Grey Matter

Security

Authentication, Authorization, Communication, and Segmentation

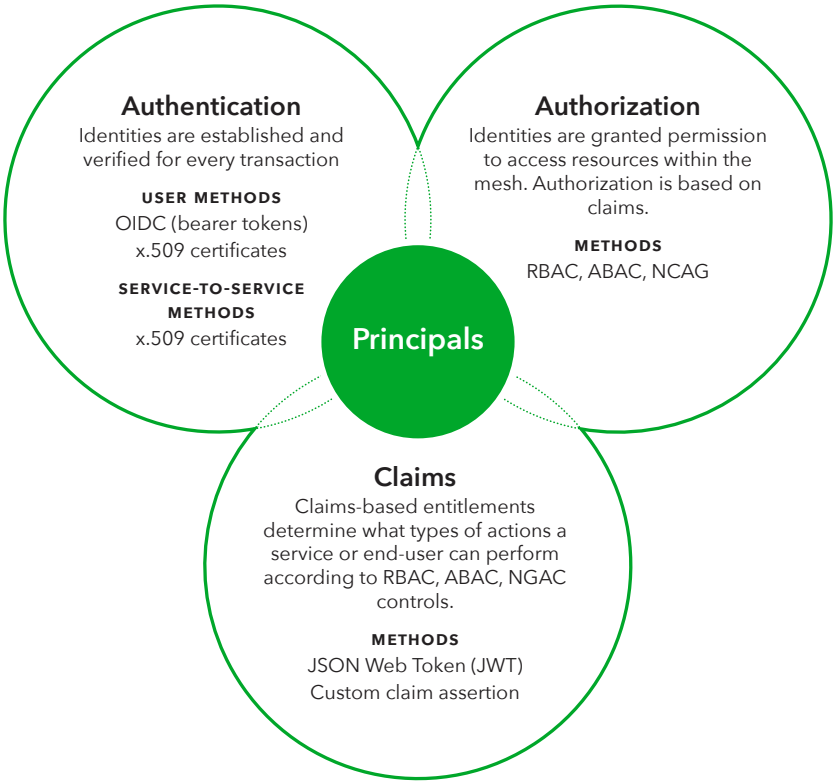
TM & © 2020 Decipher Technology Studios. All rights reserved.

Service meshes, microservices, server-less, and containers are key elements of Mesh application and service architecture (MASA) implementations. MASA, APIs, and internal traffic patterns represent one of the most effective pathways to enterprise modernization, but this doesn't come without challenges.

Industry has signaled increased interest in zero-trust infrastructure for service-to-service mTLS connections, scheduled or on-demand key rotations, service cryptographic identifiers, observability (continuous monitoring, granular audit compliance, etc.), service-level management, and policy management throughout the enterprise service fleet.

Security within Grey Matter

Understanding how the roles of Authentication, Authorization, Claims, and Principals will play within your MASA is important (figure 1). Authentication and authorization are both significant in any security model, but follow different concepts and implementation patterns. *Authentication* establishes and confirms an identity. *Authorization* takes action based on the confirmed identity authenticated. *Principals* are asserted *claims* that provide entitlements granting access to systems, services, or data based on Role-based Access Control (RBAC), Attribute-based Access Control (ABAC) and Next Generation Access Control (NGAC) controls.



Authentication

Grey Matter's authentication scheme establishes identities for every transaction within the platform. There are two types of identities: *users* and *services*.

User Authentication methods:

- OpenID Connect (OIDC)
- mTLS x.509 certificates (Distinguished names represent who the user is)

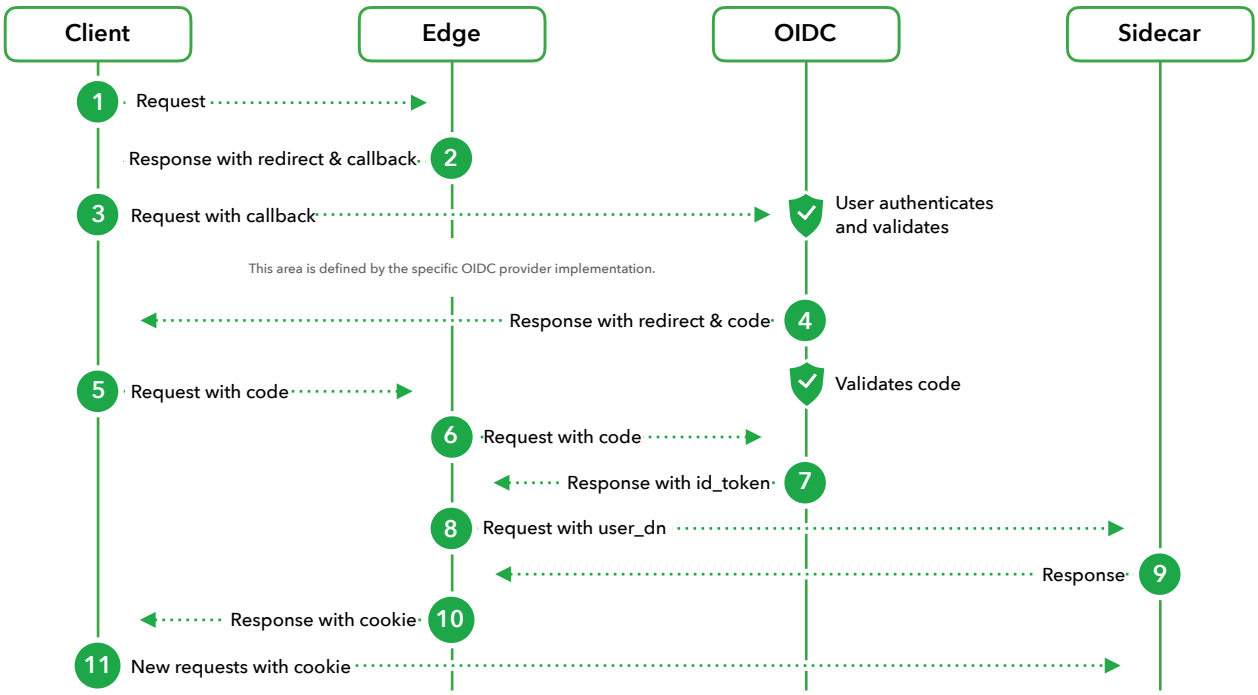
Service-to-Service Authentication methods:

- mTLS x.509 certificates (SPIFFE identities are incorporated into the x.509 certificate)

While distinct, these identities are not mutually exclusive. One of the most common access patterns within Grey Matter is a service making a request to another service on behalf of a user. In this case, there are three identities (two services and a user), each of which must be verified in order for the transaction to succeed. As users or services authenticate with Grey Matter, principals are asserted and flow to upstream services. This ensures that upstream services are aware of the entity (user or service) making a request. Grey Matter supports user authentication and service-to-service authentication methodologies identified below.

User Authentication with an OIDC Provider

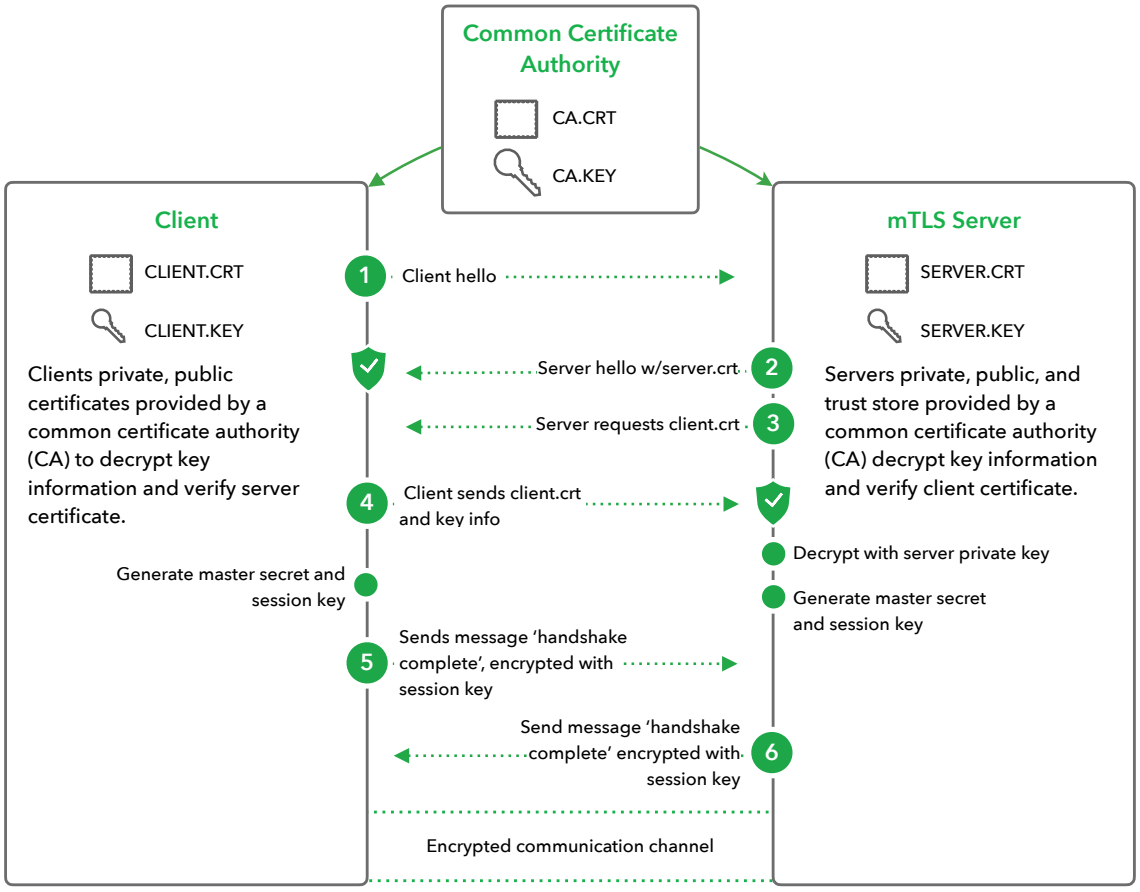
Grey Matter integrates with existing public OIDC providers (Google, Github, etc.) or private OIDC providers (e.g., Ory Hydra) to support user authentication. OIDC is an authentication protocol built on top of OAuth 2.0 that allows delegation of authentication responsibility to a trusted external identity provider. Many implementations of OIDC providers are available and support on premise, cloud or as a service via a host of underlying technologies (e.g., LDAP). This sequence diagram (figure 3) shows the OIDC flow within Grey Matter.



1. The client initiates a request to Grey Matter Edge.
2. Grey Matter Edge responds with a 302 HTTP status code used to perform URL redirection, along with a callback URL.
3. Based on the redirect URL, the client initiates a request to the specified OIDC provider.
4. Once the client is authenticated, the OIDC provider responds with a 302 HTTP code (based on the callback URL) and provides an OIDC code.
5. The client is redirected back to Grey Matter Edge sending the OIDC provided code.
6. The Edge sends the OIDC code to the OIDC provider, validating and verifying the code.
7. Once validating, the OIDC provider sends back the id_token. The id_token claims associated with the user, issuer, and audience.
8. The Edge inspects the id_token and extracts the subject claim and expiration and prepends a user_dn header with the subject claim to the request and forwards it to the upstream sidecar and service.
9. The upstream sidecar and service respond to the request.
10. The edge prepends a signed cookie containing the user_dn and expiration to the response received from the upstream sidecar and service and forwards the response to the client.
11. The client makes an additional request using the signed cookie that allows the edge to extract the user_dn directly up to the point of expiration at which point the client must re-authenticate.

x.509 Certificates

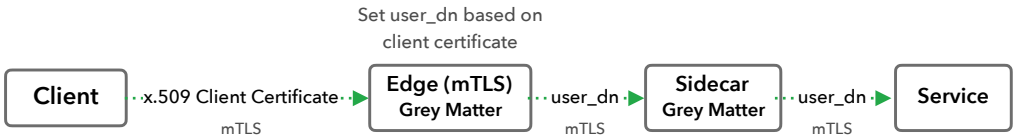
Grey Matter supports x.509 for both users and for service to service transactions.



1. A client (user or service) initiates a request to a server.
2. The server responds with its server certificate.
   2.1. The client verifies that the server's certificate is valid based on its certificate information.
3. The server requests the client's certificate.
4. The client sends its certificate and key information to the server.
   4.1. The server verifies that the client's certificate is valid based on its certificate information.
   4.2. The server is able to decrypt the information sent to it based on the established trust.
5. The client acknowledges that the handshake is complete.
6. The server acknowledges that the handshake is complete.
7. At this stage, the client and server certificates are validated and authenticated. All traffic is now passed through an encrypted communication channel.

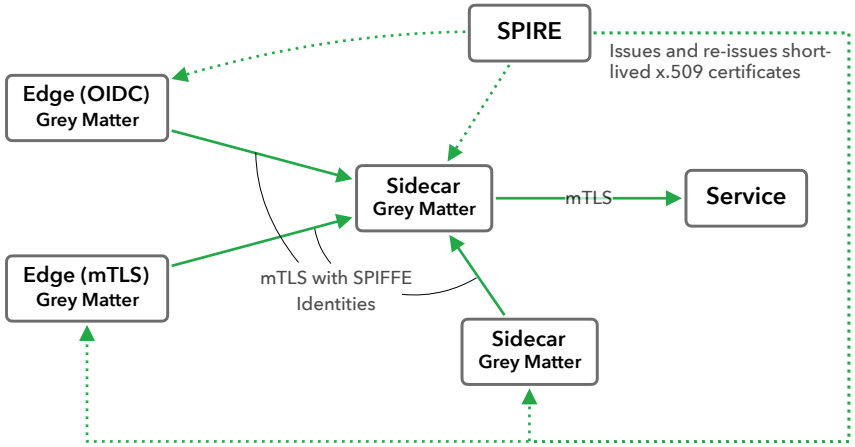
USER AUTHENTICATION WITH X.509 CERTS

Enterprise IT organizations that have existing public key infrastructure (PKI) in place for user authentication can pass their certifications with requests made to the Grey Matter Edge.



SERVICE AUTHENTICATION WITH X.509 CERTS

Service authentication, (service-to-service communication), is based solely upon x.509 certificates and mTLS. Grey Matter Fabric is installed with a certificate authority that issues and reissues short- lived x.509 certificates to each sidecar proxy for intermesh communication. Each certificate contains a SPIFFE identity that uniquely identifies the sidecar to which it is issued. No sidecar will accept a connection from any service that does not present a certificate issued by the certificate authority. Like user authentication, these service identities enable authorization.



Note: In cases where requests already contain a signed cookie the edge simply verifies the signature and expiry. If valid, the edge forwards the request. If not valid, the request is treated as unauthenticated.

Authorization

Authorization is the process by which identities (users or services) are granted permission to access resources within the mesh. For example, we may wish to restrict access to a specific resource to a limited set of users, services or data. As an added complication, it is often more desirable to grant or deny access for a resource to entire classes of identities (i.e., administrative users or trusted services). Grey Matter uses the authenticated identities and their attributes to support fine-grained access controls using the following methods:

- Authorization Filters
- Data Authorization via the Grey Matter Data Platform Service

It's important to note that Sidecar-to-Sidecar (service-to-service) authorization follows similar patterns of a user with the exception that sidecar identities typically do not include additional attributes; however, there is nothing precluding the addition of attributes for a sidecar identity.

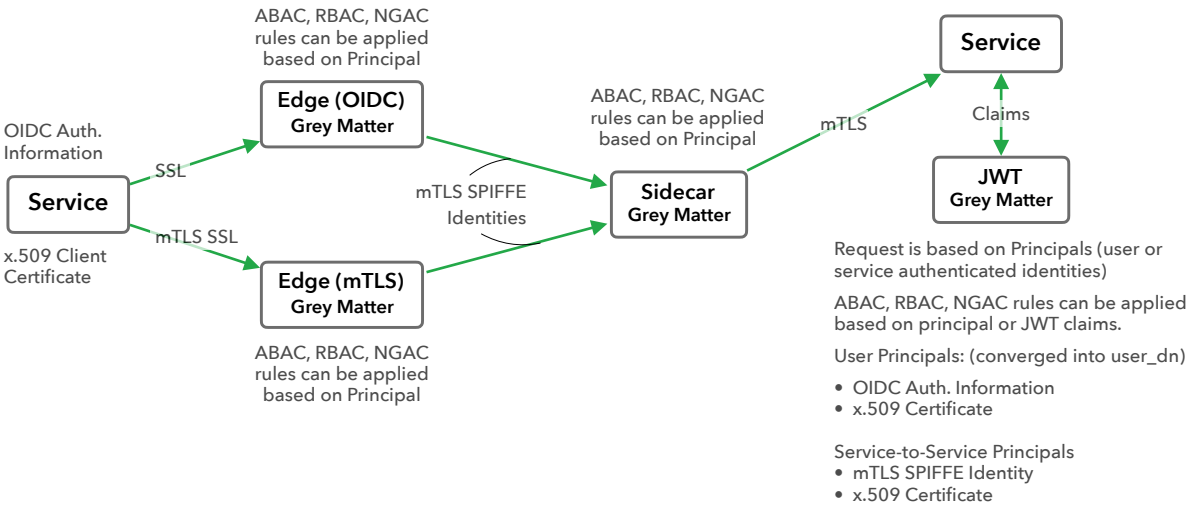
Authorization filters

Upon choosing an authorization pattern, access control becomes a deployment concern, not a development concern. Allowing microservice developers to focus on business value since their services will not receive any unauthorized request. Authenticated identity and attributes are available to the service should they be required.

The Grey Matter Sidecar uses authorization filters to manage who is allowed to access which resources and how. Since all requests to the mesh are authenticated, filters can be dynamically configured at runtime with no additional requirements. Attribute based authorization is also implemented via Grey Matter Sidecar filters but requires that requests contain a signed JSON Web Token (JWT) containing the identity claims. The creation and population of these tokens is left to the enterprise.

List Authorization Filter

The Grey Matter Sidecar supports list-based authorization decisions within the ListAuth filter. This filter allows whitelisting and blacklisting of individual identities based upon the identities distinguished name (i.e, “cn=user, dc=example, dc=com” or “cn=web server, dc=example, dc=com”) or relative distinguished name (i.e., “dc=example, dc=com”). This filter applies to all requests for the proxied service or services.



Role-Based Access Control Filter

The Grey Matter Sidecar supports fine grained authorization decisions to authorize actions by identified clients using Role-Based Access Control (RBAC). This filter allows complex whitelisting and blacklisting of individual identities based upon the identities distinguished name. Matching of regular expressions is supported to add additional flexibility. Further, whereas the ListAuth filter applies to all requests, the Role Based Access Control Filter can be defined for any combination of service, route, or verb. This is useful to explicitly manage callers to a service running within the Grey Matter mesh platform and protect the mesh from unexpected or forbidden agents.

Supported HTTP verbs include:

GET	The GET method requests a representation of the specified resource. Requests using GET should only retrieve data.*
POST	The POST method is used to submit an entity to the specified resource, often causing a change in state or side effects on the server.*
PUT	The PUT method replaces all current representations of the target resource with the request payload.*
PATCH	The PATCH method is used to apply partial modifications to a resource.*
DELETE	The DELETE method deletes the specified resource.*

* Definitions as described by the Mozilla Developer Network (MDN). https://developer.mozilla.org/en-US/docs/Web/HTTP/Methods

In situations where the identity is not sufficient to make all authorization decisions, the Grey Matter Sidecar can enforce finer-grained control based upon identity attributes, provided the request contains a sign JWT.

Using the RBAC filter, rules can be created to authorize specific claims found within a JWT to perform specific actions. This requires an external service to generate a signed JWT for each request. Since the JWT is included as a header, if a JWT is passed, it will propagate to all sidecars in the request chain. With that said, if the request is completed—meaning the destination service has received it, processed it, and invokes another service in the Mesh—this is a new request and the calling service would be required to pass the JWT for further authorization purposes.

Custom Access Control Filter

The Grey Matter sidecar offers a custom filter interface, so customers have the ability to create business-specific logic around their security and regulation concerns if required. This makes the mesh fully adaptabe to an enterprise's needs, and provides a way to take advantage of existing IT investments.

Data Authorization

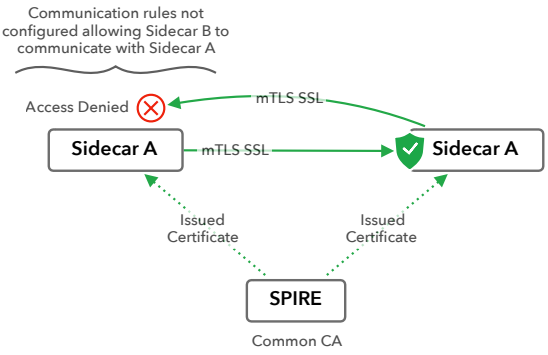
One of the unique facets of Grey Matter is that data security and sharing is addressed by Grey Matter Data. As enterprises shift from monoliths to microservices, data tends to be duplicated across the architecture. Grey Matter Data provides a service to address the secure sharing of this data without marshalling it into and out of processes. This feature is described in greater detail in the Data Segmentation portion of this document, but the pattern employed by Grey Matter Data can be used by any service to enforce complex security policies for any resource via the Grey Matter JWT Security Service or customer JWT service adhering to the Grey Matter Data interface.

Traffic Patterns

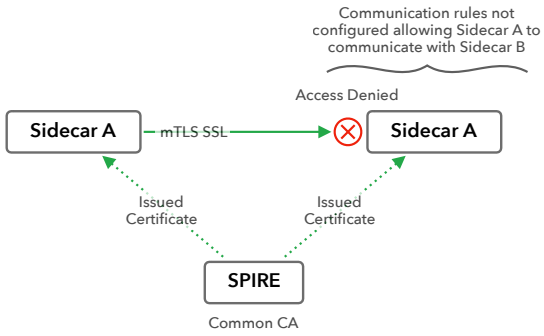
One key feature of the Grey Matter hybrid mesh is its ability to secure, manage, and govern the traffic patterns of running services.

East/West Traffic

East/West traffic within the Fabric Mesh should be done via mTLS. Grey Matter uses two methods enabling this: direct integration with existing CAs and, automatic setup via SPIFFE/SPIRE. For integration with existing CAs, each sidecar in the mesh is configured to use provided x.509 certificates. In the automatic setup, each sidecar uses a unique SPIFFE ID to authenticate with SPIRE servers. Unique short-lived x.509 certificates are then automatically created and rotated for each connection between sidecars.



Sidecar A and sidecar B have been granted mTLS certificates, to establish an encrypted communication channel between each other. However, the mesh was not configured to allow Sidecar A to communicate with Sidecar B. In this scenario, even though Sidecar A and Sidecar B were granted mTLS certificates through a common certificate authority (CA), access is still denied.



The mesh is configured to allow Sidecar A to communicate with Sidecar B. However, Sidecar B is not allowed to communicate with Sidecar A. If Sidecar B tries to send a request to Sidecar A, access is denied.

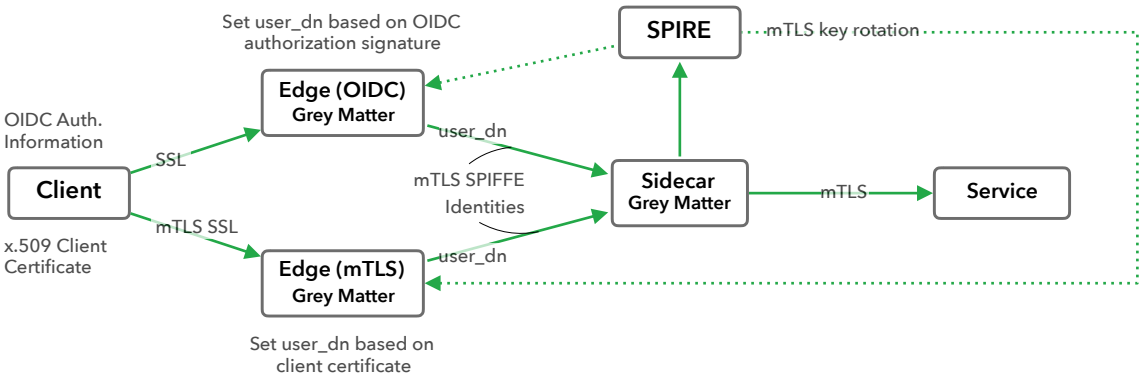
North/South Traffic

North/South traffic patterns use the Grey Matter Edge to establish principals and pass them to called services. The Edge supports both OIDC and mTLS x.509 certificate authentication modes, however, the Fabric Mesh is not limited to a single Edge. Multiple nodes can be configured to expose both authentication modes wherever an access point is needed. Note that the Edge node does not have to be exposed to a publicly addressable URL. In many cases, an API media-

tion layer may be put in front of the Edge node. In all cases, the Edge node is responsible for verifying and ensuring that the proper principals are available for downstream services within the Fabric Mesh to consume.

Principals such as user identities are moved by the Edge node into a user_dn header which flows through the entire service-to-service request chain. Each following link in the request chain is performed via mTLS, with each unique service using automatically rotating x.509 certificates established via SPIFFE Identities and the SPIRE framework.

In some cases traffic needs to flow outside of the mesh. Common scenarios include mesh-to-mesh communications, proxying to serverless functions, and supporting legacy systems that can't be moved directly into the mesh. In all these cases, proxies are setup within the mesh with the sole purpose of communicating outside. Principals are established at the Edge. Inter-mesh communication is still handled by mTLS, and requests are authenticated via the outside system by whatever method it accepts: rpc, http, mTLS, or OIDC.



Traffic Splitting

Traffic splitting is another important pattern in stable environments. Traffic splitting allows configurable service requests to siphon off percentages of requests to another source. This allows services, apps, or entire meshes to experience small amounts of live traffic while keeping most users on the original source. The percentage of users on the original service is then decreased until the service is fully migrated.

Circuit Breaking

Circuit Breaking is a way for each sidecar to protect the thing it is proxying to, but it is not a way to have that proxy harden itself. Grey Matter provides circuit breakers at every point in the mesh.

The most common place for this to occur is at the edge, where a DDOS could overwhelm the edge nodes themselves. To solve this, we employ Rate Limiting, which can protect the edge node from accepting too many requests and opening too many file handles and crashing. With proper configuration, each sidecar ceases queueing new requests before they're overwhelmed, allowing the service time to heal. This ensures capabilities can withstand malicious attacks and accidental recursive network calls without going down.

Network Segmentation

Enterprises prefer hybrid environments capable of leveraging unified on-premise and cloud resources. Traditional networking patterns use features such as VLANs to create perimeter-based firewalls, but this concept breaks down with modern mesh application service architecture (MASA) patterns. In MASA, services are designed to be ephemeral, dynamically generating different IP:PORT pairs each time a new instance spins up.

Securing this type of architecture requires network segmentation. Grey Matter isolates services and network fabric communications to specific runtime environments or infrastructure resources. Grey Matter Fabric supports segmentation to a very fine level of granularity. Each service launched onto Fabric comes online with no knowledge of or connections to any other point on the mesh. The desired mesh is then built up through configuration with the required network topology. Segmentation is enforced through routing rules, service discovery, and mTLS. Dynamic configuration can facilitate any permutation of intra-mesh communication required. In addition to segmentation of individual meshes, Grey Matter can also support multi-mesh operations. This allows the bridging of environments already physically or logically isolated from each other.

Micro-segmentation

Micro-segmentation is a method of creating secure isolation zones either on-premise or in the cloud in order to separate different workloads. Authentication plays a key role in micro-segmentation. Authentication is responsible for establishing network communications and flow through the mesh. Strong authentication models enable Grey Matter to perform micro-segmentation for users, services, and data throughout the mesh.

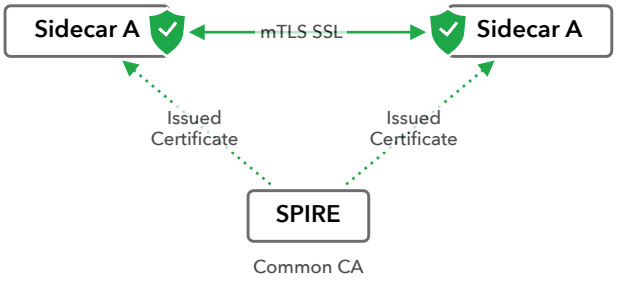
User-to-Service segmentation is controlled through user authorization signatures. These can be coupled with claim-based assertions. User identities and claims flow through mTLS-encrypted communication channels established by service-to-service micro-segmentation patterns. Complex security policies within each sidecar allow ABAC/RBAC down to the service, route, and HTTP verb permit. This enables a very high degree of isolation. ABAC/RBAC policies cannot be achieved without strong authentication methodologies establishing identities for both users and services.

Service-to-Service segmentation is controlled through mTLS certificates and SPIFFE identities. These can be coupled with claims-based assertions and ABAC/RBAC policies. Images in the following section illustrate how this is achieved.

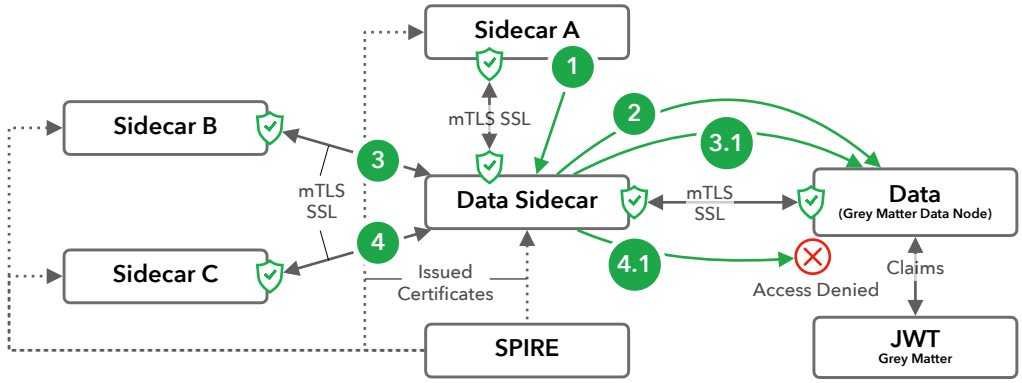
Data Segmentation

Grey Matter's data segmentation capability is a key differentiator. Data segmentation is the process of dividing data and grouping it with similar data based on set parameters. Grey Matter Data adds complex policy assertions to stored objects. These object policies govern which users or services may access the objects. Objects stored within Grey Matter Data are encrypted at rest and in transit. A JSON Web Token (JWT) is provided to gain access to an object stored in Data.

The token's claims are dynamically mapped to the policies stored with the object. JWTs for both users and services can be created enabling end-to-end security using authentication principals.



The example above shows how data-segmentation is achieved through simple policy. However, the Grey Matter Data policy engine is designed to deal with complex rules designed to suit any scenario. The following scenario presents a more complex use case.



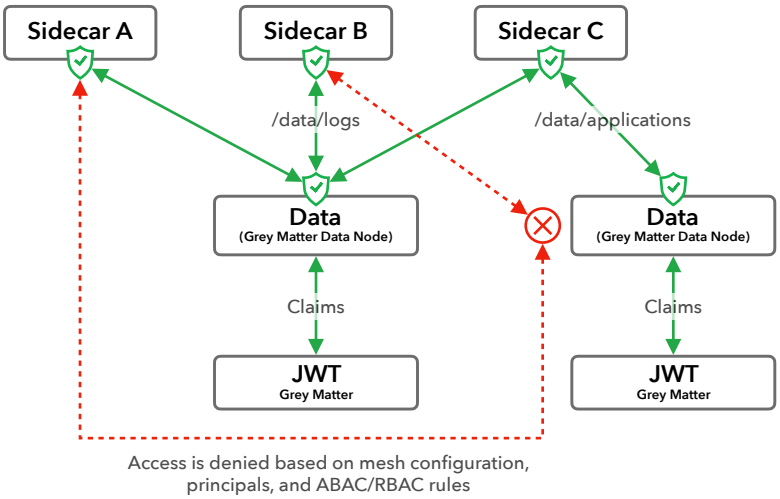
1. Sidecar A saves an object into Data and provides access privileges to Sidecar B's SPIFFE identity. Sidecar A dynamically discovers Data via the Data Sidecar routing information.
2. Data Sidecar receives Sidecar A's request and streams the object (with policy) into the Grey Matter Data node.
3. Sidecar B (through a means of event-based architecture patterns) is notified that Sidecar A just saved an object of interest into Data. Sidecar B calls into Data (through the Data Sidecar) to retrieve the object. Sidecar B's SPIFFE identity is passed along with the request.

   3.1. Data Sidecar receives the request from Sidecar B and passes it to Data. Data uses the Sidecar B principal (i.e. SPIFFE identity) to receive Sidecar B's JWT claims and authorize access to decrypt and retrieve the object.
4. Sidecar C is an outlier listening for arbitrary events. Based on the event broadcasted, Sidecar C attempts to retrieve the encrypted object stored in Data. Sidecar C is entitled to talk to Data via the Data Sidecar but does not have access to all data stored.
5. Data Sidecar receives the request from Sidecar C and passes it to Data. Using Sidecar C's principal (i.e. SPIFFE identity) Data retrieves its corresponding JWT claims and denies access to the object stored.

Since Grey Matter uses a unified principal model, data segmentation can be achieved for users as well. Grey Matter Data policies can be set to identify different access privileges for services and users on a single stored object, and can be customized around business needs. This paradigm provides a new model that combines network, information assurance, and protection concepts around zero-trust.

Grey Matter Data supports the ability to host multiple Data nodes available through different routing rules. When coupled with other segmentation features, enterprises are able to further isolate how information is stored, accessed, and controlled based on customer regulations and requirements.

For example, logs and observable traffic can be isolated based on zones. Data nodes with specific routing rules and policies are set to enforce the topology. Customer application data can be stored and accessed via different Data nodes (on-premise or in the cloud) and tightly controlled at the micro-segmentation layer or via data policies. These types of flows are depicted in the diagram below.



Conclusion

Grey Matter's zero-trust threat model ensures security across every service in the hybrid mesh. Each transaction is authenticated and authorized through a combination of mTLS and SPIFFE authentication and SPIRE authorization providing multiple layers of zero-trust security. Grey Matter also supports fine-grained access control by combining authenticated identities with policy-enforced object authorization and enables East-West and North-South traffic pattern splitting and shadowing for in-depth monitoring and configuration. Finally, Grey Matter uses network and data segmentation to decompose operations to their most basic elements, to mitigate cyber intrusion impacts, and to optimize operations.

DECIPHER
TECHNOLOGY STUDIOS

Decipher Technology Studios builds Grey Matter, the intelligent hybrid mesh platform for enterprise microservice, container, and hybrid cloud operations.

Decipher Technology Studios
110 S. Union Street, Floor 2
Alexandria, VA 22314

(877) 356-3011
deciphernow.com

TM & © 2020 Decipher Technology Studios. All rights reserved.