ISTITUTO NAZIONALE DI FISICA NUCLEARE

Direzione Sistemi Informativi

INFN-23-25-DSI
22 giugno 2023

La piattaforma alla base di sviluppo e gestione dell'Infrastruttura della Direzione Servizi Informativi

Stefano Bovina[1], Guido Guizzunti[1], Giuseppe Misurelli[1]
[1)] *INFN, Direzione Sistemi Informativi, I-00044 Frascati (Roma), Italy*

Abstract

Garantire la continuità di servizio dei sistemi gestionali usati dagli utenti dell'Istituto e favorirne l'evoluzione tecnologica sono due delle principali sfide a cui l'Ufficio Sviluppo e Gestione Strategica dell'Infrastruttura risponde. In questo lavoro viene descritta la piattaforma, creata dall'Ufficio nel corso del tempo, con cui continuità di servizio ed evoluzione tecnologica vengono soddisfatte grazie all'unione fra l'Infrastruttura di Virtualizzazione utilizzata, una serie di Servizi messi a disposizione (es. Monitoraggio e Allarmistica, CICD) e delle Metodologie (es. Efficienza Operativa, Approccio DevOps) che regolano le attività e l'interazione con la piattaforma stessa. Una piattaforma tecnologica, organizzativa e culturale che nel tempo ha fatto emergere un ecosistema in grado di alimentare processi capaci da una parte di operare infrastruttura e servizi in maniera efficiente, dall'altra di promuovere flessibilità ed elasticità nell'evoluzione dei sistemi gestionali informatici usati nell'Istituto.

DOI n. 10.15161/oar.it/77193

Published by
Laboratori Nazionali di Frascati

1. Introduzione

Progettazione, sviluppo e gestione dei sistemi gestionali informatici dell'Istituto trovano le loro fondamenta nell'infrastruttura tecnologica che, unitamente ai servizi offerti e alle pratiche messe in atto, si rende necessaria da una parte per garantire la continuità di servizio dei sistemi gestionali usati dagli utenti dell'Istituto, dall'altra per favorire le varie fasi del ciclo di vita degli stessi sistemi gestionali, oltre a garantirne l'evoluzione tecnologica.
In questa nota si vuole descrivere come, all'interno della Direzione Servizi Informativi (DSI), l'Ufficio Sviluppo e Gestione Strategica dell'Infrastruttura (nel seguito gruppo SysInfo-Ops) ha pensato l'architettura generale della stessa dettagliandone i servizi e le metodologie di cui è composta. Verranno inoltre presentate le modalità con cui servizi e metodologie concorrono a fornire un valore aggiunto in fatto di operatività, sicurezza, affidabilità, resilienza ed efficienza delle prestazioni.

2. Architettura Generale

L'architettura generale dell'infrastruttura tecnologica può essere raffigurata come una piattaforma che poggia le sue basi sul sistema di virtualizzazione dei server e dello storage presente nei tre centri di calcolo CNAF, Laboratori Nazionali di Frascati (LNF) e Laboratori Nazionali di Legnaro (LNL). Piattaforma, rappresentata in figura 1, su cui vengono creati, ospitati e gestiti tutti i sistemi gestionali usati dall'Ente e da tutti i suoi utenti grazie alla commistione di una serie di servizi e di metodologie messe in atto per garantire alta disponibilità, resilienza, sicurezza, governance e abilitazione dei processi evolutivi dei sistemi gestionali.

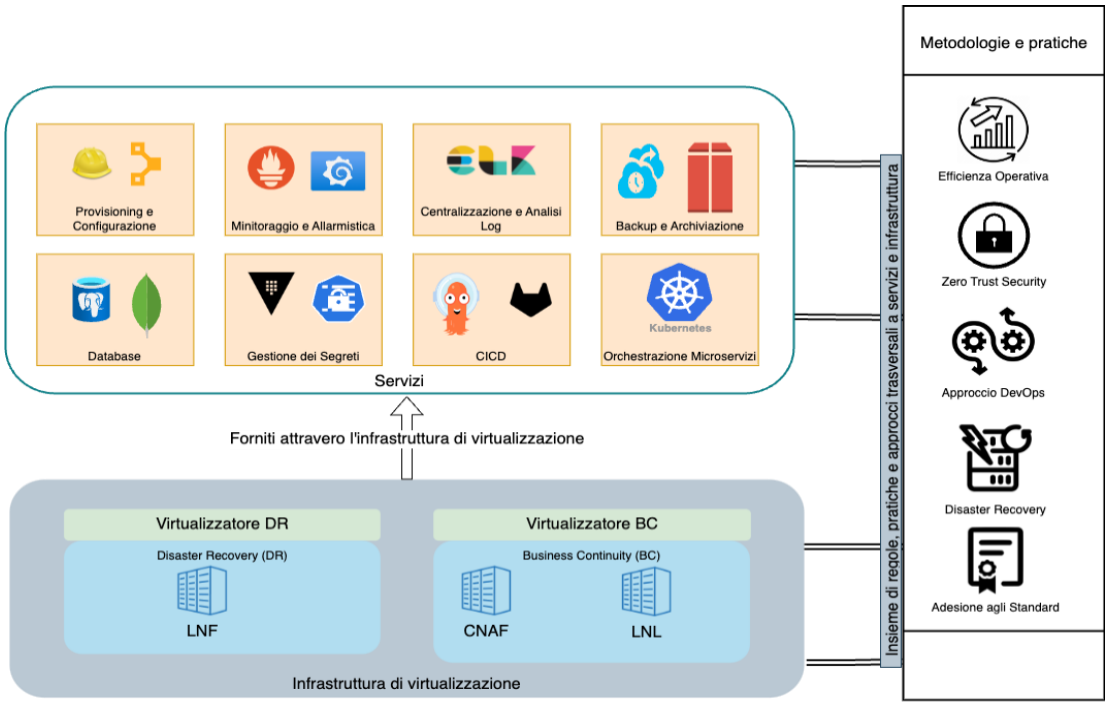

Figura 1: Architettura della DSI, pensata e realizzata come una piattaforma composta dai componenti Infrastruttura di virtualizzazione, Servizi, Metodologie e pratiche.

Una piattaforma, dunque, che può essere suddivisa nei tre moduli "Infrastruttura di Virtualizzazione", "Servizi" e "Metodologie" che saranno dettagliati nei paragrafi successivi.

2.1. Infrastruttura di Virtualizzazione

2.1.1. Business Continuity

Il servizio di Business Continuity (BC) è gestito dal gruppo dei Servizi Nazionali dell'Ente e ha come obiettivo la fornitura di una infrastruttura "hardware" di base per l'erogazione dei servizi business-critical, come quelli gestiti dal gruppo SysInfo-Ops. Nell'ambito della DSI tale infrastruttura è alla base di tutti i servizi, inclusi quelli diversi da produzione quali test e sviluppo, con l'obiettivo di garantire uniformità e continuità operativa completa anche nelle fasi di sviluppo e testing del software.

L'infrastruttura di BC, rappresentata in figura 2, è realizzata tramite uno stretched cluster VMWare distribuito geograficamente tra le sedi INFN CNAF e LNL (Laboratori Nazionali di Legnaro) ed una soluzione storage active-active (sincrona) "HyperMetro" Huawei. Tale soluzione risulta agli utilizzatori finali come un normale cluster VMWare locale, fornendo di fatto funzionalità quali migrazione a caldo, bilanciamento del carico e riavvio delle macchine virtuali in caso di problemi gravi.

2.1.2. Disaster Recovery

L'infrastruttura di Disaster Recovery (DR) è realizzata tra le sedi INFN in Business Continuity e LNF (Laboratori Nazionali di Frascati). L'obiettivo è quello di avere un terzo sito geograficamente distante che diventi operativo in caso di indisponibilità di entrambi i siti in BC. Ai LNF i sistemi, che vengono sempre installati, configurati e gestiti in maniera totalmente automatica (es: mediante Foreman e Puppet), sono live ma non raggiungibili dagli utenti, perché non puntati dal servizio di DNS. Le virtual machine in fase di deployment vengono istanziate automaticamente sia sul sito principale (BC) che su quello di DR.

A seconda del contesto e della criticità:

- il backup viene eseguito ed inviato sul sito di DR una o più volte al giorno;
- il dato viene replicato tramite funzionalità "native" (es.: replica del database).

Alcuni servizi (es.: monitoraggio e analisi dei log) sono locali al singolo sito (ed indipendenti dagli altri siti) e il dato viene aggregato a livello di dashboard.

In caso di disastro, viene attuata la procedura di disaster recovery, documentata dettagliatamente, che consiste, per esempio, nel cambio record DNS e nell'elezione dei database da slave a master.

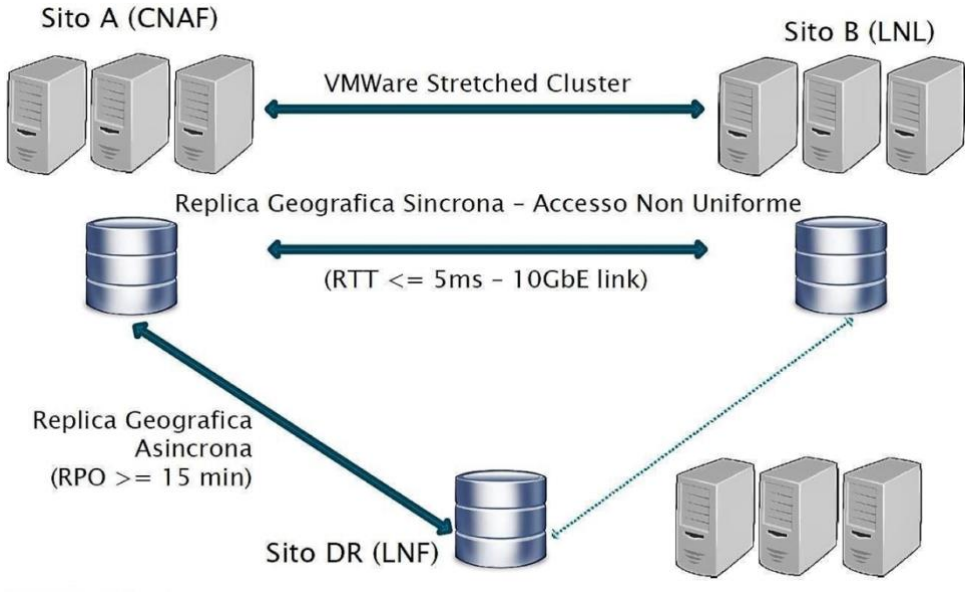

Figura 2: Schema infrastruttura di Business Continuity e Disaster Recovery distribuita rispettivamente nei siti CNAF, LNL e LNF.

2.2. Servizi

In questo paragrafo vengono descritti i servizi infrastrutturali principali che vanno a formare la piattaforma creata dal gruppo SysInfo-Ops. Per ognuno di questi servizi verranno fornite informazioni sulla loro funzionalità, sulle necessità che soddisfano e sulle modalità con cui vengono implementati.

2.2.1. Provisioning e Gestione delle Configurazioni

Il servizio di Provisioning e Gestione delle Configurazioni fa riferimento alle attività di interazione con il sistema di virtualizzazione della BC per gestire il ciclo di vita dei server virtuali – creazione, configurazione e riscontro sulle prime due azioni.

Figure 3: Ciclo di vita dei server virtuali della DSI.

La creazione e configurazione dei server virtuali, sia dei sistemi gestionali informatici dell'Istituto che di quelli di supporto ad essi, viene modellata e formalizzata in codice

tramite strumenti di Provisioning e Configuration Management (CM) [1] che velocizzano e automatizzano il deployment dei sistemi garantendo, allo stesso tempo, che le configurazioni restino quelle desiderate, riducendo complessità ed errori.

A partire dal 2015, Provisioning e Configuration Management sono realizzati utilizzando gli strumenti TheForeman [2] e Puppet [3].

Il primo viene utilizzato principalmente per modellare le tipologie di server virtuali da creare (es. partizionamento del file system, sistema operativo), raggruppare i server in domini logici (es. per ambiente di esecuzione, per tipologia di applicativo/sistema), per interagire con il sistema di virtualizzazione (es: creazione/distruzione macchine virtuali) e supportare l'intero processo tramite un'apposita applicazione web (portale web ed API REST) e CLI dedicate; tale strumento permette inoltre di tenere traccia delle risorse gestite, monitorare diverse metriche e avere un riscontro delle varie operazioni eseguite (es. output delle esecuzioni del codice di CM).

Il secondo invece viene utilizzato per la configurazione delle macchine e ad oggi rappresenta la sorgente autoritativa per descrivere ed imporre le configurazioni necessarie a garantire governance, sicurezza e funzionalità dei sistemi gestionali informatici dell'Istituto nel livello più basso di amministrazione sistemistica. Tale tecnologia permette di descrivere l'infrastruttura ed i servizi mediante codice, usando un linguaggio dichiarativo, permettendo quindi l'adozione di pratiche proprie dello sviluppo software (es: test automatici, analisi statica del codice) per la gestione della infrastruttura.

Nei fatti, il servizio di Provisioning e Configuration Management, implementato dalla coppia TheForeman/Puppet, permette di implementare i processi automatici necessari per una gestione ottimale dell'infrastruttura e dei servizi al fine di velocizzarne il setup e la manutenzione, annullando gli errori umani nella messa in opera dei sistemi gestionali informatici dell'Ente.

2.2.2. Monitoraggio e Allarmistica

Il servizio di Monitoraggio e Allarmistica si occupa di controllare costantemente lo stato di sistemi e applicativi con l'intento di prevenire eventuali situazioni di rischio e notificare i soggetti preposti a risolvere i problemi che pregiudichino il normale funzionamento dell'infrastruttura utilizzata per erogare i servizi core dell'INFN, il tutto per garantire che la disponibilità dei sistemi gestionali informatici dell'Istituto sia la più alta possibile per i suoi utenti.

Il monitoraggio avviene attraverso dei sistemi che, opportunamente configurati, raccolgono delle metriche di sistema (es. memory.used, load_avg.org) e applicative (es. mongodb.$instance.connections.current, jvm.memory.heapmemoryusage.committed).

L’allarmistica si avvale di sensori che, sulla base di soglie di attenzione e criticità ben definite, notifica i membri del gruppo DSI utilizzando differenti canali, a seconda della gravità riscontrata (es. chat, email).

I sistemi di monitoraggio usati sono due: Sensu [4] e Prometheus [5].
Essi vengono utilizzati rispettivamente nei due principali contesti in cui il servizio di Monitoraggio e Allarmistica opera, ascrivibili alle due principali tipologie di architetture a cui far ricondurre i sistemi gestionali informatici dell’Istituto: contesto monolitico e a microservizi.

Monitoraggio con Sensu
Il monitoraggio tramite Sensu si basa principalmente sull’utilizzo del pattern “publish-subscribe” (per natura asincrono) mediante l’utilizzo di un apposito middleware per lo scambio di messaggi.
Il flusso di monitoraggio inizia con la registrazione da parte del Sensu server dei vari sensori, raggruppati per sottoscrizioni/canali tematici, necessari per la raccolta di metriche e gestione degli allarmi. Tali sottoscrizioni e comandi collegati sono registrati nel middleware ed in un apposito database contenente gli stati.
Su ogni client da monitorare viene installato e configurato un apposito agent Sensu. Nella configurazione di ogni agent vengono dichiarati i canali tematici che esso sottoscrive a seconda del proprio ruolo (es: server apache, database MongoDB), eventuali personalizzazioni ed i parametri necessari per il collegamento con il middleware. Sulla base delle sottoscrizioni e delle informazioni presenti nel middleware, i singoli client procedono ad eseguire i comandi e successivamente ne comunicano il risultato al server sfruttando un canale dedicato.
Il server infine legge i messaggi provenienti dai client, ai quali può eventualmente applicare filtri, trasformazioni, processamenti, per poi salvarne lo stato in un apposito database ed in ultimo eseguire delle azioni come la generazione di un allarme e/o la scrittura persistente della misurazione in InfluxDB [6] – il time series database [7] adottato per questo tipo di dati dal gruppo SysInfo-Ops.

Monitoraggio con Prometheus
La modalità di funzionamento di Prometheus si basa sull’interazione con endpoint che forniscono dati di monitoraggio, nella maggior parte dei casi, messi a disposizione degli applicativi stessi. Questo lo ha fatto diventare molto popolare fra gli orchestratori di containers come Kubernetes, orchestratore adottato dal gruppo SysInfo-Ops e tramite il quale vengono erogati alcuni dei servizi gestionali informatici dell’Istituto come Concorsi, Consuntivi, Fondi Personale.
Il flusso di monitoraggio tramite Prometheus parte dal server stesso che, interrogando appositi endpoint ricavati sfruttando il sistema di service discovery nativo in Kubernetes (endpoint identificati grazie ad oggetti come ServiceMonitor), recupera i dati di monitoraggio (metriche e rispettivi valori nel formato Prometheus) esposti dai singoli servizi in esecuzione sull’orchestratore.

In questo modo è stato possibile estendere il servizio di Monitoraggio e Allarmistica anche al contesto Kubernetes e di farlo aderendo agli standard che regolano il monitoraggio e l'allarmistica dei microservizi.

Visualizzazione dei dati di monitoraggio
In ultimo, al fine di poter visualizzare e analizzare tali dati in grafici di vario tipo, l'applicazione web Grafana Open Source [8] viene messa a disposizione dei membri della DSI interessati attraverso apposite dashboard (esempio in figura 4).

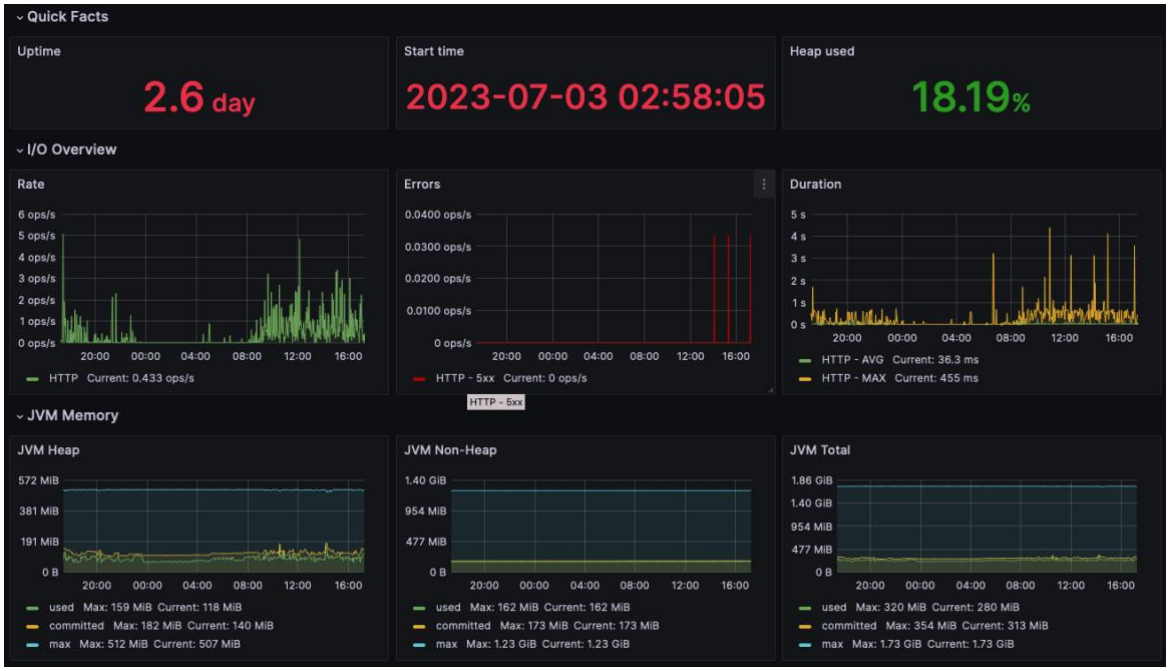

Figure 4: Dashboard con informazioni di rilevanza sullo stato della JVM di uno dei sistemi gestionali informatici dell'Istituto.

2.2.3. Centralizzazione e Analisi dei Log

Raccolta, processamento, indicizzazione e archiviazione dei log, infrastrutturali e dei sistemi gestionali informatici dell'Istituto, vengono forniti dal servizio di Centralizzazione e Analisi dei log con l'intento di conservarli in un unico punto, fornire una modalità di analisi con un'interazione molto simile a quella che si ha utilizzando i motori di ricerca web e, allo stesso tempo, dare la possibilità di creare dashboard con pannelli e grafici d'interesse.

Il servizio consiste di tre strumenti open source – Elasticsearch, Logstash e Kibana (ELK).

- Elasticsearch è un motore di ricerca full-text (metodologia di ricerca che compara ogni parola della richiesta con ogni parola contenuta nei documenti in cui la ricerca viene fatta) capace di indicizzare e conservare dati sottoforma di documenti non strutturati.

- Logstash è un software che permette la definizione ed esecuzione di pipelines di processamento di dati (in questo caso log) provenienti da sorgenti multiple, la trasformazione di tali informazioni per arricchirne il contenuto e ottimizzarne l'indicizzazione ed in ultimo l'invio a destinazioni multiple (es. Elasticsearch, servizio di archiviazione).
- Kibana è un'applicazione web che fornisce una serie di utilità per la visualizzazione dei dati, memorizzati in Elasticsearch, attraverso grafici e ricerche nei documenti indicizzati.

Attraverso tale servizio il gruppo SysInfo-Ops fornisce una soluzione robusta e completa che permette, in un contesto multi-tenant di isolamento, a sviluppatori e operatori di ricavare conoscenza dai log in tutte quelle situazioni di risoluzione ed analisi dei problemi (esempio in figura 5) e approfondimenti in merito a questioni di sicurezza (esempio in figura 6).

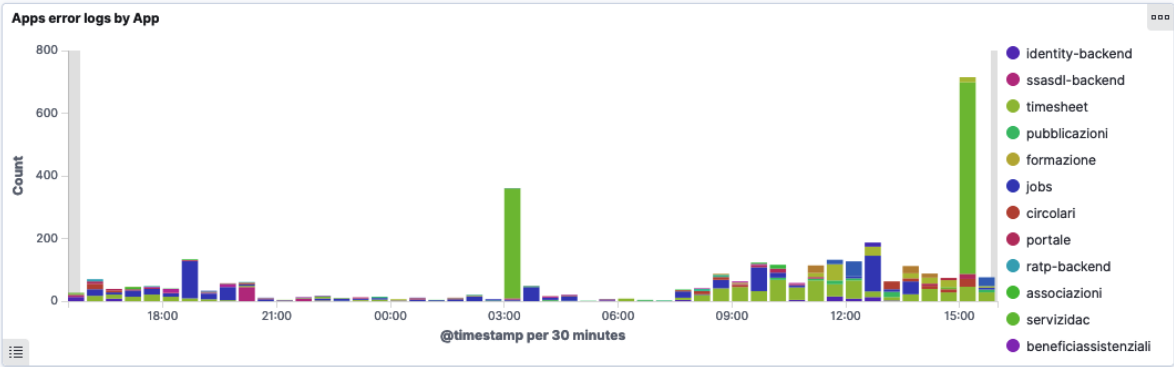

Figura 5: Errori riscontrati nei sistemi gestionali informatici dell'Istituto.

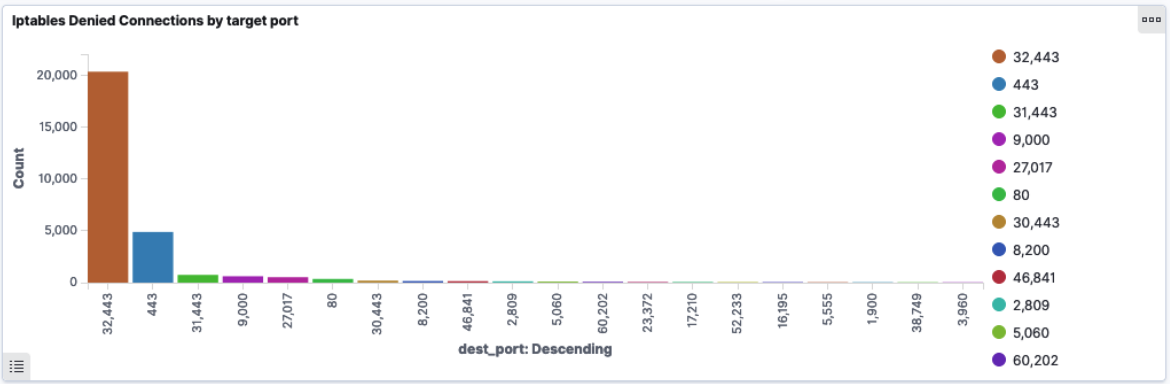

Figura 6: Connessioni rifiutate, provenienti da sorgenti non consentite, ai servizi gestiti relativamente alle porte di networking.

2.2.4. Backup e Archiviazione

La protezione dei dati e delle configurazioni dei sistemi gestionali informatici dell'Istituto viene demandata al servizio Backup e Archiviazione. Grazie a questo servizio è possibile conservare più copie degli stessi dati e delle configurazioni in più siti in modo da ripristinarli nel caso in cui gli originali vadano persi o corrotti.

I backup vengono effettuati quotidianamente con le due seguenti modalità:

- Per tutti i casi in cui è supportato l'utilizzo di sistemi di object storage, backup e archiviazione vengono demandate in toto alla soluzione adottata dal gruppo SysInfo-Ops basata su MinIO [9] e capace di offrire funzionalità quali flessibilità nell'aggiunta di storage, versionamento degli oggetti, object lock per garantire l'immutabilità degli oggetti versionati e gestione del ciclo di vita dei dati conservati (es. transizione verso tipi di storage diversi, cancellazione dei dati più vecchi di 30 giorni).
- In tutti gli altri casi, il backup avviene su disco nell'infrastruttura principale (dove risiedono per 7 giorni) e parallelamente archiviati su nastro dove vengono conservati per 30 giorni.

In entrambi i casi tutti i dati d'interesse vengono replicati sull'infrastruttura di Disaster Recovery.

2.2.5. Database

La persistenza dei dati, le relazioni fra essi e l'operatività dei vari database è prerogativa dell'omonimo servizio messo a disposizione dal gruppo SysInfo-Ops per consentire l'accesso ai dati da parte dei sistemi gestionali, controllarne l'accesso e assicurarne sicurezza e integrità.

Sono due le tecnologie di database offerte dal gruppo:

- Relazionale (o SQL) i cui dati sono strutturati in tabelle la cui interazione avviene con il linguaggio SQL. Dal punto di vista del setup i cluster di database relazionali vengono configurati per scalare verticalmente (aumento di risorse in termini di memoria, CPU, disco).
- Non Relazionale (o NoSQL) i cui dati sono descritti in documenti composti da una serie di coppie chiave/valore la cui interazione avviene via documenti JSON non strutturati. Dal punto di vista del setup i cluster di database non relazionali vengono configurati per scalare orizzontalmente (aggiunta di uno o più nodi a quelli esistenti).

È bene notare che le due tipologie di database possono non essere mutualmente esclusive per uno o più dei sistemi gestionali informatici dell'Istituto. La scelta della tecnologia di database da utilizzare per un dato servizio dipende da svariati fattori ma, in generale, se

non è strettamente necessario l'utilizzo di un database SQL la scelta ricade tipicamente su database NoSQL, in particolare su MongoDB.

I database usati, rispettivamente nelle due tipologie SQL e NoSQL sono:

- SQL: Oracle, PostgreSQL, MariaDB
- NoSQL: MongoDB, InfluxDB, Elasticsearch

2.2.6. Gestione dei Segreti

Il servizio Gestione dei Segreti viene fornito per soddisfare la necessità di avere un punto centrale in grado di conservare in maniera sicura dati e informazioni sensibili catalogate come segreti (es. password, token), di renderli fruibili in maniera programmatica (es. username e password di servizio per l'accesso ad un database da parte di un applicativo, credenziali "short-lived" generate sul momento per l'accesso ai database) e di fare auditing sull'accesso ai segreti da parte di utenti, server e applicazioni.

Tale servizio è stato implementato utilizzando lo strumento Hashicorp Vault [10], che permette al gruppo SysInfo-Ops di offrire un sistema, basato su autenticazione e autorizzazione dei soggetti che interagiscono con i dati sensibili, per la criptazione dei segreti in transito (quando richiesti) e a riposo (quando stoccati), la loro rotazione (es. validità temporanea con successiva rigenerazione) e, nel caso ce ne fosse bisogno, la revoca di privilegi/accessi.

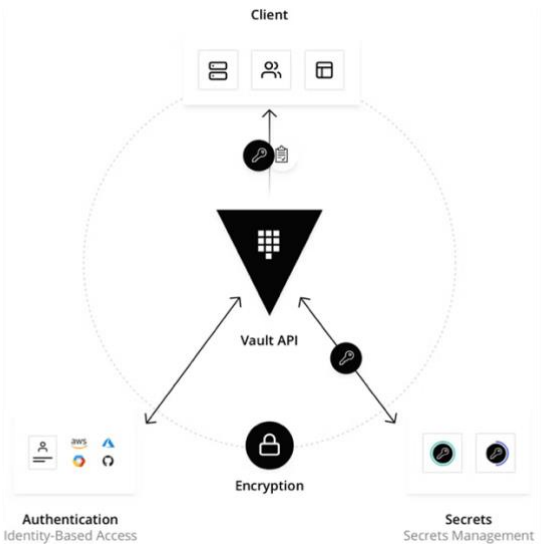

Figure 7: Il sistema di gestione dei segreti basato su Hashicorp Vault.

2.2.7. Integrazione, Deployment e Delivery Continuo (CICD)

L'abilitazione dei team di sviluppo dei sistemi gestionali informatici dell'Istituto alle pratiche di integrazione, deployment e rilascio frequente del codice applicativo avviene

attraverso il servizio di Integrazione, Deployment e Delivery Continuo (nel seguito CICD).

Grazie a tale servizio il gruppo SysInfo-Ops garantisce ai gruppi di sviluppo l'automazione e il controllo continuo in tutto il ciclo di vita delle applicazioni sviluppate (integrazione, test, distribuzione e deployment). Garanzia che, per quel che riguarda integrazione e test (CI), si concretizza nella definizione di una serie di pipeline modello, create dal gruppo SysInfo-Ops per preservare governance e sicurezza, messe a disposizione degli sviluppatori per analizzare e compilare il codice, creare un artefatto da testare e, in caso di successo dei test, caricarlo su un repository che agirà da unica sorgente da cui reperire l'artefatto validato.

Per la parte di deployment continuo (CD) invece, così come visto per il servizio di Monitoraggio e Allarmistica, l'implementazione del servizio viene differenziata nei due contesti monolitico e a microservizi.

Nel primo contesto lo strumento usato è Rundeck [11], grazie al quale vengono messi a disposizione dei cosiddetti jobs (modalità di incapsulamento di processi rappresentati da esecuzioni di comandi da eseguire su uno o più server target) che si occupano di orchestrare le attività relative alla installazione e configurazione del nuovo artefatto mediante l'ausilio di Puppet.

Nel contesto a microservizi invece il deployment continuo viene implementato dal sistema ArgoCD [12] mentre la pipeline governa le azioni annesse (es: verifiche e test sul rilascio, feedback agli sviluppatori, promozione tra ambienti). ArgoCD astrae del tutto le attività in capo ai team di sviluppo mettendosi in ascolto del repository di riferimento di un microservizio e, in seguito ad un'azione come la creazione di una nuova immagine per il rilascio, aggiorna il repository di riferimento e rilascia la nuova versione del microservizio nell'orchestratore (Kubernetes) per mezzo di automatismi che risolvono le seguenti limitazioni riscontrate in altri sistemi di CD:

- Librerie e dipendenze per il deployment da installare e gestire per eventuali agent CD.
- Token e credenziali di autenticazione e autorizzazione da inserire nel sistema di CD.
- Nessuna visibilità, da parte del sistema di CD, sullo stato del deployment nel contesto Kubernetes.

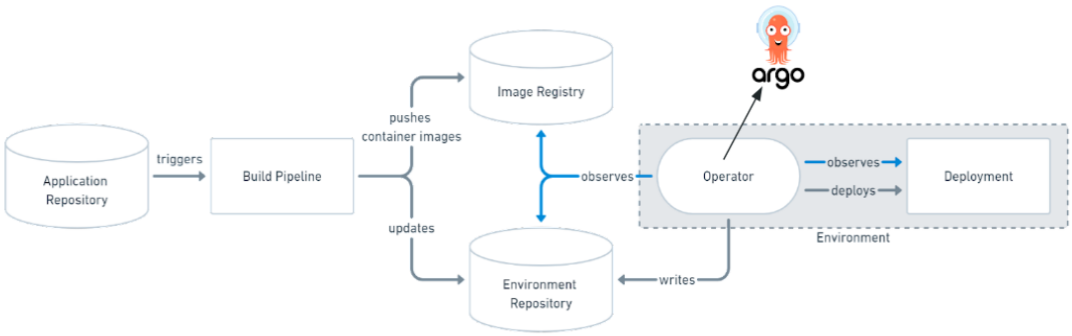

Figure 8: Schema del funzionamento di un deployment tramite ArgoCD.

2.2.8. Orchestrazione dei Microservizi

Il servizio Orchestrazione dei Microservizi si occupa di fornire cluster Kubernetes [12] per orchestrare quei sistemi gestionali informatici dell'Istituto migrati ad architetture a microservizi (es. Fondi Personali, Concorsi) oltre a tutta una serie di altri sistemi ancillari. L'intento dell'orchestrazione si concretizza principalmente nel fornire la giusta automatizzazione per i deployment, elasticità delle risorse allocate per microservizio (aumento in condizioni di carico, diminuzione quando il carico rientra), networking (interna ai microservizi, esterna da e verso uno o più di questi) e sicurezza (interna ai cluster Kubernetes, ai microservizi, alle pipeline di deployment).

Al fine di governare la complessità insita nella gestione di un tale orchestratore, i principi cardine di governance dei cluster Kubernetes della DSI identificati e implementati dal gruppo SysInfo-Ops sono i seguenti:

- Isolamento della multi-tenancy, realizzato circoscrivendo i vari deployment a namespace specifici e, allo stesso tempo, controllando il traffico di networking (internamente ed esternamente ai cluster) con network policy ben definite.
- Osservabilità demandata ai due servizi di Monitoraggio e Allarmistica (nel contesto microservizi) e di Centralizzazione e Analisi dei log.
- Deployment delle applicazioni guidati dal sistema ArgoCD.
- Sicurezza dei cluster basata sulla restrizione degli accessi al solo sistema ArgoCD (oltre agli amministratori dei cluster), sulla gestione automatica dei certificati digitali per le comunicazioni cifrate e sulla centralizzazione dei segreti applicativi (es. password, token) nello strumento Hashicorp Vault.

2.3. Metodologie

In questo paragrafo verranno presentate l'insieme di regole, pratiche e approcci che vanno a formare le metodologie utilizzate dal gruppo SysInfo-Ops e messe a disposizione degli sviluppatori.

Tali metodologie costituiscono la base per governance e sicurezza delle fasi di progettazione, sviluppo e amministrazione dei sistemi gestionali dell'Istituto.

2.3.1. Efficienza Operativa

Con efficienza operativa si intende una serie di principi guida attraverso i quali il gruppo SysInfo-Ops amministra l'infrastruttura. Tali principi sono riconducibili ai seguenti:

- Operatività programmatica (o basata su codice) grazie alla quale la totalità delle attività di creazione e gestione sull'infrastruttura e i suoi servizi sono formalizzate in codice e scripts che garantiscono procedure operative automatiche, controllate e prive di possibili errori umani.
- Risposta a eventi per prepararsi a tutte quelle occasioni, pianificate (es. deployment, fallimenti di test) e non (es. aggiornamenti di sicurezza critici), attraverso procedure collaudate per garantire risultati consistenti alla loro applicazione.
- Rivisitazione cadenzata delle procedure operative con cui, approfittando degli aggiornamenti costanti fatti ai componenti della piattaforma, si cerca di migliorarle e adattarle sulla base sia degli aggiornamenti fatti che dei limiti mostrati in determinate circostanze (es. interventi di ripristino post fallimenti, nuove funzionalità implementate dagli aggiornamenti).

2.3.2. Zero Trust Security

È l'approccio con il quale il gruppo mette in pratica la sicurezza dell'infrastruttura, dei suoi dati e degli accessi basato sul mantra "non fidarsi di niente, autenticare e autorizzare tutto": autenticazione guidata da Identity Provider autoritativi (INFN AAI/GODIVA) ed autorizzazione per mezzo di policy basate sul principio del fornire il minor privilegio possibile.

In questo modo in situazioni quali: applicazioni che devono accedere a database, utenti che accedono a server e servizi, microservizi che devono interagire fra di loro, è possibile superare il limite della protezione strettamente basato su regole di accesso/restrizione collegate a indirizzi IP delle entità da proteggere, limite che diventa molto concreto soprattutto in ambito microservizi dove la dinamicità del ciclo di vita e il cambio costante di indirizzi IP la fa da padrone.

Zero Trust Security, come metodologia, supportata nel concreto dal servizio Gestione dei Segreti, viene instillata negli automatismi di integrazione e deployment continui del software dei sistemi gestionali informatici dell'Istituto in modo da garantire il giusto livello di sicurezza in tutte le fasi di sviluppo e operatività dell'infrastruttura, dei servizi ospitati e dei dati in transito e stoccati.

2.3.3. Approccio DevOps

Attraverso l'adozione di pratiche proprie sia del DevOps che del SRE si punta a migliorare il ciclo di rilascio del software dei sistemi gestionali informatici dell'Istituto; da una parte aiutando gli addetti allo sviluppo e all'operatività ad avere un riscontro sui rispettivi aspetti del ciclo di vita dei sistemi gestionali, dall'altra favorendo automazione e osservabilità con l'intento di diminuire le tempistiche che vanno dallo sviluppo del software alla sua messa in produzione, sacrificando il meno possibile la bontà di infrastruttura e servizi dal punto di vista dell'alta affidabilità, della resilienza e della sicurezza. Una metodologia che mette insieme alcune pratiche di DevOps e SRE, che il gruppo SysInfo-Ops declina culturalmente e tecnologicamente.

Culturalmente con la condivisione e la sensibilizzazione, a favore sia degli sviluppatori che degli operatori delle questioni di alta affidabilità, resilienza e sicurezza di infrastruttura e servizi.

Tecnologicamente mettendo a disposizione una serie di utilità, di controlli e di paletti che, in maniera sempre più automatica, agevolino sviluppo e operatività in alta affidabilità di soluzioni resilienti e sicure.

2.3.4. Disaster Recovery

L'implementazione di un piano/infrastruttura di Disaster Recovery (DR) è necessaria per permettere la continuità di servizio anche a seguito di eventi catastrofici che rendano di fatto inutilizzabile la soluzione di business continuity.

L'implementazione di una soluzione di DR deve tenere in considerazione svariati aspetti, quali RPO (Recovery Point Objective) e RTO (Recovery Time Objective) derivati dalla criticità del servizio, eventuali limiti tecnologici ed il costo relativo all'implementazione: più bassi sono RPO/RTO, maggiore sarà il costo.

A prescindere dal servizio, viene sempre previsto un meccanismo iniziale di ripristino attingendo ai dati di backup, mentre in quelli business-critical che lo consentono (es. Database PostgreSQL, MongoDB) la soluzione adottata è di tipo warm standby. Nell'implementazione warm standby i servizi sono attivi nel sito di DR, riducendo di fatto al minimo la quantità di dati persi e i tempi di ripristino, mentre il traffico di produzione non è attivo. In caso di disastro viene attivata la procedura semi-automatica che include l'aggiornamento dei record DNS per abilitare il traffico verso il sito di DR.

La verifica delle procedure di ripristino viene fatta periodicamente in maniera automatica attraverso il riallineamento degli ambienti partendo dal backup (es. Oracle database), mentre il corretto funzionamento dei servizi warm è intrinseco nell'approccio stesso e monitorato real-time da appositi sistemi di monitoraggio.

Il sito di DR è autonomo per servizi di base necessari alla sua corretta operatività indipendentemente dal corretto funzionamento del sito principale; tra questi troviamo il sistema di monitoraggio, analisi log e backup.

L'allineamento dei dati tra sito principale e sito di DR viene implementato mediante apposite tecnologie di replica proprie degli strumenti utilizzati, mentre la gestione della configurazione e deployment dei servizi viene gestita da Puppet/Foreman e annessi sistemi di deployment/orchestrazione (es: Rundeck) su entrambi i siti (vedi figura 9).

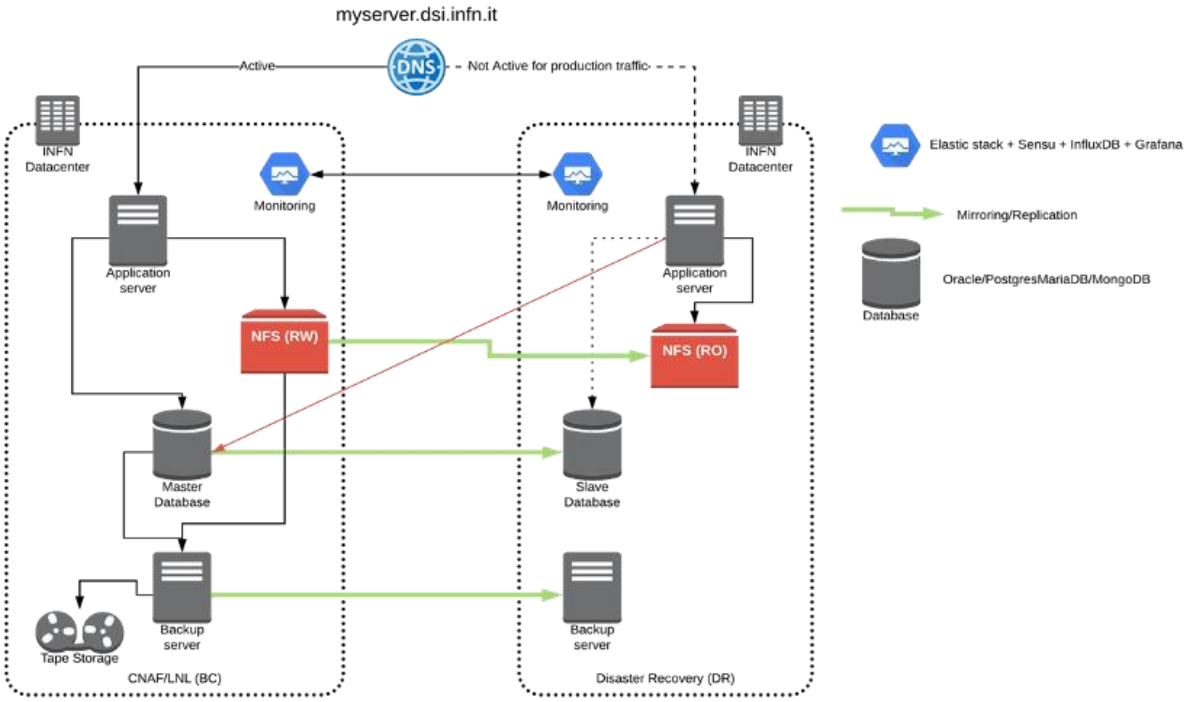

Figure 9: Schema implementativo del Disaster Recovery nella DSI

2.3.5. Adesione agli standard

Il gruppo SysInfo-Ops approccia le varie fasi di progettazione, sviluppo e amministrazione dei sistemi gestionali dell'Istituto attingendo ad una serie di standard (ufficiali, de facto e creati internamente al gruppo) che regolano i due ambiti: adozione di strumenti e software tecnologico e linee guida da osservare nelle varie attività.

Per quel che riguarda gli strumenti e software tecnologico, il principio standard che ne regola l'adozione può essere riassunto nello slogan: strumenti e software open source basati su una vasta comunità di utilizzatori in un contesto fortemente collaborativo. È questo il caso, ad esempio, di tutti quei componenti utilizzati nei cluster Kubernetes e approvati dalla Cloud Native Computing Foundation (CNCF) [14] proprio perché rilasciati con una delle licenze di proprietà intellettuale approvate dalla stessa CNCF [15].

Le linee guida da osservare riguardano invece tutta una serie di processi, passi e prassi descritti in documenti che regolano attività come: scrittura/manutenzione del software, on boarding di nuovi sviluppatori/amministratori, accesso alle risorse (es. VPN, servizi Web, Database), formulazione di richieste da fare al gruppo SysInfo-Ops (es. come

migrare un progetto prototipo di una nuova applicazione negli ambienti di sviluppo, test e produzione).

3. Conclusioni

La piattaforma fin qui descritta è il risultato del continuo lavoro, svolto dall'Ufficio Sviluppo e Gestione Strategica dell'Infrastruttura in collaborazione con tutte le terze parti interessate, con l'intento di fornire un valore aggiunto in fatto di operatività, sicurezza, affidabilità, resilienza ed efficienza delle prestazioni in tutte le fasi di progettazione, sviluppo e gestione dei sistemi gestionali informatici dell'Istituto.

Le tre caratteristiche (tecnologica, organizzativa e culturale) che contraddistinguono la piattaforma hanno innescato un modus operandi in grado di far leva sulle infrastrutture virtualizzate di Business Continuity e Disaster Recovery per garantire alta affidabilità, disponibilità e sicurezza dei servizi erogati dalla Direzione Sistemi Informativi INFN e di farlo grazie ai servizi (es. Provisioning e Configurazione, Monitoraggio e Allarmistica) e alle metodologie (es. Efficienza Operativa, Disaster Recovery) descritte.

Allo stesso modo, l'evoluzione tecnologica dei sistemi gestionali informatici dell'Istituto viene favorita dal giusto bilanciamento fra flessibilità, governance e sicurezza forniti dai servizi (es. CICD, Gestione dei Segreti, Orchestrazione dei Microservizi) creati e dalle metodologie (es. Approccio DevOps, Adesione agli Standard, Zero Trust Security) messe in atto.

Riferimenti

[1] https://www.gartner.com/en/information-technology/glossary/cm-configuration-management

[2] https://theforeman.org

[3] https://www.puppet.com

[4] https://www.usenix.org/conference/lisa16/conference-program/presentation/porter

[5] Julius Volz and Björn Rabenstein, USENIX Association, May 2015

[6] https://www.influxdata.com/products/influxdb-overview

[7] Time Series Databases, Ted Dunning, Ellen Friedman, New Ways to Store and Access data, 2015

[8] https://grafana.com/oss/grafana

[9] https://min.io/product/overview

[10] https://www.vaultproject.io

[11] https://www.rundeck.com/

[12] https://argoproj.github.io/cd/

[13] https://kubernetes.io

[14] https://www.cncf.io/

[15] https://github.com/cncf/foundation/blob/main/allowed-third-party-license-policy.md