dell'applicazione nel Pod) e potremo verificarla con il comando describe pod:

```
$ kubectl describe pods $POD_NAME
```

Vediamo che ora l'etichetta è collegata al nostro Pod. E ora possiamo interrogare l'elenco dei pod usando la nuova etichetta:

```
$ kubectl get pods -l version=v1
```

E vediamo il Pod.

## Step 3 - Eliminare un servizio

Per eliminare i servizi si può usare il comando delete service. Le etichette possono essere utilizzate anche in questo caso:

```
$ kubectl delete service -l app=kubernetes-bootcamp
```

Confermare che il servizio è stato eliminato:

```
$ kubectl get services
```

Questo conferma che il nostro servizio è stato rimosso. Per confermare che la route non è più esposta, è possibile eseguire il comando curl dell'IP e della porta precedentemente esposti:

```
$ curl $(minikube ip):$NODE_PORT
```

Questo dimostra che l'applicazione non è più raggiungibile dall'esterno del cluster. È possibile confermare che l'applicazione è ancora in esecuzione con un curl all'interno del pod:

```
$ kubectl exec -ti $POD_NAME -- curl localhost:8080
```

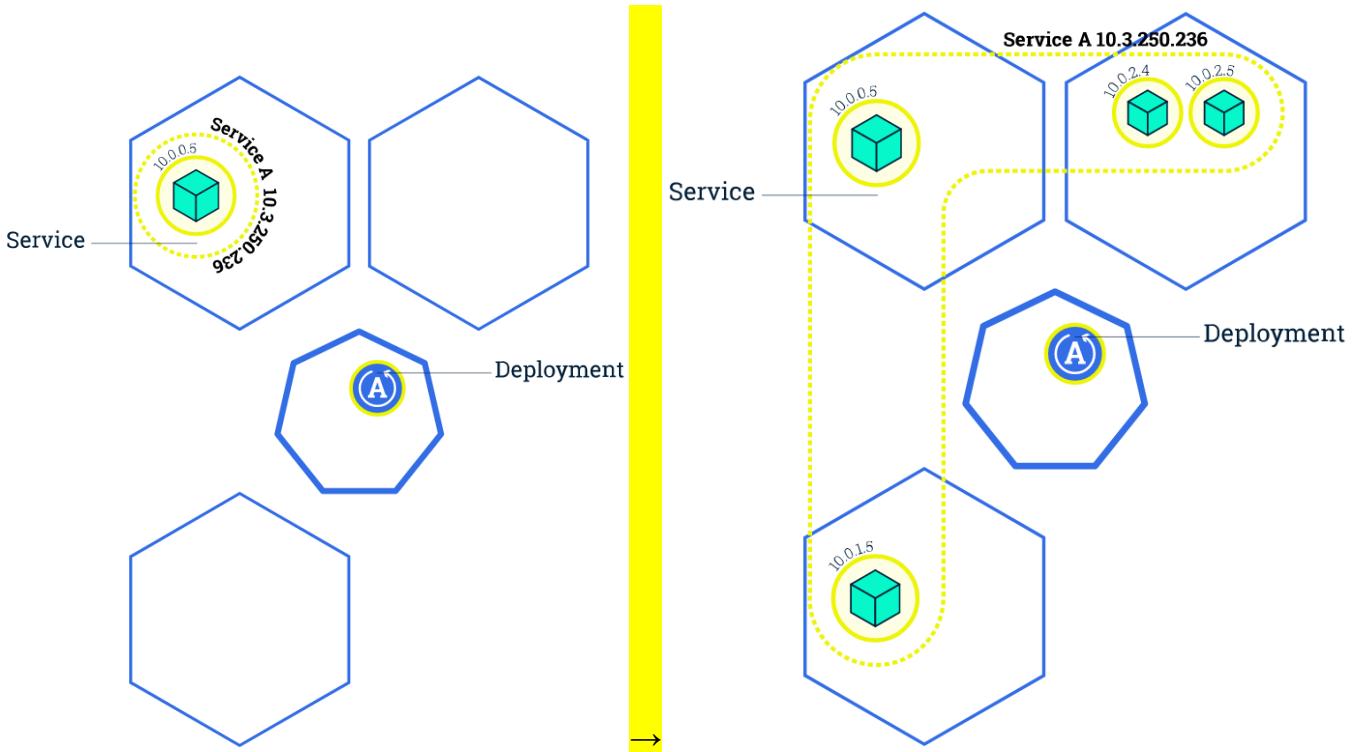
Possiamo notare che l'applicazione è attiva. Questo perché il Deployment sta gestendo l'applicazione. Per chiudere l'applicazione, è necessario eliminare anche l'installazione client.

## Esecuzione di più istanze dell'applicazione, [1]

Scalare un'applicazione

Nei moduli precedenti abbiamo creato un deployment e poi lo abbiamo esposto pubblicamente tramite un Service. Il deployment ha creato un solo Pod per l'esecuzione della nostra applicazione. Quando il traffico aumenterà, dovremo scalare l'applicazione per tenere il passo con la domanda degli utenti.

Il ridimensionamento si ottiene cambiando il numero di repliche in un deployment.



Lo Scaling di un Deployment garantirà la creazione di nuovi Pod e la loro pianificazione sui nodi con risorse disponibili. Lo scaling aumenterà il numero di Pod fino al nuovo stato desiderato. È possibile anche scalare a zero, terminando tutti i Pod del deployment specificato.

L'esecuzione di più istanze di un'applicazione richiede un modo per distribuire il traffico a tutte le istanze. I servizi hanno un bilanciatore di carico integrato che distribuisce il traffico di rete a tutti i Pod di un'installazione client esposta. I servizi monitorano continuamente i Pod in esecuzione tramite endpoint, per garantire che il traffico sia inviato solo ai Pod disponibili.

La scalabilità si ottiene modificando il numero di repliche in un'installazione client.

Una volta che si hanno più istanze di un'applicazione in esecuzione, si è in grado di eseguire aggiornamenti continui senza tempi di inattività. Ora verifichiamo il tutto con un esempio pratico.

## Step 1 - Scalare un'installazione client

Per elencare i deployment, utilizzare il comando:

```
$ kubectl get deployments
```

L'output è simile a questo:

```
$ kubectl get deployments
NAME             READY   UP-TO-DATE   AVAILABLE   AGE
kubernetes-bootcamp   1/1     1           1          6m2s
```

Dovremmo avere 1 Pod. In caso contrario, eseguire nuovamente il comando. Quest'ultimo visualizza:

- NAME - elenca i nomi dei deployment nel cluster;
- READY - mostra il rapporto tra le repliche CORRENTI e quelle DESIDERATE;
- UP-TO-DATE - mostra il numero di repliche che sono state aggiornate per raggiungere lo stato desiderato;
- AVAILABLE - mostra quante repliche dell'applicazione sono disponibili per gli utenti;
- AGE - indica la quantità di tempo in cui l'applicazione è stata in esecuzione.

Per vedere il ReplicaSet creato dal deployment, eseguire:

```
$ kubectl get rs
```

Si noti che il nome del ReplicaSet è sempre formattato come [DEPLOYMENT-NAME]-[RANDOM-STRING]. La stringa casuale è generata in modo casuale e utilizza il pod-template-hash come seed.

Due colonne importanti di questo comando sono:

- DESIRED - visualizza il numero desiderato di repliche dell'applicazione, definito al momento della creazione dell'installazione client. Questo è lo stato desiderato;
- CURRENT - mostra quante repliche sono attualmente in esecuzione.

Quindi, scaliamo il Deployment a 4 repliche. Utilizzeremo il comando kubectl scale, seguito dal tipo di deployment, dal nome e dal numero di istanze desiderate:

```
$ kubectl scale deployments/kubernetes-bootcamp --replicas=4
```

Per elencare nuovamente i deployment, usare:

```
$ kubectl get deployments
```

La modifica è stata applicata e sono disponibili 4 istanze dell'applicazione:

```
$ kubectl get deployments
NAME             READY   UP-TO-DATE   AVAILABLE   AGE
kubernetes-bootcamp   4/4     4           4          14m
```

Controlliamo poi se il numero di Pod è cambiato:

```
$ kubectl get pods -o wide
```

Ora ci sono 4 Pod, con indirizzi IP diversi. La modifica è stata riportata nel registro degli eventi di Deployment. Per verificarlo, usare il comando describe:

```
$ kubectl describe deployments/kubernetes-bootcamp
```

Nell'output di questo comando si può vedere che ora ci sono 4 repliche:

Events:					
Type	Reason	Age	From	Message	
Normal	ScalingReplicaSet	16m	deployment-controller	Scaled up replica set kubernetes-bootcamp-fb5c67579 to 1	
Normal	ScalingReplicaSet	3m17s	deployment-controller	Scaled up replica set kubernetes-bootcamp-fb5c67579 to 4	

## Step 2 - bilanciamento del carico

Verifichiamo se il servizio sta bilanciando il traffico. Per scoprire l'IP e la porta esposti, possiamo usare il servizio describe, come abbiamo imparato nel modulo precedente:

```
$ kubectl describe services/kubernetes-bootcamp
```

Creare una variabile d'ambiente chiamata NODE\_PORT che abbia il valore di Node port:

```
$ export NODE_PORT=$(kubectl get services/kubernetes-bootcamp -o go-template='{{(index .spec.ports 0).nodePort}}')
echo NODE_PORT=$NODE_PORT
```

Successivamente, eseguiremo un curl verso l'IP e la porta esposti. Eseguire il comando più volte:

```
$ curl $(minikube ip):$NODE_PORT
```

Ad ogni richiesta viene assegnato un Pod diverso. Questo dimostra che il bilanciamento del carico funziona:

```
$ curl $(minikube ip):$NODE_PORT
Hello Kubernetes bootcamp! | Running on: kubernetes-bootcamp-fb5c67579-1vpt7 | v=1
$ curl $(minikube ip):$NODE_PORT
Hello Kubernetes bootcamp! | Running on: kubernetes-bootcamp-fb5c67579-whgcb | v=1
$ curl $(minikube ip):$NODE_PORT
Hello Kubernetes bootcamp! | Running on: kubernetes-bootcamp-fb5c67579-plx1j | v=1
$ curl $(minikube ip):$NODE_PORT
Hello Kubernetes bootcamp! | Running on: kubernetes-bootcamp-fb5c67579-plx1j | v=1
$ curl $(minikube ip):$NODE_PORT
Hello Kubernetes bootcamp! | Running on: kubernetes-bootcamp-fb5c67579-plx1j | v=1
$ curl $(minikube ip):$NODE_PORT
Hello Kubernetes bootcamp! | Running on: kubernetes-bootcamp-fb5c67579-plx1j | v=1
$ curl $(minikube ip):$NODE_PORT
Hello Kubernetes bootcamp! | Running on: kubernetes-bootcamp-fb5c67579-whgcb | v=1
```

## Step 3 - Scale Down (Ridimensionamento)

Per ridimensionare il servizio a 2 repliche, eseguire nuovamente il comando scale:

```
$ kubectl scale deployments/kubernetes-bootcamp --replicas=2
```

Elencare i deployment per verificare se la modifica è stata applicata con il comando:

```
$ kubectl get deployments
```

I numero di repliche è sceso a 2:

NAME	READY	UP-TO-DATE	AVAILABLE	AGE
kubernetes-bootcamp	2/2	2	2	24m

Elencare il numero di pod, con:

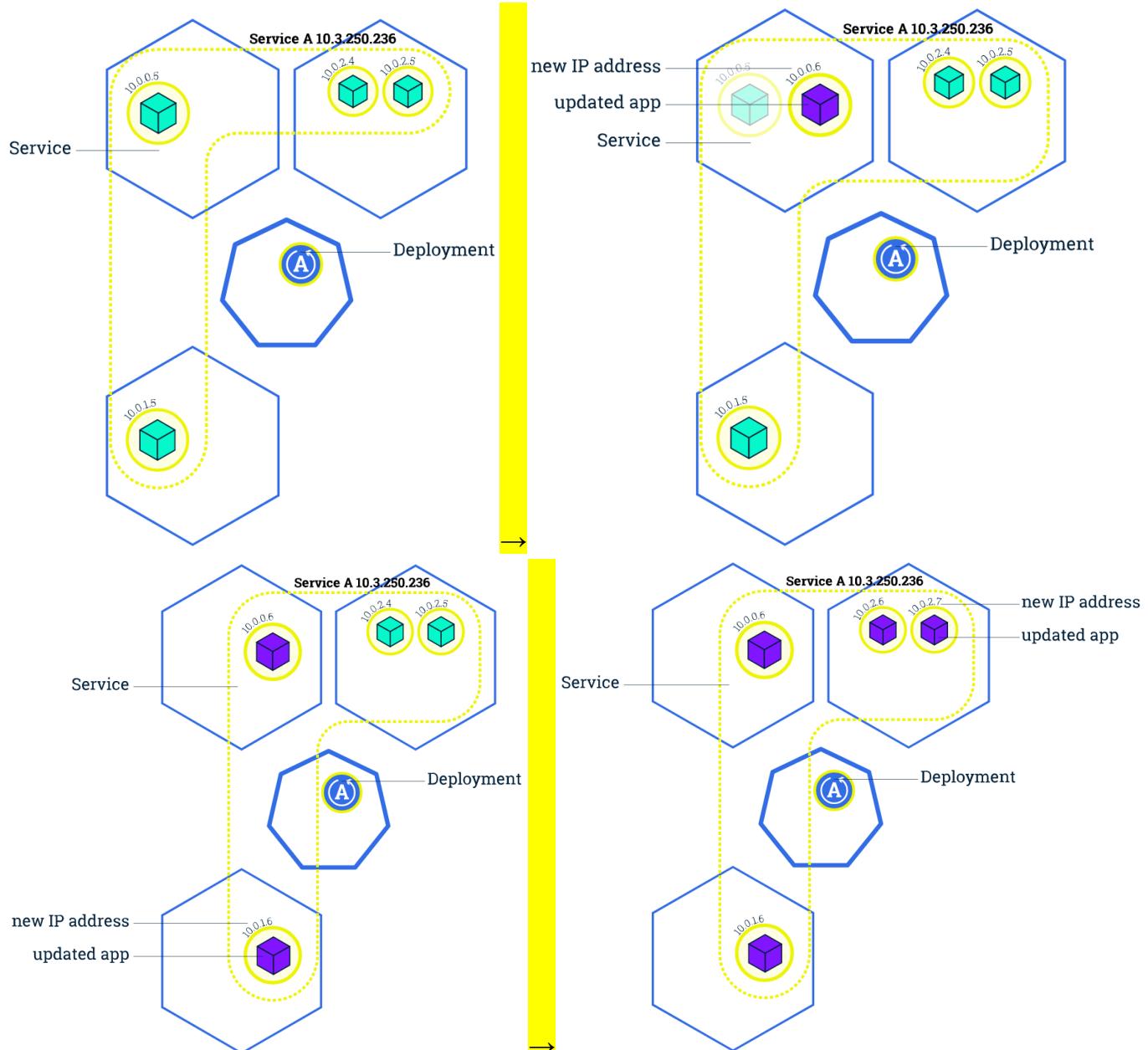
```
$ kubectl get pods -o wide
```

Questo conferma che i 2 Pod sono stati terminati.

## Eseguire l'aggiornamento continuo, [1]

Gli utenti si aspettano che le applicazioni siano sempre disponibili e, allo stesso tempo, gli sviluppatori sono tenuti a distribuire nuove versioni più volte al giorno. In Kubernetes questo viene definito rolling updates (aggiornamenti continui). Gli aggiornamenti continui consentono di aggiornare le distribuzioni senza tempi di inattività, aggiornando in modo incrementale le istanze Pod con quelle nuove. I nuovi Pod saranno programmati sui nodi con risorse disponibili.

Nel modulo precedente abbiamo scalato la nostra applicazione per eseguire più istanze. Questo è un requisito per eseguire gli aggiornamenti senza influire sulla disponibilità dell'applicazione. Per impostazione predefinita, il numero massimo di Pod che possono essere non disponibili durante l'aggiornamento e il numero massimo di nuovi Pod che possono essere creati è uno. Entrambe le opzioni possono essere configurate con numeri o percentuali (di Pod). In Kubernetes, gli aggiornamenti sono basati sulle versioni e qualsiasi aggiornamento del Deployment può essere ripristinato a una versione precedente (stabile).



Analogamente allo Scaling dell'applicazione, se un Deployment è esposto pubblicamente, il Servizio bilancerà il traffico solo verso i Pod disponibili durante l'aggiornamento. Un Pod disponibile è un'istanza disponibile per gli utenti dell'applicazione.

Gli aggiornamenti continui consentono le seguenti azioni:

- Promuovere un'applicazione da un ambiente a un altro (tramite aggiornamenti dell'immagine del contenitore);
- Rollback a versioni precedenti;
- Continuous Integration e Continuous Delivery di applicazioni con zero tempi di inattività.

Nella seguente esercitazione, aggiorneremo la nostra applicazione a una nuova versione ed eseguiremo anche un rollback.

## Step 1 - aggiornare la versione dell'applicazione

Per elencare i deployments, eseguire il comando:

```
$ kubectl get deployments
```

Per elencare i Pod in esecuzione, eseguire il comando:

```
$ kubectl get pods
```

Per visualizzare la versione attuale dell'immagine dell'applicazione, eseguire il comando describe pods e cercare il campo Image:

```
$ kubectl describe pods
```

Per aggiornare l'immagine dell'applicazione alla versione 2, utilizzare il comando set image, seguito dal nome del deployment e dalla nuova versione dell'immagine:

```
$ kubectl set image deployments/kubernetes-bootcamp  
kubernetes-bootcamp=jocatalin/kubernetes-bootcamp:v2
```

Il comando ha notificato al deploy di utilizzare un'immagine diversa per l'applicazione e ha avviato un aggiornamento continuo. Controllare lo stato dei nuovi pod e vedere quello vecchio che termina con il comando:

```
$ kubectl get pods
```

```
$ kubectl get pods  
NAME                               READY   STATUS    RESTARTS   AGE  
kubernetes-bootcamp-7d44784b7c-bqjtl 1/1     Running   0          77s  
kubernetes-bootcamp-7d44784b7c-m8bqc 1/1     Running   0          80s  
kubernetes-bootcamp-7d44784b7c-wprxt 1/1     Running   0          80s  
kubernetes-bootcamp-7d44784b7c-xjf1z 1/1     Running   0          77s  
kubernetes-bootcamp-fb5c67579-hjwds  0/1     Terminating   0          4m40s
```

## Step 2 - Verifica di un aggiornamento

Innanzitutto, verificare che l'applicazione sia in esecuzione. Per trovare l'IP e la porta esposti, eseguire il comando:

```
$ kubectl describe services/kubernetes-bootcamp
```

Creare una variabile d'ambiente chiamata NODE\_PORT con il valore della porta del nodo assegnato:

```
$ export NODE_PORT=$(kubectl get services/kubernetes-bootcamp -o  
go-template='{{(index .spec.ports 0).nodePort}}')  
echo NODE_PORT=$NODE_PORT
```

Quindi, eseguire un curl verso l'IP e la porta esposti:

```
$ curl $(minikube ip):$NODE_PORT
```

Ogni volta che si esegue il comando curl, si troverà un Pod diverso. Si noti che tutti i Pod stanno eseguendo l'ultima versione (v2).

Si può anche confermare l'aggiornamento eseguendo il comando rollout status:

```
$ kubectl rollout status deployments/kubernetes-bootcamp
```

Per visualizzare la versione attuale dell'immagine dell'applicazione, eseguire il comando:

```
$ kubectl describe pods
```

Nel campo Image dell'output, verificare che si stia eseguendo l'ultima versione dell'immagine (v2).

### Step 3 - rollback di un aggiornamento

Eseguiamo un altro aggiornamento e distribuiamo un'immagine contrassegnata con v10:

```
$ kubectl set image deployments/kubernetes-bootcamp  
kubernetes-bootcamp=gcr.io/google-samples/kubernetes-bootcamp:v10
```

Usare get deployments per vedere lo stato della deployment:

```
$ kubectl get deployments
```

Si noti che l'output non elenca il numero desiderato di Pod disponibili. Eseguire il comando get pods per elencare tutti i Pod:

```
$ kubectl get pods
```

\$ kubectl get pods					
NAME	READY	STATUS	RESTARTS	AGE	
kubernetes-bootcamp-59b7598c77-pktkl	0/1	ErrImagePull	0	115s	
kubernetes-bootcamp-59b7598c77-rmdd4	0/1	ImagePullBackOff	0	115s	
kubernetes-bootcamp-7d44784b7c-cm7mh	1/1	Running	0	2m27s	
kubernetes-bootcamp-7d44784b7c-k5t2t	1/1	Running	0	2m31s	
kubernetes-bootcamp-7d44784b7c-lqkv6	1/1	Running	0	2m31s	

Si noti che alcuni pod hanno uno stato di ImagePullBackOff.

Per approfondire il problema, eseguire il comando describe pods:

```
$ kubectl describe pods
```

Nella sezione Events dell'output per i Pod interessati, si può notare che la versione dell'immagine v10 non esisteva nel repository.

Per ripristinare il deployment all'ultima versione funzionante, usare il comando rollout undo:

```
$ kubectl rollout undo deployments/kubernetes-bootcamp
```

Il comando di rollout undo riporta il deployment allo stato precedente conosciuto (v2 dell'immagine). Gli aggiornamenti sono basati sulle versioni e si può tornare a qualsiasi stato precedentemente noto di un deployment.

Usare i comandi get pods per elencare nuovamente i pods:

```
$ kubectl get pods
```

Sono in esecuzione quattro pod:

\$ kubectl get pods					
NAME	READY	STATUS	RESTARTS	AGE	
kubernetes-bootcamp-7d44784b7c-bwfkm	1/1	Running	0	81s	
kubernetes-bootcamp-7d44784b7c-cm7mh	1/1	Running	0	7m7s	
kubernetes-bootcamp-7d44784b7c-k5t2t	1/1	Running	0	7m11s	
kubernetes-bootcamp-7d44784b7c-lqkv6	1/1	Running	0	7m11s	

Per verificare l'immagine distribuita su questi Pod, usare il comando describe pods:

```
$ kubectl describe pods
```

Il deployment utilizza nuovamente una versione stabile dell'applicazione (v2). Il rollback è riuscito.

Redatti da Pierpaolo Gumina