TETRATE ISTIO WITH TSB

TOWARDS HIPAA COMPLIANCE

White Paper | *by Bart Van Bos*

tetrate

www.tetrate.io

Table of contents

Introduction

This white paper provides a brief overview of healthcare industry regulations for protecting patient data and describes how a service mesh solution can help health systems comply with those regulations.

We will start with a short overview and explanation of regulations derived from the Health Insurance Portability and Accountability Act of 1996, known as HIPAA regulations [1]. Many organizations are now required to show HIPAA compliance in the use of relevant patient data.

We will then describe how software can be developed and delivered in a more secure, agile, and reliable manner using a service mesh architecture. The leading service mesh software is the Istio open source software. The leading management software for service mesh is Tetrate Service Bridge (TSB), offered by Tetrate, started by founders and maintainers of Istio to offer complementary software and services.

Service mesh and Istio are the reference standards for an approach called zero trust architecture (ZTA) [2], which has recently been mandated for use across the federal government. Tetrate has federal government customers who use Istio and TSB to implement zero trust approaches. You may hear of ZTA being used to help enforce HIPAA standards and in other applications where security is paramount.

Bringing these streams together, we will show how TSB can be used effectively to help organizations achieve, maintain, and prove HIPAA compliance.

HIPAA regulations

With the advent of electronic processing, communication, and storage of medical data, it's much easier to share patient information amongst health data networks. In fact, the scope of health data sharing is widening as organizations seek to harness the value of patient data and use it to continuously improve patient care. . But how can people's private health information be kept confidential and secure at the same time? In the US, HIPAA is the federal law that establishes standards for data sharing and is designed to prevent disclosure of sensitive patient information without that patient's consent. .

The rules

HIPAA establishes three rules for safeguarding the privacy and security of medical information:

- The **Privacy Rule** gives patients specific rights regarding their health information. It also regulates who else can have access to this information;
- The **Security Rule** establishes standards for safeguarding this information when it is transmitted and stored in electronic form;
- The **Enforcement Rules** set up procedures for investigating potential violations of HIPAA regulations and establish penalties to help enforce compliance.

HIPAA was followed by two related acts: the Genetic Information Non-Discrimination Act [3] of 2008, known as GINA, focuses on protecting people's genetic information. The Health Information Technology for Economic and Clinical Health Act [4] of 2009, known as HITECH, extended the reach of HIPAA requirements and updated the penalties for violating them.

In 2013, a final Omnibus Rule [5] officially integrated GINA and HITECH with HIPAA, forming the health information regulations that are in force in the US today.

The data

HIPAA defines Protected Health Information [6] (PHI) as any data about a person's health, their health care, or payment for their health care, that:

- is created or collected by a healthcare provider, health plan, or health care clearinghouse, their business associates and subcontractors;
- is transmitted or maintained in electronic form or any other medium;
- identifies the person, or could be used to identify the person that the data relates to.

PHI can include things such as physician's notes and health care billing information, blood test results, doctor's telephone records, an MRI scan, and appointment scheduling notes. PHI that is stored or transmitted in electronic form is sometimes referred to as ePHI.

The personas

HIPAA groups the organizations and people who are responsible for protecting health information into three categories:

Covered entities - health care providers who electronically transmit health information, including health care professionals (such as doctors), organizations they work for, such as hospitals, and programs that pay for health care, such as health insurance agencies, Medicare [7] and Medicaid [8].

- **Business associates** - people or businesses who have access to PHI as part of working with or providing services to a covered entity, such as billers, medical transcriptionists, or auditors.
- **Subcontractors** - a person or a business who has access to PHI while working with or providing services to a business associate, such as a CPA, an attorney, or a cloud storage company.

If you or your employer falls into one of these categories, you will need to comply with HIPAA regulations.

Requirements of HIPAA security rule

The **HIPAA Security Rule** [9] deals with protecting the confidentiality and integrity of PHI when it is in electronic form, also known as ePHI. To accomplish this, the Security Rule requires the use of **administrative, physical** and **technical safeguards** on the part of entities that have custody of this information [10].

Administrative safeguards

Administrative safeguards are policies and procedures that limit access to ePHI, such as security management processes and staff training.

Physical safeguards

Physical safeguards restrict access to computers and other equipment that store and transmit ePHI and include locks for doors, alarm systems, and cable locks for computers.

Technical safeguards

Technical safeguards protect the data storage and transmission systems that handle ePHI from inside computer systems and networks. These safeguards include hardware, software, and other technology that protects access to ePHI. Examples of required technical safeguards include:

- access controls to restrict access to ePHI to authorized personnel only;
- audit controls to monitor activity on systems containing ePHI, such as an electronic health record system;
- integrity controls to prevent improper ePHI alteration or destruction;
- transmission security measures to protect ePHI when transmitted over an electronic network.

Technical safeguards are often thought of as being "from the outside in" — that is, protecting data from being accessed via a device used by individuals, such as those listed above. What is often less thought of is access "from the inside out," by directly attacking stored data and data being transmitted between devices.

Breach notifications and penalties

The impermissible access, acquisition, use, or disclosure of PHI is called a breach. When a breach is reasonably suspected, HIPAA presumes that a breach has actually occurred. Covered entities must inform patients of any breach that affects them within 60 days of the date of the breach. If the breach affects 500 people or more, the covered entity must alert the news media as well. This is why you so often hear about breaches, or potential breaches, in the news.

HIPAA also requires that the Department of Health and Human Services [11] be notified of all breaches [12]. The penalties for data breaches can be significant: up to $1.5M per violation. If the data for many individuals is exposed, for instance, the exposure of each person's information may be considered a separate breach, and each such breach may be subject to the full penalty. Anyone who creates, receives, maintains, or transmits PHI on behalf of a covered entity can be subject to these penalties, including individuals and business entities. There are strong incentives to follow HIPAA guidelines carefully.

Nonetheless, healthcare data breaches are staggeringly common. Healthcare data breaches between 2009 and 2021 resulted in the loss, theft, exposure, or impermissible disclosure of more than 314 million healthcare records, equating to about 95% of the 2021 population of the United States. In 2021, an average of 1.95 healthcare data breaches of 500 or more records were reported each day [13].

Why service mesh for HIPAA?

We've shown that HIPAA is a very important set of standards and that penalties are steep. Preventing health data breaches is critical both to the ethical handling of sensitive information and to maintaining patient and public trust.

But why is service mesh relevant to implementing HIPAA standards? Because so many data privacy violations, including those relating to HIPAA, occur "on the back end" – for stored data, for data in transit, or within systems that process data. Without a service mesh, it is very complex to consistently encrypt health data in transit or maintain a centralized governance of authorization policy.

Service mesh is a back-end system that dictates how data processing and certain kinds of data transmission are conducted. It's inherently more secure than other approaches for a number of reasons, especially because all data transmissions are authenticated, authorized, and encrypted. Messages within a service mesh are much less likely to be sent to the wrong destination, and any data that is intercepted is encrypted, dramatically reducing its potential visibility to attackers.

In the following sections we'll describe how a service mesh works and how it is particularly well suited to use in systems for which data privacy is highly valuable.

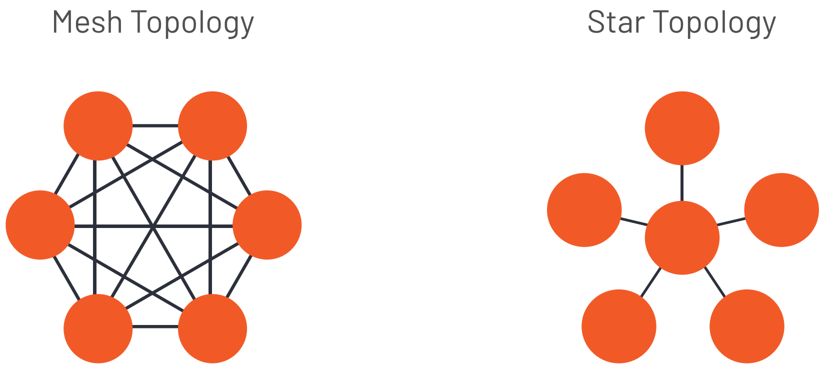
Service mesh

Meeting the requirements of HIPAA regulations is a significant IT challenge for health care organizations. The stakes only get higher as cloud migration and application modernization efforts increase the operational complexity of managing application components and data across cluster, clouds, and on-premises environments. A service mesh can help tame that complexity and offers critical security, connectivity, observability, and reliability capabilities to **achieve**, **maintain**, and **prove** HIPAA compliance, especially with regard to the technical safeguards mandated in the HIPAA Security Rule.

Topology

What is a service mesh? A service mesh is a dedicated infrastructure layer that allows you to add observability, security, and reliability features to an application. Let's break the concept down to its core, being a **mesh** of **services**.

- **Mesh:** a mesh topology allows for a direct data path between all entities within it. Unlike a star topology [Fig 1], a mesh has no central point of failure or bottleneck and is, therefore, inherently more robust and scalable. Meshes typically come with a built-in service discovery mechanism for services to find each other without the need to go through a single central point.
- **Services:** a service is a piece of software that delivers certain business functionality, typically exposed through some API of sorts or through a GUI or a web interface if user interaction is offered. It can be an application running on bare metal, or a virtual machine, or a container in a container orchestrator like Kubernetes or Openshift Container Platform (OCP), or even a serverless function from a cloud vendor.


Mesh Topology

Star Topology

Figure 1: *mesh vs star topology*

Istio is currently the most popular, most feature complete, and most adopted service mesh implementation available [Fig 2]. The Istio project was started by teams from Google and IBM in partnership with the Envoy [14] team from Lyft. It's been developed fully in open source on GitHub [15]. With a history of prompt CVE patches, paid security audits, and currently active bug bounties, Istio is the only service mesh with an ecosystem that enjoys both groundswell as well as support from multiple institutions large and small. The full list of backing organizations, which includes Tetrate as a core contributor, can be found here [16].

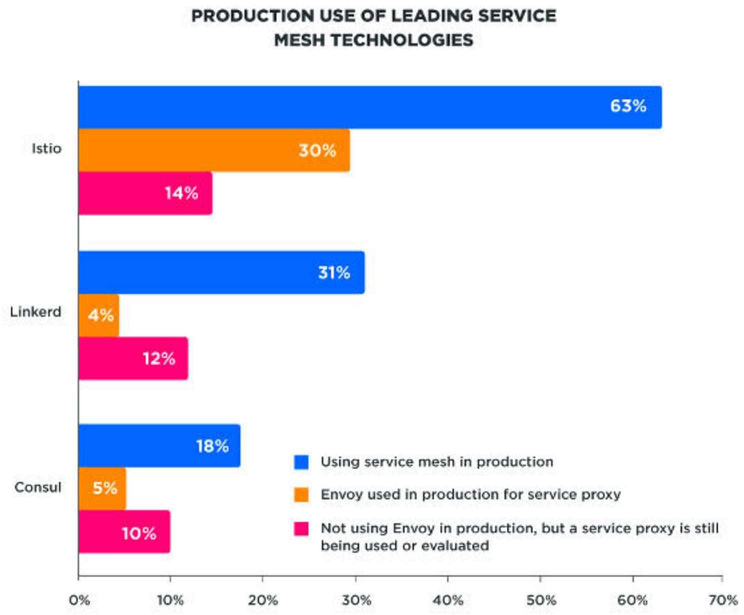
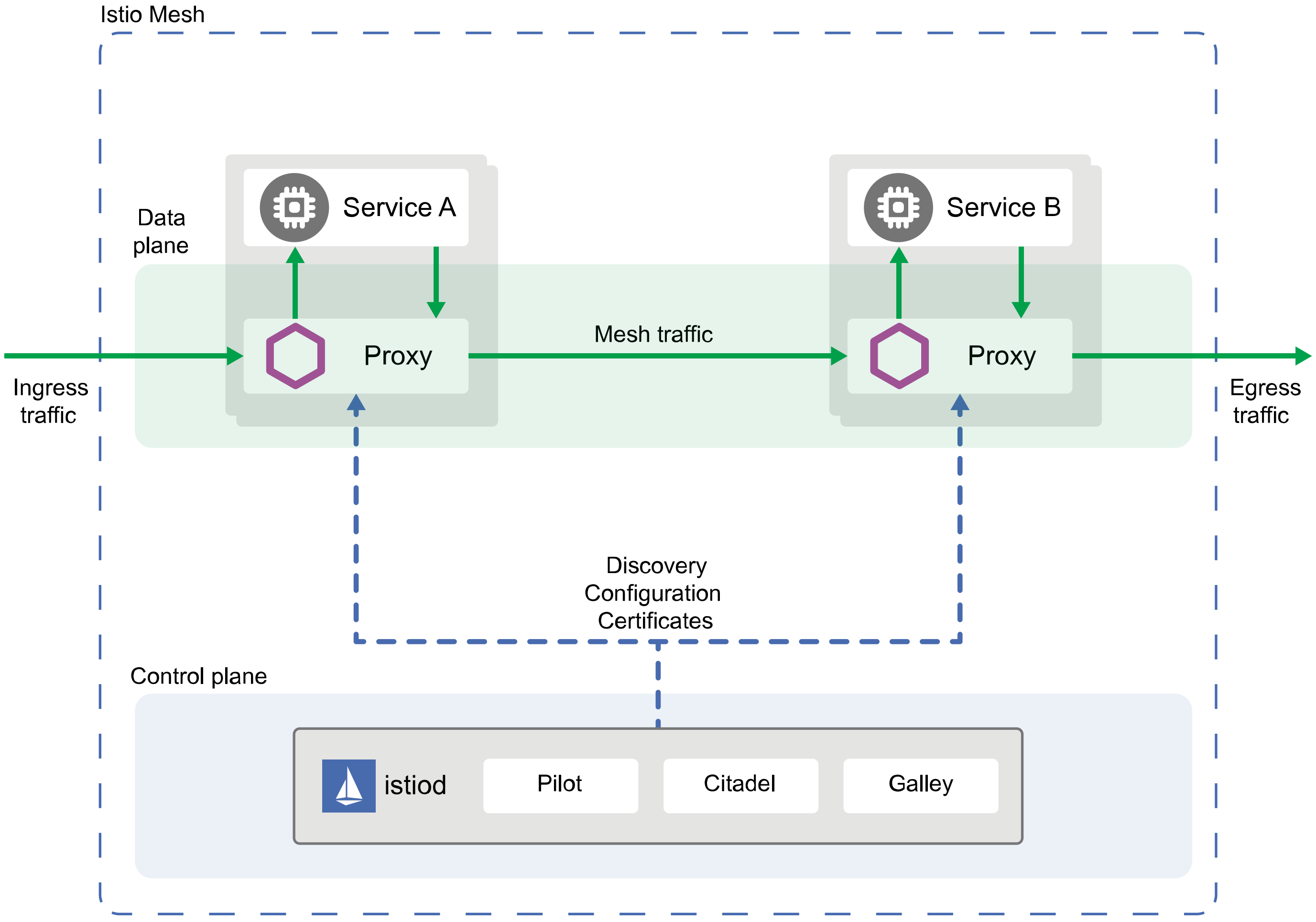


Figure 2: *Service mesh adoption*

Although a service mesh topology is most commonly associated with a microservices architecture and containerized Kubernetes environments, it can provide value beyond purely cloud-native applications as well, especially when traditional server-based, monolithic apps are brought into the mesh. The data plane of a service mesh acts on L5 (session) and L7 (application) of the OSI model [17] and can offer significant benefit to applications of any form factor. Service mesh can provide value anywhere there is existing L3 (network) and L4 (transport) connectivity.

A useful way to contextualize service mesh is to think of it as distributed middleware and a successor, in concept at least, to the older, centralized Enterprise Service Bus (ESB). In large organizations, an ESB is often used to facilitate communication between mutually interacting software components in a Service Oriented Architecture [18] (SOA). Although it has a lot of benefits, ESB presents a set of specific challenges as well. In many organizations, the star topology of an ESB has proven to be a communications bottleneck, limiting scale and increasing operational overhead. ESB also suffers from noisy neighbor problems, such that making updates to one application could destabilize other unrelated applications that use that same bus. Despite good intentions, SOA using a centralized ESB has an unavoidable single point of failure. The distributed topology of service mesh eliminates those bottlenecks, failure points, and noisy neighbor problems.

Architecture



Istio Architecture

Figure 3: *Istio Architecture*

Istio's main components can be divided into the **data plane** and the **control plane**. The data plane proxies communication between services, while the control plane is responsible for applying configuration to and collecting runtime information (like telemetry and traces) from the data plane. This separation of concerns buffers each plane from outages or performance spikes in the other.

In the data plane, communication between workloads or services are proxied by instances of the light-weight, fast, and highly configurable Envoy proxy. In containerized environments, this Envoy proxy is typically deployed following a sidecar pattern. On bare metal or VMs, the Envoy proxy is a binary installed on the host OS. All traffic that is destined for a certain service will go through this proxy, at which point important features like traffic control, security, and observability can be enabled. Likewise for all traffic leaving a service, which is routed through the same proxy.

The control plane contains several components, bundled into a single daemon called "istiod." The main responsibilities of the control plane include service discovery and advertising, configuration updates, certificate distribution, and life cycle management.

A service mesh is typically also deployed with dedicated **ingress** and **egress gateways** for controlling traffic entering or leaving the mesh.

Features

Istio's features can be divided into three major categories: network traffic control, security, and observability.

Network traffic control features [19] include:

- circuit breaking
- external DNS
- failover
- fault injection
- HTTP routing
- load balancing
- retries
- timeouts
- traffic mirroring

Security features [20] include:

- strong identity
- powerful policy enforcement
- transparent TLS encryption
- authentication
- authorization
- audit
- allow/block listing

Observability features [21] include:

- log collection
- metrics
- distributed tracing

As large organizations typically run multiple interconnected clusters, Istio also ships with rudimentary support for multi-cluster [22] deployments. Requirements driving multi-cluster deployment strategies might include locality, organizational boundaries, failure and trust domain separation, hybrid and multi-cloud environments, cost center isolation, and more. Istio provides automatic cross-cluster service discovery, but trust domain initialisation, scalability, and policy granularity fall beyond the scope of stock open source Istio. More on that in the chapter on TSB.

How Istio ensures HIPAA compliance

Let's take a closer look at how Istio's features and capabilities map to the technical safeguards required by the HIPAA Security Rule described earlier.

Access controls

Access controls restrict access to ePHI to authorized personnel only. Access control can be further split down into **identity** management, identity verification or **authentication**, and identity-based access decision making or **authorization**.

- Service-to-service **identity** management is handled through Istio's implementation of the Secure Production Identity Framework for Everyone [23] (SPIFFE). Identities are cryptographically embedded into certificates, which are automatically distributed throughout all the components in the mesh. Lifecycle management, like certificate rotation and revocation, comes out of the box.
- User **identity** management is not included in Istio, but Istio can handle user identity verification or authentication in a variety of ways, as described later.
- Service-to-service (also known as peer) **authentication** [24] is handled through mutual TLS (mTLS). As all Envoy proxy components in the system are equipped with strong cryptographic identities, these certificates are exchanged and verified through the TLS establishment to verify the identity of the caller and the callee.
- User or request **authentication** [25] is based on JSON Web Tokens (JWT). Istio checks the presented token, if presented against the rules in the request authentication policy, and rejects requests with invalid tokens.
- **Authorization**, based on peer or request identities, is provided by a strong and flexible built-in policy engine. Istio offers fine-grained ALLOW and DENY policy configuration based on a variety of parameters.
- Istio provides an external **authorization** API [26] to integrate any type of authentication and authorization mechanism, including OAuth2, OIDC, LDAP, SAML, and more. Solutions like IBM Cloud App ID, Auth0, OPA, Okta, Ping Identity, AWS Cognito, Azure AD B2C, to name a few, all have out-of-the box integrations with Istio.

Audit controls

Istio and Envoy provide integration points for the three pillars of observability: **metrics, logs**, and **distributed tracing**. However, Istio does not ship with its own observability and auditing stack. The building blocks are there, but with the open source version, integration and configuration is up to the customer. TSB addresses these concerns, as we will see later.

Integrity controls

Message integrity is provided by the mTLS capabilities between the proxies in the mesh. With TLSv1.3, perfect forward secrecy (PFS) [27] offers assurances that session keys will not be compromised even if long-term secrets used in the session key exchange are compromised.

Origin or destination integrity and **non-repudiation** are provided by mTLS, which requires clients and servers to prove identity to each other. mTLS can be leveraged as a foundation to implement a zero trust-based approach within your organization, as mentioned above.

Transmission controls

Encryption on the wire can be configured and automatically enforced throughout the whole mesh, without the need for application or system modifications.

Gradual migration towards a fully mTLS enforced mesh is possible in a completely ubiquitous and automated way, which is one of the main reasons service mesh is gaining traction across security-sensitive industries.

Applications that do not natively support encryption can be made HIPAA compliant without any coding changes simply by integrating them into a service mesh.

Tetrate Service Bridge

Tetrate Service Bridge [28] (TSB) is Tetrate's application connectivity platform, built on top of Istio foundations. As mentioned previously in the architectural overview, open source Istio comes with a **control plane** and a **data plane**. But decentralized enforcement, without centralized governance, is only half of the solution. TSB provides the essential element that is missing in this offering: a dedicated **management plane**. Open source solutions often solve lower-level problems while leaving management to commercial products, as with Envoy and Istio.

Management plane

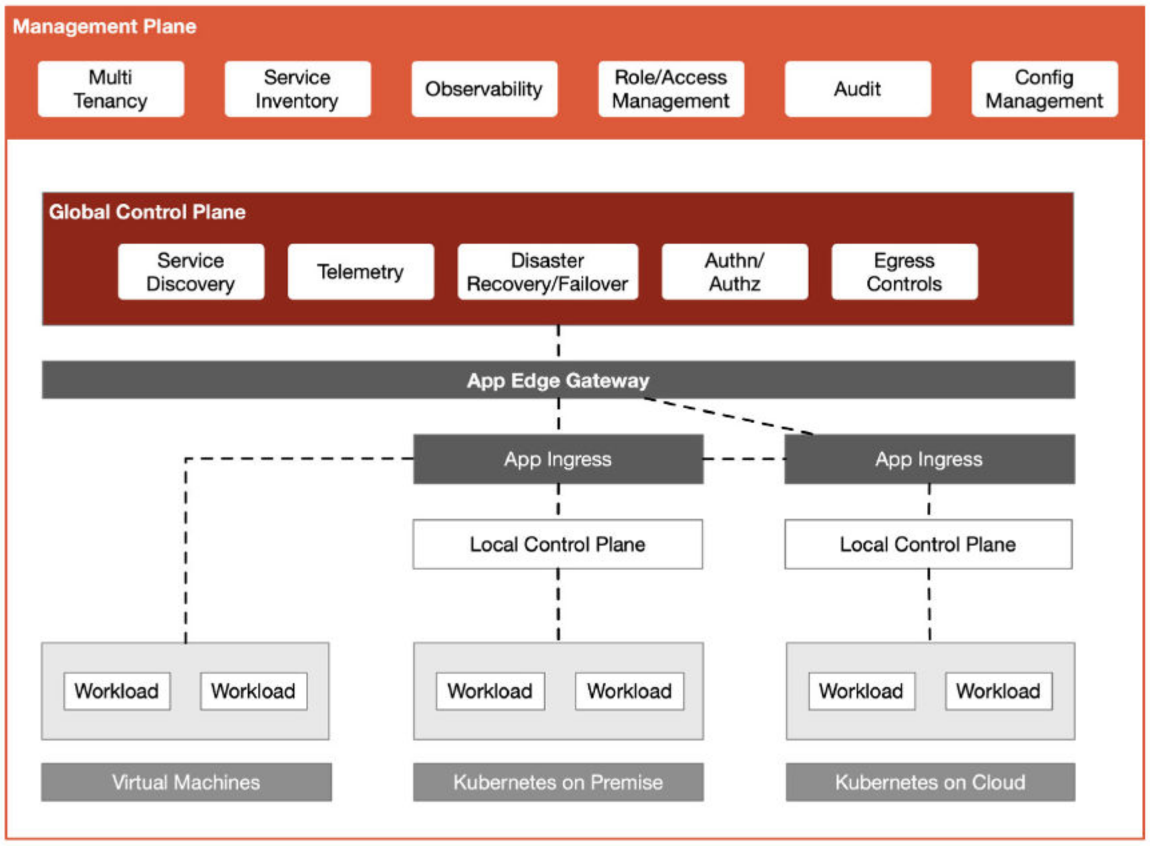


Figure 4: *Tetrate Service Bridge Architecture*

The TSB management plane [29] is your primary access point to everything within your mesh-managed environment.The management plane enables easy management of your environment by splitting up your infrastructure into workspaces, groups, and services. With these logical groupings, TSB offers improved user experience for managing your environment. It allows gateway, traffic, and security-related configuration to be handled by different personae with different roles within your organization or cross organizational.

Any changes that impact your mesh-managed environment are controlled from the management plane, including runtime actions like traffic management, service discovery, service-to-service communication, and ingress/egress controls, as well as administrative actions like managing user access (IAM), security controls, and audit.

Global observability

One of the most powerful benefits of a service mesh [30] is providing consistent operational metrics (RED - Rate, Error, Duration) logging across every application in the mesh, and facilitating distributed tracing. However, when it comes to managing a service mesh for an organization's entire infrastructure (across clusters, clouds, and data centers) you're left to your own devices to piece together a view of the entire world for application developers and security owners.

TSB makes it possible to understand what's happening across an entire application fleet by consolidating the metrics from every cluster enrolled in the service mesh into a single view. This means you can view application communication and topologies across clusters, availability zones, and regions.

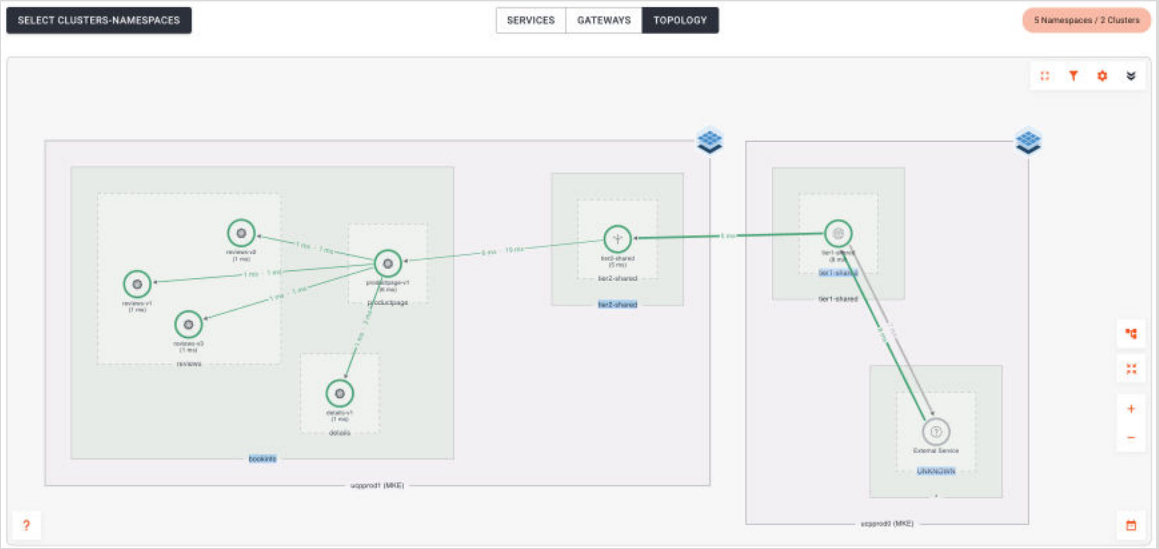
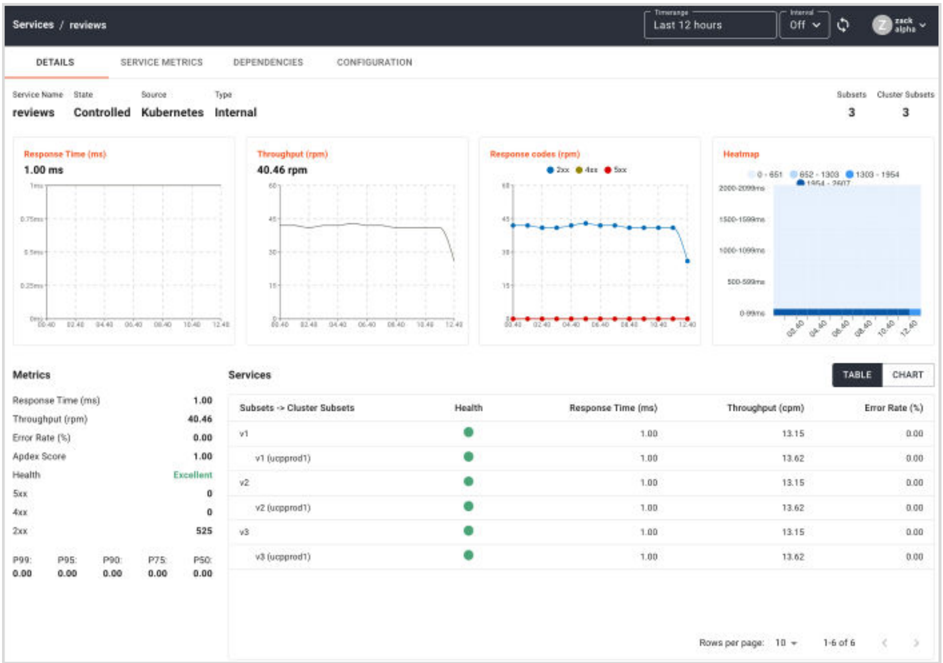


Figure 5: *TSB global topology overview*

TSB also supports a service-centric view, giving a single view of your application's health regardless of where it's deployed or what versions exist.



Service Name	State	Source	Type	Subsets	Cluster Subsets
reviews	Controlled	Kubernetes	Internal	3	3

Metrics

Response Time (ms)	1.00
Throughput (rpm)	40.46
Error Rate (%)	0.00
Apdex Score	1.00
Health	Excellent
5xx	0
4xx	0
2xx	525

P99:	P95:	P90:	P75:	P50:
0.00	0.00	0.00	0.00	0.00

Services

Subsets -> Cluster Subsets	Health	Response Time (ms)	Throughput (cpm)	Error Rate (%)
v1	●	1.00	13.15	0.00
v1 (ucpprod1)	●	1.00	13.62	0.00
v2	●	1.00	13.15	0.00
v2 (ucpprod1)	●	1.00	13.62	0.00
v3	●	1.00	13.15	0.00
v3 (ucpprod1)	●	1.00	13.62	0.00

Rows per page: 10 1-6 of 6

Figure 6: *TSB service observability*

Security

Security is integrated into every part of TSB, and zero trust is the basis for all security within your mesh-managed environment. TSB offers unique security advantages derived from smart user controls, and hardened access policies that you map to your organization's structure. TSB helps to protect your applications and support compliance efforts by making it easier to manage resources and prevent outages.

These security features [31] include:

- Tenancy
- Access control policies in TSB
- Auditability
- Service identities

TSB provides a logic-based approach that takes the policies that your organization needs and maps them to your infrastructure in such a way that it aligns to your organizational setup.

Conclusion

As we have seen throughout this paper, HIPAA compliance has a large focus on the security of ePHI data, as described in the HIPAA security ruling. We have broken down those HIPAA security rules into their different components and mapped them onto features that an open source service mesh like Istio offers.

Although open source Istio adoption can help keep you on track in your journey to HIPAA compliance, we also highlighted some of the missing components:

- Lack of a centralized management plane: decentralized enforcement without **centralized governance** is only half of the solution.
- Lack of **standardized observability and auditing**: to manage a variety of different workloads and vendors, healthcare companies have to integrate them, ideally under a "single pane of glass" management interface, which can be a daunting and expensive endeavor.
- Lack of **fine-grained tenancy**: control over who can do what or who has access to what applications and data is an essential requirement in the HIPAA Security Rule.

Tetrate Service Bridge, Tetrate's enterprise-grade service mesh solution, allows you to take a zero trust stance towards HIPAA compliance. At Tetrate, we look at the world from an application perspective. Applications offer business functionality and access to data, of which ePHI is an important consideration in today's healthcare companies.

As applications, data, hosting form factors (bare metal, IoT devices, VMs, containers and serverless functions), and locality thereof, are getting more and more dispersed, we strongly believe a new paradigm is needed — and is quickly gaining traction. That paradigm is a service mesh driven approach:

- fueled by light-weight, highly performant and highly configurable proxies - **Envoy**
- controlled and meshed together by a modern control plane - **Istio**
- administered and audited by a multi-tenant-first management plane - **TSB**

Although we believe TSB is an all-encompassing solution that would drive you toward HIPAA compliance, we also realize the need for various pathways to adoption. A service mesh solution allows for "all at once" adoption, but also for gradual adoption, by integrating new or existing applications into the mesh on a step-by-step basis, while still maintaining key connectivity and existing security measures in place.

Greenfield scenarios are rare in real life, and at Tetrate we have years of experience in introducing service mesh technology into highly complex and sensitive environments. Leveraging our expertise and product is a must-consider option to **achieve**, **maintain** and **prove** HIPAA compliance.

Get started quickly with Tetrate

Enterprise ready service mesh for any workload on any environment

Contact Us   Schedule Demo

About Tetrate

Tetrate enables a safe and fast modernization journey for enterprises. Built atop Envoy and Istio, its flagship product, Tetrate Service Bridge, spans traditional and modern workloads so customers can get consistent baked-in observability, runtime security, and traffic management—for all their workloads, in any environment. In addition to the technology, Tetrate brings a world-class team that leads the open Envoy and Istio projects, providing best practices and playbooks that enterprises can use to modernize their people and processes.

Location: Tetrate, 691 S Milpitas Blvd, Suite 217, Milpitas, CA 95035, USA
www.tetrate.io | info@tetrate.io