Un approccio multilivello alla sicurezza di Kubernetes e dei container

Proteggere i container dalla creazione al deployment, fino all'esecuzione

Sommario

Introduzione

Offrendo la possibilità di inserire un'applicazione e tutte le sue dipendenze in una singola immagine in grado di evolversi dallo sviluppo al test, fino alla produzione, i container hanno riscosso immediatamente un notevole successo. I container garantiscono coerenza in ambienti e deployment diversi, tra cui server fisici, macchine virtuali e cloud pubblici o privati, semplificando lo sviluppo e la gestione di applicazioni capaci di aumentare l'agilità dell'azienda.

- **Applicazioni:** grazie ai container, per gli sviluppatori è più facile creare e promuovere un'applicazione e le relative dipendenze come una singola unità. Il deployment dei container richiede solo pochi secondi. In un ambiente containerizzato, il codice dell'applicazione viene integrato con le librerie di runtime necessarie durante la generazione del software.
- **Infrastruttura:** i container rappresentano processi applicativi all'interno di sandbox in un kernel condiviso del sistema operativo Linux®. Sono più compatti, più leggeri e meno complessi delle macchine virtuali, inoltre sono portabili da un ambiente all'altro, dalle piattaforme on premise a quelle di cloud pubblico.

Kubernetes è la piattaforma di orchestrazione dei container preferita dalle grandi imprese e, poiché oggi molte aziende utilizzano i container per l'esecuzione dei servizi essenziali, la loro sicurezza diventa sempre più importante. In questo documento vengono illustrati gli aspetti chiave della protezione delle applicazioni containerizzate.

Protezione completa di Kubernetes e dei container: livelli e ciclo di vita

Per molti aspetti, la protezione dei container è simile alla protezione di qualsiasi altro processo Linux in esecuzione. Prima di distribuire ed eseguire un container, è necessario valutarne la sicurezza a tutti i livelli dello stack di soluzioni, oltre che per l'intero ciclo di vita del container e dell'applicazione. Soprattutto, la sicurezza deve essere anche vista come un processo continuo, integrato in tutto il ciclo di vita dell'ambiente IT, capace di estendersi per adattarsi alle nuove soluzioni e rispondere alle nuove minacce a mano a mano che si manifestano. La Figura 1 illustra un approccio esaustivo alla sicurezza dei container.

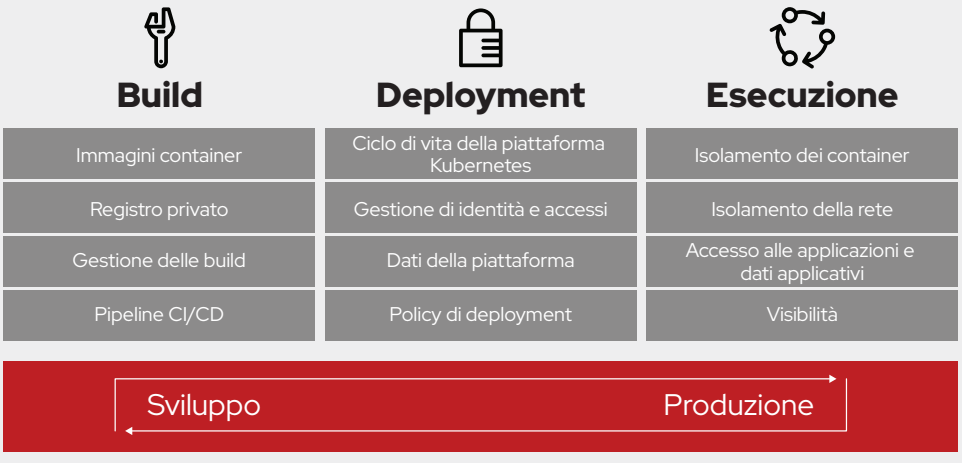

Figura 1. Per garantire la sicurezza completa dei container occorre proteggere la build, il deployment e il runtime, oltre a estendere la sicurezza con strumenti specializzati ove appropriato.

Grazie ai container, per gli sviluppatori è più facile creare e promuovere un'applicazione e le relative dipendenze come una singola unità. Inoltre, è possibile ottimizzare l'utilizzo dei server attraverso il deployment di applicazioni multi-tenant in un host condiviso. Puoi distribuire senza problemi più applicazioni in un singolo host, così come attivare e arrestare singoli container in base alle esigenze del momento. A differenza della virtualizzazione tradizionale, non è necessario utilizzare un hypervisor per gestire i sistemi operativi guest sulle diverse macchine virtuali perché, anziché virtualizzare l'hardware, i container virtualizzano i processi applicativi.

In genere le applicazioni impiegano più di un container. Anche quelle più semplici sono formate da un front end, da un back end e da un database. Per eseguire il deployment delle moderne applicazioni basate su microservizi occorre distribuire più container, a volte sullo stesso host e a volte su host o nodi diversi, come mostrato nella Figura 2.

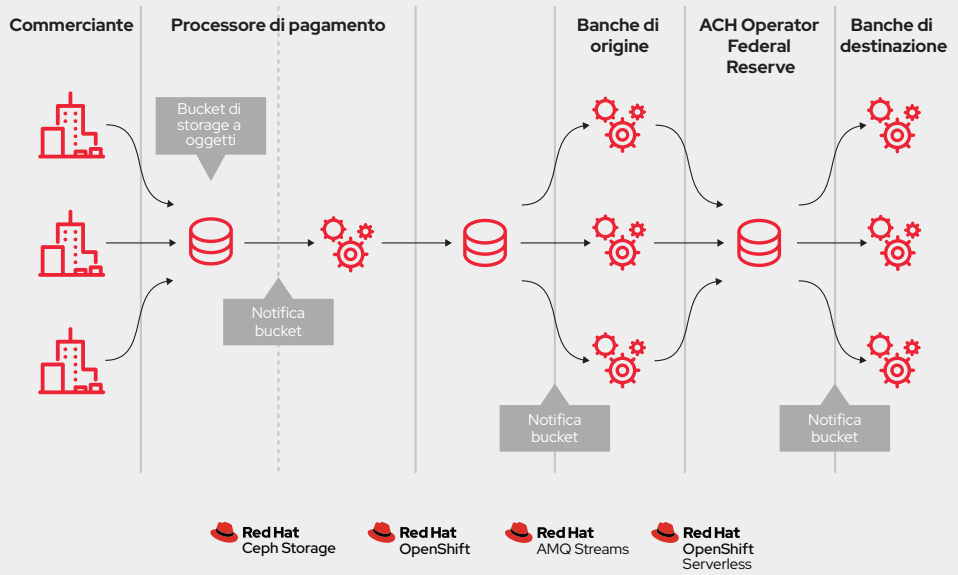

Figura 2. Gli ambienti container possono essere complessi e distribuiti fra più nodi.

Per gestire il deployment dei container su vasta scala, è necessario considerare diversi aspetti:

- I container da distribuire nei diversi host.
- L'host con la massima capacità.
- I container che hanno l'esigenza di accedere ad altri container e come possono individuarsi a vicenda.
- I metodi da utilizzare per controllare gli accessi e gestire le risorse condivise, ad esempio quelle di rete e storage.
- Come monitorare l'integrità del container.
- Come aumentare automaticamente la capacità delle applicazioni per rispondere alla domanda.
- Come abilitare il self service per gli sviluppatori e al contempo soddisfare i requisiti di sicurezza.

Puoi scegliere se creare un ambiente di gestione dei container personalizzato, dedicando tutto il tempo necessario per l'integrazione e la gestione dei singoli componenti, o adottare una piattaforma container dotata di funzionalità di gestione e sicurezza integrate. In questo modo il tuo team può concentrarsi completamente sulla realizzazione di applicazioni che forniscono valore aziendale, anziché essere costretto a reinventare l'infrastruttura.

La piattaforma container Red Hat® OpenShift® offre una piattaforma di cloud ibrido enterprise per Kubernetes, che consente di realizzare applicazioni containerizzate scalabili. Per utilizzare Kubernetes nell'intera azienda sono necessarie ulteriori funzionalità di protezione, che consentano di integrare la sicurezza direttamente nelle applicazioni, policy automatizzate per la gestione della sicurezza dei container e funzionalità per proteggerne il runtime.

Integrare la sicurezza nelle applicazioni realizzate

Nel deployment cloud native è fondamentale integrare la sicurezza direttamente nelle applicazioni. Per proteggere le applicazioni containerizzate devi:

1. Utilizzare container con contenuti affidabili.
2. Utilizzare un registro dei container enterprise.
3. Controllare e automatizzare la generazione dei container.
4. Integrare la sicurezza nella pipeline applicativa.

1. Utilizzare container con contenuti affidabili

Il contenuto dei container è importante per la gestione della sicurezza. Da qualche tempo si utilizzano applicazioni e infrastrutture formate da componenti pronti all'uso, molti dei quali sono costituiti da pacchetti open source, come il sistema operativo Linux, Apache Web Server, Red Hat JBoss® Enterprise Application Platform, PostgreSQL e Node.js. Ora questi pacchetti sono disponibili anche in versione containerizzata, che evita alle aziende di realizzare autonomamente tali componenti. Tuttavia, come sempre avviene con il codice scaricato da origini esterne, è importante conoscerne la provenienza, chi lo ha realizzato e se contiene elementi nocivi. Devi pertanto chiederti:

- Il contenuto del container rischia di compromettere la mia infrastruttura?
- Il livello applicativo contiene vulnerabilità note?
- I livelli di runtime e sistema operativo del container sono aggiornati?
- Con quale frequenza viene aggiornato il container e come posso sapere quando avviene l'aggiornamento?

Red Hat crea e distribuisce da anni affidabili pacchetti di contenuti Linux in Red Hat Enterprise Linux in tutto il suo portafoglio, e oggi quegli stessi pacchetti di contenuti affidabili sono disponibili sotto forma di container Linux. Grazie all'introduzione di Red Hat Universal Base Images, puoi contare sui livelli superiori di affidabilità, sicurezza e prestazioni delle immagini dei container Red Hat ovunque vengano eseguiti i tuoi container Linux conformi agli standard OCI (Open Container Initiative). Puoi pertanto creare un'applicazione containerizzata in Red Hat Universal Base Images, eseguirne il push nel registro dei container che preferisci e condividerla.

Red Hat fornisce inoltre una vasta gamma di immagini e operatori certificati per runtime, middleware, database e altri componenti in vari linguaggi, tramite il catalogo dell'ecosistema Red Hat. Gli operatori e i container certificati Red Hat, che possono essere eseguiti ovunque venga eseguito Red Hat Enterprise Linux, dai sistemi bare metal alle macchine virtuali, fino al cloud, sono supportati da Red Hat e dai suoi partner.

L'integrità delle immagini fornite viene costantemente monitorata da Red Hat. È disponibile un indice di integrità dei container che indica il "grado" di ciascuna immagine container, spiegando in dettaglio come gestirla, utilizzarla e valutarla, in modo da soddisfare le esigenze dei sistemi di produzione. Il grado di un container si basa in parte sull'età e sugli effetti delle correzioni di sicurezza non applicate a tutti i suoi componenti, per fornire una valutazione aggregata della sua sicurezza con termini comprensibili indipendentemente dal livello di esperienza dell'utente.

Ogni volta che Red Hat rilascia un aggiornamento della sicurezza, come le correzioni per runc CVE-2019-5736, MDS CVE-2019-11091 o VHOST-NET CVE-2019-14835, ricompila anche le immagini dei container e ne esegue il push al registro pubblico. Red Hat pubblica avvisi di sicurezza per segnalare agli utenti tutti i nuovi problemi riscontrati nelle immagini dei container certificate e indirizzarli ai relativi aggiornamenti, per consentirne l'implementazione in tutte le applicazioni che utilizzano tali immagini.

In alcuni casi, potresti avere bisogno di contenuti che non vengono forniti da Red Hat. Quando utilizzi immagini dei container provenienti da altre fonti, ti suggeriamo di usare strumenti di scansione dei container che utilizzano database di vulnerabilità continuamente aggiornati, per disporre sempre delle informazioni più recenti sulle vulnerabilità note. Poiché l'elenco delle vulnerabilità note è in continua evoluzione, ogni volta che scarichi nuove immagini dei container devi verificarne il contenuto, e continuare a monitorare lo stato delle vulnerabilità nel tempo per tutte le immagini che hai approvato e distribuito, proprio come fa Red Hat con le immagini dei suoi container.

2. Utilizzare un registro dei container enterprise per garantire un accesso più sicuro alle immagini dei container

Naturalmente, il tuoi team realizzano container che sovrappongono i propri contenuti alle immagini dei container pubbliche scaricate. L'accesso e la promozione delle immagini dei container scaricate e di quelle realizzate internamente devono essere gestiti esattamente come quelli di qualsiasi altro tipo di file binario. Esistono moltissimi registri privati che supportano la memorizzazione delle immagini dei container. Ti suggeriamo di utilizzare un registro privato che consenta l'automazione delle policy per l'uso delle immagini dei container memorizzate al suo interno.

Red Hat OpenShift include un registro privato che fornisce le funzionalità di base per la gestione delle immagini dei container. Il registro di Red Hat OpenShift supporta il controllo degli accessi basato sui ruoli (RBAC, Role-Based Access Control), che consente di gestire gli utenti autorizzati a eseguire il push e il pull di determinate immagini dei container. Red Hat OpenShift supporta anche l'integrazione con gli altri registri privati eventualmente in uso, come JFrog Artifactory e Sonatype Nexus.

Red Hat Quay, disponibile come registro enterprise standalone, offre numerose funzionalità enterprise aggiuntive, come la replica geografica e i trigger per la compilazione delle immagini.

Il progetto Clair fornisce il motore open source alla base degli scanner di sicurezza di Red Hat Quay, che consentono di rilevare vulnerabilità in tutte le immagini registrate in Red Hat Quay. Red Hat OpenShift Container Security Operator si integra con Red Hat Quay per fornire una visione a livello di intero cluster delle vulnerabilità per le immagini distribuite nella console OpenShift.

3. Controllare e automatizzare la compilazione delle immagini dei container

Per garantire la sicurezza dello stack software è fondamentale gestire il processo di compilazione delle immagini. Compilando una volta sola l'immagine da distribuire ovunque, hai la certezza di distribuire nell'ambiente di produzione il risultato esatto del processo di compilazione. È importante anche garantire l'immutabilità dei container. In pratica, anziché applicare patch ai container in esecuzione, devi compilarli e distribuirli nuovamente.

Red Hat OpenShift offre numerose funzionalità che consentono di automatizzare la compilazione in base ad eventi esterni, al fine di migliorare la sicurezza delle immagini personalizzate.

- I trigger di Red Hat Quay offrono la possibilità di distribuire un repository compilato per un file Docker in seguito a un evento esterno, come un push da GitHub, BitBucket o GitLab o un webhook.
- Il framework open source Source-to-image (S2I) consente di combinare il codice sorgente con le immagini di base (Figura 3). S2I semplifica la collaborazione fra i team operativi e di sviluppo grazie a un ambiente di compilazione riproducibile. Quando uno sviluppatore esegue il commit del codice con git, in S2I, Red Hat OpenShift può:
  - Avviare l'assemblaggio automatico della nuova immagine a partire dagli elementi disponibili, come un'immagine base S2I, e dal nuovo codice inviato (tramite webhook al repository di codice o un altro processo CI automatizzato).
  - Eseguire automaticamente il deployment della nuova immagine compilata, a scopo di test.
  - Promuovere allo stato di produzione l'immagine testata e distribuirla automaticamente tramite il processo di integrazione e deployment continui (CI/CD, Continuous Integration/ Continuous Deployment).

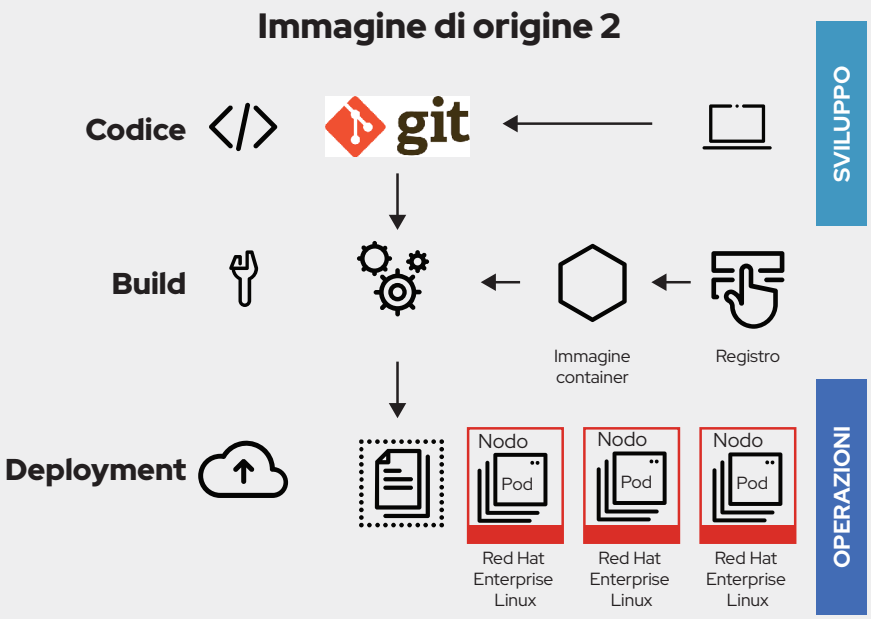

Figura 3. S2I è un toolkit e un workflow per la creazione di immagini dei container riproducibili e pronte per l'esecuzione, a partire dal codice sorgente.

- È possibile utilizzare flussi di immagini Red Hat OpenShift per monitorare le modifiche apportate alle immagini esterne distribuite nel cluster. I flussi di immagini interagiscono con tutte le risorse native disponibili in Red Hat OpenShift, quali build, deployment, processi, controller di replica o set di repliche. Monitorando un flusso di immagini è possibile a segnalare a build e deployment l'aggiunta di nuove immagini o la modifica di quelle esistenti, affinché possano reagire in automatico avviando una compilazione o un deployment, rispettivamente.

  Considera ad esempio un'applicazione realizzata con tre livelli di immagini container, ovvero base, middleware e applicativo, quindi supponi che venga rilevato un problema con l'immagine base e che quest'ultima venga ricompilata da Red Hat e reinserita nel catalogo dell'ecosistema Red Hat. Se i flussi delle immagini sono abilitati, Red Hat OpenShift è in grado di rilevare che l'immagine è cambiata e ricompilare automaticamente l'immagine dell'applicazione, incorporando l'immagine base corretta, per le build che dipendono da tale immagine e per cui sono stati definiti i trigger appositi.

  Al termine della compilazione viene eseguito il push dell'immagine personalizzata aggiornata nel registro interno di Red Hat OpenShift. Red Hat OpenShift rileva immediatamente le modifiche alle immagini contenute nel proprio registro interno e, per le applicazioni che dispongono di trigger appositi, esegue automaticamente il deployment dell'immagine aggiornata, per garantire che il codice eseguito nell'ambiente di produzione sia sempre identico all'ultima immagine aggiornata. Tutte queste funzionalità operano in sinergia per integrare la sicurezza nella pipeline e nel processo CI/CD.

4. Integrare la sicurezza nella pipeline applicativa.

Red Hat OpenShift include istanze integrate di Jenkins per CI e Tekton, una pipeline Kubernetes CI/CD di nuova generazione che supporta i container (inclusi quelli serverless). Red Hat OpenShift fornisce anche avanzate API RESTful che puoi utilizzare per integrare la tua build o i tuoi strumenti CI/CD, includendo un registro immagini privato.

Per la sicurezza delle applicazioni, è consigliabile integrare test di sicurezza automatizzati nella pipeline, includendo il registro, l'IDE (Integrated Development Environment) e gli strumenti CI/CD.

Registro: la scansione delle immagini dei container può e deve essere eseguita nel registro container privato. Puoi usare Red Hat Quay con lo scanner di sicurezza Clair per segnalare agli sviluppatori le vulnerabilità rilevate. OpenShift Container Security Operator si integra con Red Hat Quay per fornire una visione a livello di intero cluster delle vulnerabilità per le immagini distribuite nella console OpenShift. In alternativa, nel catalogo dell'ecosistema Red Hat è possibile trovare numerose soluzioni certificate di terze parti per la scansione dei container.

IDE: i plug-in Red Hat Dependency Analytics per l'IDE generano avvisi sulle vulnerabilità e consigli sulla correzione per le dipendenze dei progetti, in occasione dell'introduzione iniziale del codice nell'IDE.

CI/CD: è possibile integrare scanner con il processo CI per eseguire il rilevamento in tempo reale delle vulnerabilità note, al fine di catalogare i pacchetti open source nei container, segnalare tutte le vulnerabilità note e mantenersi aggiornati sulle nuove vulnerabilità scoperte nei pacchetti analizzati in precedenza.

Inoltre, il processo CI dovrebbe includere policy che contrassegnano le build in cui sono stati rilevati problemi durante la scansione di sicurezza, per consentire al tuo team di adottare le misure appropriate al fine di risolverli nel più breve tempo possibile.

Ti consigliamo infine di firmare i container personalizzati, per assicurarti che non vengano manomessi fra la compilazione e il deployment.

Deployment: gestione della configurazione, della sicurezza e della conformità del deployment

Per proteggere efficacemente il deployment, è necessario proteggere la piattaforma Kubernetes e automatizzare le policy di deployment. Red Hat OpenShift include le seguenti funzionalità pronte all'uso:

1. Configurazione della piattaforma e gestione del ciclo di vita.
2. Gestione di identità e accessi.
3. Protezione dei dati della piattaforma e dello storage collegato.
4. Policy di deployment.

5. Configurazione della piattaforma e gestione del ciclo di vita.

Secondo il Cloud Native Computing Foundation (CNCF) Kubernetes Security Audit, pubblicato nell'estate 2019, la principale minaccia per Kubernetes è costituita dalla complessità della configurazione e della protezione avanzata dei suoi componenti. Red Hat OpenShift risolve il problema tramite gli Operatori Kubernetes.

Un Operatore Kubernetes è un metodo che consente di creare il pacchetto di un'applicazione Kubernetes nativa, eseguirne il deployment e gestirla. L'Operatore si comporta come un controller personalizzato in grado di estendere l'interfaccia di programmazione delle applicazioni (API) di Kubernetes con la logica specifica necessaria per gestire l'applicazione. Ogni singolo componente della piattaforma Red Hat OpenShift viene inserito in un Operatore, che fornisce funzionalità di configurazione, monitoraggio e gestione automatiche per OpenShift. Gli Operatori singoli configurano direttamente componenti come il server API e la rete software defined (SDN, Software Defined Network), mentre la versione cluster dell'Operatore gestisce più Operatori in tutta la piattaforma. Gli Operatori consentono di automatizzare la gestione del cluster, inclusi gli aggiornamenti, dal kernel ai servizi nei livelli superiori dello stack.

Uno dei principali vantaggi della piattaforma container è costituito dal supporto del self service per gli sviluppatori, che permette ai team di sviluppo di distribuire in modo più semplice e veloce applicazioni basate su livelli approvati. Un portale self service offre ai team un livello di controllo sufficiente a promuovere la collaborazione, senza sacrificare la sicurezza. Operator Lifecycle Manager (OLM) fornisce agli utenti del cluster Red Hat OpenShift un framework che consente di trovare e utilizzare gli Operatori necessari per eseguire il deployment dei servizi richiesti dalle loro applicazioni. Con OLM gli utenti possono eseguire l'installazione e l'upgrade degli Operatori, oltre ad assegnare il controllo degli accessi basato sui ruoli a quelli disponibili.

Red Hat OpenShift fornisce un Operatore di conformità che automatizza la gestione della conformità della piattaforma, eseguendo i controlli tecnici richiesti dai framework di conformità. L'Operatore di conformità consente agli amministratori di Red Hat OpenShift di descrivere lo stato di conformità desiderato per un cluster, oltre a fornire una panoramica sulle carenze e delle possibili soluzioni. L'Operatore di conformità valuta la conformità di tutti i livelli della piattaforma, inclusi i nodi che eseguono il cluster. È disponibile anche un Operatore di integrità dei file, che esegue regolarmente i controlli di integrità dei file nei nodi del cluster.

6. Gestione di identità e accessi

Considerato l'alto numero di funzioni per sviluppatori e amministratori disponibile in Kubernetes, è fondamentale implementare una gestione efficace delle identità e controlli di accesso basati sui ruoli nella piattaforma container. Le API di Kubernetes costituiscono un elemento chiave per automatizzare la gestione dei container su vasta scala. Ad esempio, le API vengono utilizzate per avviare e convalidare le richieste, inclusi il deployment e la configurazione di pod e servizi.

Per proteggere la piattaforma container, è essenziale autenticare e autorizzare le API. Occorre prestare la massima attenzione alla sicurezza del server API, che costituisce un punto di accesso centrale. Il piano di controllo di Red Hat OpenShift offre autenticazione integrata tramite l'Operatore di autenticazione dei cluster. Gli account di sviluppatori, amministratori e servizi ricevono token di accesso OAuth per autenticarsi con l'API. L'amministratore può configurare il provider di identità che desidera per il cluster, in modo da consentire agli utenti di autenticarsi prima di ricevere il token. Sono supportati nove provider di identità, incluse le directory LDAP (Lightweight Directory Access Protocol).

Per impostazione predefinita, Red Hat OpenShift applica un controllo degli accessi basato sui ruoli (RBAC) a grana fine. Gli oggetti RBAC determinano se un utente è autorizzato a eseguire una particolare operazione all'interno di un cluster. Gli amministratori del cluster possono utilizzare i ruoli e i binding del cluster per controllare i livelli di accesso ai cluster OpenShift e ai progetti all'interno del cluster.

7. Protezione dei dati della piattaforma

Red Hat OpenShift protegge Kubernetes per impostazione predefinita, allo scopo di proteggere i dati in transito, e include anche opzioni per la protezione dei dati inattivi.

Per proteggere i dati della piattaforma in transito, Red Hat OpenShift esegue le operazioni seguenti:

- Utilizza il protocollo HTTPS per crittografare i dati in transito per tutti i componenti container della piattaforma che comunicano fra loro.
- Utilizza TLS (Transport Layer Security) per l'invio di tutte le comunicazioni con il piano di controllo.
- Verifica che l'accesso al server API sia basato su certificati X.509 o su token.
- Utilizza quote di progetto per limitare i potenziali danni causati da un token malevolo.
- Configura con un'Autorità di certificazione (CA, Certificate Authority) e certificati propri (in Kubernetes, lo stato master permanente viene memorizzato in etcd, mentre gli altri componenti monitorano le modifiche in etcd per adattarsi allo stato specificato).
- Esegue automaticamente la rotazione dei certificati della piattaforma.

Per proteggere i dati inattivi della piattaforma, Red Hat OpenShift esegue le operazioni seguenti:

- Se necessario, crittografa i dischi di Red Hat Enterprise Linux CoreOS e il datastore etcd per aumentare il livello di sicurezza.
- Predispone Red Hat OpenShift per gli standard FIPS (Federal Information Processing Standards). FIPS 140-2 è uno standard di sicurezza utilizzato dal governo statunitense per approvare i moduli di crittografia. Quando Red Hat Enterprise Linux CoreOS viene avviato in modalità FIPS, i componenti della piattaforma Red Hat OpenShift chiamano i moduli di crittografia di Red Hat Enterprise Linux.

I container possono essere utilizzati sia con le applicazioni stateless che con le applicazioni stateful. Red Hat OpenShift supporta sia lo storage volatile che quello permanente. La protezione dello storage collegato è fondamentale per la sicurezza dei servizi stateful. Red Hat OpenShift supporta vari tipi di storage, quali NFS (Network File System), Amazon web Services (AWS) Elastic Block Stores (EBS), Google Compute Engine (GCE) Persistent Disks, Azure Disk, iSCSI e Cinder.

Inoltre, Red Hat OpenShift Container Storage, offre storage software defined permanente integrato con una piattaforma container ottimizzata per Red Hat OpenShift. OpenShift Container Storage offre storage permanente altamente scalabile per le applicazioni cloud native che richiedono funzionalità quali crittografia, replica e disponibilità in tutto l'ambiente multicloud ibrido.

- In un host è possibile montare un **volume permanente (PV, Persistent Volume)** con qualsiasi modalità supportata dal provider di risorse. Ogni provider offre funzionalità diverse e ciascuna modalità di accesso ai volumi permanenti viene configurata per le modalità specifiche supportate da un determinato volume. Ad esempio, NFS è in grado di supportare più client di lettura/scrittura, ma uno specifico volume permanente NFS può essere esportato sul server in modalità di sola lettura. Ogni volume permanente riceve un proprio set di modalità di accesso che ne descrivono le capacità specifiche, ad esempio ReadWriteOnce, ReadOnlyMany e ReadWriteMany.
- Nel caso dello **storage condiviso** (come NFS, Ceph o Gluster), è consigliabile registrare l'ID di gruppo (GID) del volume permanente dello storage condiviso come annotazione sulla risorsa del volume permanente. Quando il volume permanente viene richiesto dal pod, il GID registrato come annotazione viene aggiunto ai gruppi supplementari del pod, per consentirgli di accedere ai contenuti dello storage condiviso.
- Nel caso dello **storage a blocchi** (come EBS, GCE Persistent Disks o iSCSI), le piattaforme container possono utilizzare le funzionalità di SELinux per proteggere dai pod senza privilegi la root del volume montato, che rimane visibile esclusivamente al container associato a cui appartiene.

Naturalmente, devi sfruttare anche le funzionalità di sicurezza offerte dalla soluzione di storage in uso.

8. Automatizzare il deployment basato su policy

Una protezione efficace richiede policy automatizzate che consentono di gestire in tutta sicurezza il deployment di container e cluster.

- Deployment dei container basato su policy

I cluster Red Hat OpenShift possono essere configurati in modo da consentire o impedire il pull delle immagini da registri immagini specifici. Per i cluster di produzione è consigliabile consentire il deployment delle immagini solo dal tuo registro privato.

Il plug-in del controller di ammissione dei vincoli del contesto di sicurezza (SCC, Security Context Constraints) di Red Hat OpenShift definisce una serie di condizioni che devono essere soddisfatte durante l'esecuzione di un pod affinché venga accettato nel sistema. I **vincoli del contesto di sicurezza** consentono di rimuovere i privilegi per impostazione predefinita, un aspetto importante che costituisce ancora la procedura consigliata. I vincoli del contesto di sicurezza di Red Hat OpenShift impediscono per impostazione predefinita l'esecuzione di qualsiasi container con privilegi nei nodi di lavoro di OpenShift. L'accesso agli ID della rete e ai processi host viene negato per impostazione predefinita.

Se lo desiderano, gli utenti dotati delle autorizzazioni appropriate possono impostare le policy SCC con criteri meno restrittivi.

Red Had Advanced Cluster Management for Kubernetes consente la **gestione avanzata del ciclo di vita delle applicazioni**, utilizzando standard open per il deployment delle applicazioni tramite policy di posizionamento integrate nei controlli di governance e nelle pipeline CI/CD esistenti.

- Gestione di più cluster basata su policy

Al fine di garantire l'alta disponibilità delle applicazioni fra più zone di disponibilità o funzionalità per la gestione comune di deployment o migrazioni in più provider di servizi cloud, come Amazon web Services (AWS), Google Cloud e Microsoft Azure, può essere utile installare più cluster. Quando si gestiscono più cluster, gli strumenti di orchestrazione devono fornire il livello di sicurezza necessario fra le diverse istanze distribuite. Come sempre, configurazione, autenticazione e autorizzazione svolgono un ruolo chiave, così come la capacità di passare dati

in tutta sicurezza alle applicazioni, indipendentemente dalla posizione di esecuzione, e di gestire le policy applicative tra i diversi cluster. Red Had Advanced Cluster Management for Kubernetes fornisce:

- **Gestione del ciclo di vita di più cluster**, che consente di creare, aggiornare e distruggere cluster di Kubernetes in modo affidabile e coerente su vasta scala.
- **Governance, gestione dei rischi e conformità basata su policy**, che utilizza le policy per configurare automaticamente e garantire la coerenza dei controlli di sicurezza conformemente agli standard aziendali di settore. Puoi anche specificare una policy di conformità da applicare a uno o più cluster gestiti.

Protezione delle applicazioni in esecuzione

Oltre a proteggere l'infrastruttura, è fondamentale anche garantire la sicurezza delle applicazioni. Per proteggere le applicazioni containerizzate è necessario:

1. Isolare i container.
2. Isolare applicazioni e rete.
3. Proteggere l'accesso alle applicazioni.
4. Garantire visibilità.

9. Isolamento dei container

Per sfruttare al meglio la tecnologia di packaging e orchestrazione dei container, i team operativi hanno bisogno dell'ambiente ottimale per l'esecuzione dei container. I team operativi hanno bisogno di un sistema operativo in grado di proteggere i container ai confini, proteggendo il kernel dell'host dalla fuga di container e i container l'uno dall'altro.

I container sono processi Linux isolati, con risorse confinate, che consentono di eseguire applicazioni all'interno di sandbox nel kernel di un host condiviso. Per proteggere i container devi adottare lo stesso approccio che utilizzi per la protezione di qualsiasi processo in esecuzione su Linux.

La pubblicazione speciale NIST 800-190 raccomanda di utilizzare un sistema operativo ottimizzato per i container, in modo da aumentare i livelli di sicurezza. Essendo il sistema operativo di base per Red Hat OpenShift, Red Hat Enterprise Linux CoreOS limita la superficie di attacco riducendo al minimo l'ambiente host e ottimizzandolo per i container. Red Hat Enterprise Linux CoreOS contiene solo i pacchetti necessari per eseguire Red Hat OpenShift e il suo spazio utente è di sola lettura. La piattaforma viene testata e rilasciata insieme a Red Hat OpenShift, allineandone le versioni, e viene gestita dal cluster. L'installazione e gli aggiornamenti di Red Hat Enterprise Linux CoreOS sono automatizzati e sempre compatibili con il cluster. Inoltre, ereditando gran parte dell'ecosistema di Red Hat Enterprise Linux, la piattaforma supporta qualunque infrastruttura desideri.

Tutti i container Linux in esecuzione su una piattaforma Red Hat OpenShift sono protetti dalle avanzate funzionalità di sicurezza Red Hat Enterprise Linux integrate nei nodi Red Hat OpenShift. Per proteggere i container in esecuzione in Red Hat Enterprise Linux vengono utilizzati gli spazi dei nomi e le capacità di Linux, SELinux e una modalità di elaborazione sicura (seccomp).

- Gli spazi dei nomi Linux gettano le basi per l'isolamento dei container. All'interno dello spazio dei nomi, i processi si comportano come se disponessero di una propria istanza delle risorse globali. Gli spazi dei nomi offrono l'astrazione necessaria per dare l'impressione che i processi all'interno di un container vengano eseguiti in un sistema operativo dedicato.

- SELinux fornisce un livello di sicurezza aggiuntivo, per isolare i container fra loro e dall'host. SELinux consente agli amministratori di imporre controlli di accesso obbligatori per ogni singolo utente, applicazione processo e file. SELinux è una sorta di parete che blocca chiunque tenti di compromettere l'astrazione fornita dallo spazio dei nomi (accidentalmente o di proposito). SELinux limita le vulnerabilità di runtime dei container e, se configurato correttamente, può impedire ai processi di fuoriuscire dal container in cui si trovano.
- Cgroups (gruppi di controllo) consente di limitare, contabilizzare e isolare l'utilizzo delle risorse (come CPU, memoria, I/O del disco e rete) da parte di un insieme di processi. Puoi usare Cgroups per evitare che un altro container sullo stesso host interferisca con le risorse del tuo container. Cgroups consente anche di controllare gli pseudodispositivi, che costituiscono un vettore di attacco molto diffuso.
- Puoi usare le capacità di Linux per bloccare i privilegi in un container. Le capacità sono unità di privilegi distinte, che possono essere abilitate e disabilitate separatamente e consentono ad esempio di inviare pacchetti IP non elaborati o di eseguire il binding a porte inferiori alla 1024. Durante l'esecuzione dei container puoi rimuovere più capacità senza interferire con gran parte delle applicazioni containerizzate.
- Infine, puoi associare un profilo di elaborazione sicura (seccomp) a un container per limitare le chiamate di sistema disponibili.

10. Isolare applicazioni e rete

La sicurezza degli ambienti multi-tenant è essenziale per utilizzare Kubernetes su scala enterprise. La multi-tenancy consente ai diversi team di utilizzare lo stesso cluster impedendo a ciascuno di utilizzare gli ambienti degli altri senza autorizzazione. Red Hat OpenShift supporta la multi-tenancy attraverso una combinazione di spazi dei nomi del kernel, SELinux, RBAC, spazi dei nomi Kubernetes (progetto) e policy di rete.

- **I progetti Red Hat OpenShift** sono spazi dei nomi Kubernetes con annotazioni SELinux. I progetti isolano le applicazioni a livello di team, gruppo e reparto. Per controllare l'accesso ai singoli progetti, vengono utilizzati binding e ruoli locali.
- **I vincoli del contesto di sicurezza** consentono di rimuovere i privilegi per impostazione predefinita, un aspetto importante che costituisce ancora la procedura consigliata. I vincoli del contesto di sicurezza di Red Hat OpenShift impediscono per impostazione predefinita l'esecuzione di qualsiasi container con privilegi nei nodi di lavoro di OpenShift. L'accesso agli ID della rete e ai processi host viene negato per impostazione predefinita.

Per eseguire il deployment delle moderne applicazioni basate su microservizi nei container è spesso necessario distribuire più container fra più nodi. Tali microservizi devono individuarsi a vicenda e comunicare fra loro. Per proteggere la rete, ti serve una piattaforma container che consenta di segmentare il traffico di un singolo cluster in modo da isolare i diversi utenti, team, applicazioni e ambienti all'interno di tale cluster. Ti occorrono anche strumenti per gestire l'accesso al cluster dall'esterno e l'accesso dai servizi del cluster ai componenti esterni. Per isolare la rete sono necessari i seguenti elementi fondamentali:

- **Controllo del traffico in entrata.** Red Hat OpenShift include CoreDNS, che fornisce un servizio di risoluzione dei nomi ai pod. Red Hat OpenShift router (HAProxy) supporta il traffico in entrata e le route per consentire l'accesso esterno ai servizi in esecuzione all'interno del cluster. Entrambi i componenti supportano policy di riapplicazione della crittografia e passthrough: "reencrypt" rimuove e riapplica la crittografia al traffico HTTP che inoltra, mentre "passthrough" trasmette il traffico senza interrompere TLS.

- **Spazi dei nomi di rete.** La prima linea di difesa di una rete è costituita dal suo spazi dei nomi. Ogni insieme di container (o "pod") riceve un intervallo di indirizzi IP e porte a cui eseguire il bind, in modo da isolare le reti di pod l'una dall'altra nel nodo. Gli indirizzi IP del pod sono indipendenti dalla rete fisica a cui sono connessi i nodi Red Hat OpenShift.
- **Policy di rete:** la rete SDN di Red Hat OpenShift utilizza le policy di rete per garantire il controllo a grana fine delle comunicazioni fra i pod. È possibile controllare il traffico di rete in entrata in qualsiasi pod proveniente da qualunque altro pod, su porte specifiche e in direzioni specifiche. Se le policy di rete sono configurate in modalità multi-tenant, ciascun progetto riceve un ID di rete virtuale dedicato, che ne isola le reti da tutte le altre reti del nodo. In modalità multi-tenant, i pod interni di un progetto possono comunicare fra loro per impostazione predefinita, ma non possono scambiare pacchetti con pod appartenenti a spazi dei nomi diversi.
- **Controllo del traffico in uscita.** Red Hat OpenShift offre anche la possibilità di controllare il traffico in uscita dai servizi in esecuzione nel cluster, utilizzando metodi del router o del firewall. Ad esempio, puoi utilizzare whitelist di indirizzi IP per consentire l'accesso a un database esterno.

11. Proteggere l'accesso alle applicazioni

Per proteggere le applicazioni è necessario gestire l'autenticazione e l'autorizzazione delle API e degli utenti dell'applicazione.

- **Controllo dell'accesso utente**

Le capacità single sign on (SSO) web costituiscono un aspetto essenziale delle applicazioni moderne. Le piattaforme container possono offrire servizi containerizzati utilizzabili dagli sviluppatori per creare le proprie applicazioni. La capacità single sign on di Red Hat è un servizio di autenticazione, single sign on (SSO) web e federazione basato su OpenID Connect o SAML (Security Assertion Markup Language) 2.0, completamente supportato del progetto upstream Keycloak. La capacità single sign on di Red Hat utilizza adattatori client per Red Hat Fuse e Red Hat JBoss Enterprise Application Platform, abilita autenticazione e single sign on web per le applicazioni Node.js e può essere integrata con i servizi directory basati su LDAP, come Microsoft Active Directory e Red Hat Enterprise Linux Identity Management. La capacità single sign on di Red Hat si integra anche con provider di accesso a social network quali Facebook, Google e Twitter.

- **Controllo dell'accesso tramite API**

Le API sono fondamentali per le applicazioni formate da microservizi. Poiché tali applicazioni utilizzano più servizi API indipendenti, si verifica una proliferazione degli endpoint di servizio, che impone l'uso di ulteriori strumenti di governance. È consigliabile utilizzare uno strumento di gestione delle API. Red Hat 3scale API Management offre una vasta gamma di opzioni standard per la protezione e l'autenticazione delle API, che possono essere utilizzati separatamente o in combinazione per generare credenziali e controllare l'accesso.

Le funzionalità di controllo dell'accesso disponibili in Red Hat 3scale API Management vanno ben oltre la protezione e l'autenticazione di base. I piani per applicazioni e account consentono di limitare l'accesso a specifici endpoint, metodi e servizi, oltre che di applicare policy di accesso per gruppi di utenti. I piani per le applicazioni consentono di impostare limiti di velocità per l'uso delle API e controllare il flusso di traffico per i gruppi di sviluppatori. Puoi impostare limiti a livello di periodo per le chiamate API in entrata, in modo da proteggere l'infrastruttura e regolarizzare il flusso del traffico. Puoi anche attivare automaticamente avvisi per le applicazioni che raggiungono o superano i limiti di velocità e definire il comportamento per quelle che superano i limiti.

Protezione del traffico applicativo

La protezione del traffico applicativo con opzioni di entrata e uscita dal cluster viene illustrata nella Sezione 10 del presente documento. Per le applicazioni basate su microservizi, la sicurezza del traffico scambiato fra i servizi nel cluster è altrettanto importante. Questo livello di gestione può essere fornito utilizzando una service mesh. Una "service mesh" è costituita dalla rete dei microservizi che formano le applicazioni in un'architettura distribuita basata su microservizi e dalle interazioni fra tali microservizi.

Grazie al progetto open source Istio, Red Hat OpenShift Service Mesh aggiunge alle applicazioni distribuite un livello trasparente per la gestione delle comunicazioni fra servizi, senza richiedere alcuna modifica al codice dei servizi. Poiché Red Hat OpenShift Service Mesh si avvale di un operatore multi-tenant per gestire il ciclo di vita del piano di controllo, può essere utilizzata a livello di singolo progetto. Inoltre, OpenShift Service Mesh non richiede risorse RBAC a livello di cluster.

Red Hat OpenShift Service Mesh fornisce servizi di rilevamento, bilanciamento del carico, crittografia e autenticazione fra servizi, fondamentali per la sicurezza, oltre a recupero dai guasti, metriche e monitoraggio.

3scale Istio Adapter è un adattatore facoltativo che consente di etichettare un servizio in esecuzione all'interno di Red Hat OpenShift Service Mesh.

12. Garantire visibilità

Per proteggere un cluster Red Hat OpenShift e i suoi utenti dall'utilizzo inappropriato, è necessario essere in grado di monitorare e controllare il cluster. Red Hat OpenShift integra funzionalità di monitoraggio e audit, oltre a uno stack di logging facoltativo.

I servizi della piattaforma container OpenShift si connettono alla soluzione di monitoraggio integrata formata da Prometheus e dal suo ecosistema. È disponibile un dashboard per la gestione degli avvisi. Se lo desiderano, gli amministratori del cluster possono abilitare il monitoraggio per i progetti definiti dagli utenti. Le applicazioni distribuite in Red Hat OpenShift possono essere configurate in modo da sfruttare i componenti di monitoraggio del cluster.

L'audit degli eventi, che è una procedura consigliata di sicurezza, è solitamente necessario per rispettare le norme di legge. La funzione di audit di Red Hat OpenShift era stata inizialmente progettata in modo da utilizzare un approccio cloud native, allo scopo di garantire resilienza e centralizzazione. In Red Hat OpenShift, l'audit di host ed eventi è abilitato per impostazione predefinita su tutti i nodi. Red Hat OpenShift offre un livello di flessibilità straordinario per la configurazione della gestione e dell'accesso ai dati di audit. Puoi scegliere il profilo di policy del log di audit, per controllare la quantità di informazioni registrata nei log di audit del server API.

I dati di monitoraggio, audit e log sono protetti tramite RBAC. I dati dei progetti sono accessibili agli amministratori dei progetti e i dati del cluster sono accessibili agli amministratori cluster.

È consigliabile configurare il cluster in modo da inoltrare tutti gli eventi di audit e log a un sistema SIEM (Security Information and Event Management), a scopo di gestione dell'integrità, conservazione e analisi. Gli amministratori del cluster possono adottare il logging del cluster per aggregare tutti i log generati dal cluster Red Hat OpenShift, come i log di audit di host e API, oltre a quelli dell'infrastruttura e dei container applicativi. La funzione di logging del cluster aggrega i log generati da tutti i nodi del cluster e li memorizza in un archivio di log predefinito. Sono disponibili varie opzioni per inoltrare i log al sistema SIEM desiderato.

Estensione della sicurezza con un ecosistema affidabile

Puoi scegliere di integrare strumenti di sicurezza di altri fornitori per aumentare ulteriormente il livello di sicurezza di Kubernetes e dei container, o per soddisfare le policy esistenti. Red Hat dispone di un vasto ecosistema di partner certificati, che offrono soluzioni quali:

- Gestione degli accessi privilegiati.
- Autorità di certificazione esterne.
- Vault esterni e soluzioni di gestione delle chiavi.
- Scanner di contenuti dei container e strumenti per la gestione delle vulnerabilità.
- Strumenti di analisi del runtime dei container.
- SIEM.

Conclusioni

Il deployment di microservizi e applicazioni basati su container non è solo un problema di sicurezza. La tua piattaforma container deve offrire un'esperienza adeguata per i tuoi team operativi e di sviluppo. Ti serve una piattaforma applicativa basata su container, di livello enterprise e incentrata sulla sicurezza, che consenta a sviluppatori e operatori di lavorare senza compromettere le funzioni necessarie a ciascun team, migliorando anche l'efficienza operativa e l'utilizzo dell'infrastruttura.

Red Hat OpenShift si basa su un nucleo di container Linux standard e portabili, che integrano funzionalità di sicurezza come:

- Strumenti integrati di compilazione e CI/CD, per le procedure DevOps sicure.
- Una versione di Kubernetes dotata di sicurezza avanzata e predisposta per l'ambiente enterprise, con funzioni integrate per la configurazione, la conformità e la gestione del ciclo di vita della piattaforma.
- Affidabili funzioni RBAC integrabili con i sistemi di autenticazione enterprise.
- Opzioni per la gestione del traffico in entrata e in uscita dal cluster.
- Rete SDN e service mesh integrate, con supporto per la microsegmentazione della rete.
- Supporto per la protezione dei volumi di storage remoti
- Red Hat Enterprise Linux CoreOS, ottimizzato per l'esecuzione di container su vasta scala con livelli di isolamento elevati.
- Policy di deployment per automatizzare la sicurezza al runtime.
- Funzioni di monitoraggio, audit e logging integrate.

Red Hat OpenShift fornisce inoltre la più ampia gamma di linguaggi di programmazione, framework e servizi supportati (Figura 4). Red Hat Advanced Cluster Management for Kubernetes fornisce funzioni strettamente integrate per la gestione di più cluster.

Red Hat OpenShift è disponibile per l'esecuzione in OpenStack, VMware, sistemi bare metal, AWS, Google Cloud Platform (GCP), Azure, IBM Cloud e qualsiasi piattaforma che supporti Red Hat Enterprise Linux. Red Hat fornisce anche il servizio di cloud pubblico Red Hat OpenShift Dedicated su AWS e GCP. Il servizio Azure Red Hat OpenShift viene offerto congiuntamente da Red Hat e Microsoft. Red Hat OpenShift Service viene offerto congiuntamente da Red Hat e Amazon.

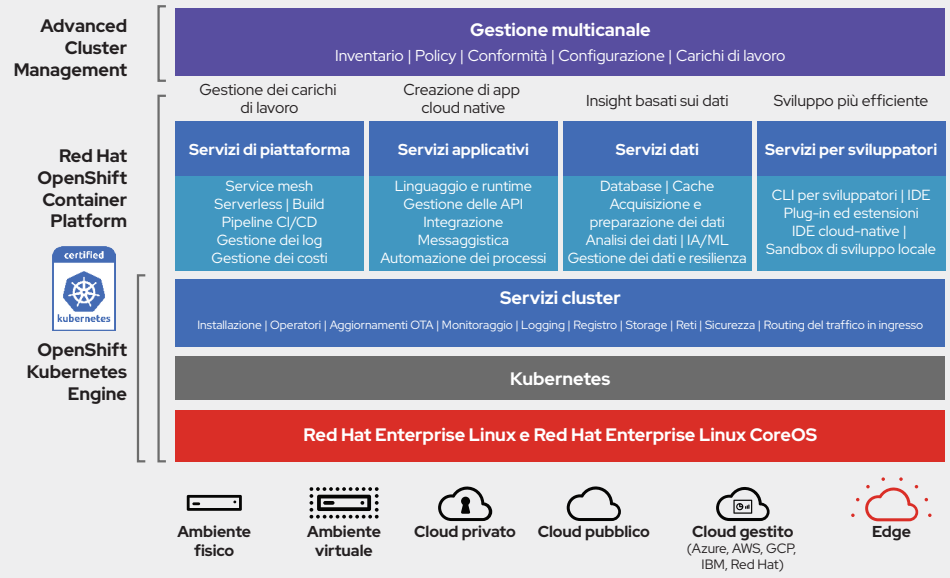

Figura 4. Red Hat OpenShift fornisce uno stack di tecnologie dotato di applicazioni complete, per ambienti fisici, virtuali e di cloud pubblico o privato.

Red Hat è un provider leader di settore che da oltre vent'anni fornisce affidabili soluzioni open source ai clienti enterprise e ora, tramite soluzioni come la piattaforma container Red Hat OpenShift, Red Hat Advanced Cluster Management for Kubernetes e il suo portafoglio di prodotti Red Hat abilitati per i container, applica gli stessi livelli di affidabilità e sicurezza anche ai container.

INFORMAZIONI SU RED HAT

Red Hat è leader mondiale nella fornitura di soluzioni software open source. Con un approccio basato sul concetto di community, distribuisce tecnologie come Kubernetes, container, Linux e hybrid cloud caratterizzate da affidabilità e prestazioni elevate. Red Hat favorisce l'integrazione di applicazioni nuove ed esistenti, lo sviluppo di applicazioni cloud-native, la standardizzazione su uno tra i principali sistemi operativi enterprise, e consente di automatizzare e gestire ambienti complessi in modo sicuro. I pluripremiati servizi di consulenza, formazione e assistenza hanno reso Red Hat un partner affidabile per le aziende della classifica Fortune 500. Lavorando al fianco di provider di servizi cloud e applicazioni, system integrator, clienti e community open source, Red Hat prepara le organizzazioni ad affrontare un futuro digitale.

facebook.com/RedHatItaly
twitter.com/RedHatItaly
linkedin.com/company/red-hat

ITALIA
it.redhat.com
italy@redhat.com

EUROPA, MEDIO ORIENTE, E AFRICA (EMEA)
00800 7334 2835
it.redhat.com
europe@redhat.com

it.redhat.com
#F26463_1220