How SPIFFE helps Istio in Service Mesh Federation

Yonggang (Oliver) Liu yonggangl@google.com
Wencheng Lu wlu@google.com
What is Istio?

A service mesh. But more: An open services platform to manage service interactions across container- and VM-based workloads

- Uniform observability
- Operational agility
- Policy driven security
What is SPIFFE?

A set of open-source standards to provide a secure production identity framework in a heterogeneous environment.

- SVID (SPIFFE Verifiable Identity Document): Standardize SPIFFE identity in X.509 certificate format
- SPIFFE APIs: A set of APIs and specs to describe how to securely provision and federate SPIFFE identities
A secure identity framework to provide strong identities for service-to-service authentication

- SVID compliant
- Support SPIFFE federation API: work in progress
Why Service Mesh Federation Matters?

Provide the interoperability between:

- Service meshes from different Orgs
- Service meshes from the same Org
- Service meshes from different Cloud and On-prem
- Service meshes from different env: k8s mesh and VM mesh
Identity Federation Challenges

- Build identity trust between meshes
- Identity isolations
Service Mesh and Trust Domain

- In current Istio, the applications in a service mesh share common roots of trust and the same trust domain.
- A trust domain could represent an individual, organization, environment or department running their own independent SPIFFE infrastructure.
- The trust domain is encoded in Istio/SPIFFE identities: `spiffe://<trust_domain>/ns/<K8s namespace>/sa/<K8s service account>`
Federation of Meshes

- For the applications in two meshes to authenticate, they need to verify each other’s certificates using their own trusted roots.

Service Mesh 1
Trust domain: foo.com

Service Mesh 2
Trust domain: bar.com
Federation within an Organization

- Common root CA
- Intermediate CA name constraints help to isolate trust domains
Federation across Organizations (1)

Service Mesh 1
Trust domain: foo.com

Service Mesh 2
Trust domain: bar.com

Root CA1
Service A
Trust:

Root CA2
Service B
Trust:

root cert for CA1
cert for Service A
root cert for CA2
cert for Service B
Root CA Cross Certification

- High complexity and not scalable. $O(N^2)$ cross-signings for $N$ trust domains
Federation across Organizations (2)

SPIFFE Trust Bundle (recommended)

- Automated root of trust exchange
- Authentication using the root certs corresponding to the peer’s trust domain
A SPIFFE trust bundle is an RFC 7517 compliant JWK set containing a trust domain's cryptographic keys, for the validation of certificates issued in that trust domain.
SPIFFE Trust Bundle Example

```json
{
  "keys": [
    {
      "use": "x509-svid",
      "kty": "EC",
      "crv": "P-256",
      "x": "fK-wKTnKL7KFLM27lqq5DC-bxrVaH6rDV-IcCSE0eL4",
      "y": "wq-g3TQwxY1V51TCPH030yXsRxvujD4hUuaIqrXk4KI",
      "x5c": [
        "MIIBkjCB0aADAgECAgEBMAoGCCqGSM49BAMCMAAwIhgPMDAwMTAxMDAwMDBaGA85OTk5MTIzMTIzNTk1OvwADBZMBMByqGSM49AgEGCCqGSM49AwEHA0IABHyvsCk5yi+yhsz5u5aquQwvm88a1wh+qw1fiHAKhDni+wq+g3TQwxY1V51TCPH030yXsRxvujD4hUuaIqrXk4KKjODA2MA8GA1uEwEB/wQFMAMBAf8wJwYDVR0RAQH/BBKwF4YVc3BpZmZl0i8vZ9tYW1uM55QXN0MAoGCCqGSM49BAMCA0gAMEUCIA2d009Xmakw2ekuHKWC4hBhCkpr5qY4bI8YUCXfxg/1AiEA67kMy7bQmnr70VLUrL+b9y1AdZg155kKnYigmwDh+/U=
      ]
    }
  ],
  "spiffe_refresh_hint": 600
}
```
Federation with SPIFFE Trust Bundle (1)

Publishing the trust bundle

- **HTTPS** endpoint, TLS cert based on WebPKI or SPIFFE

Consuming the trust bundle

- Admin configures the `<trust_domain, endpoint>` mapping
- Istio authenticates the HTTPS endpoint and retrieves the bundle
- `<trust_domain, bundle>` tuples are propagated to each workload and used in cert verification
Federation with SPIFFE Trust Bundle (2)

Service Mesh 1
Trust domain: foo.com

Service A

Service Mesh 2
Trust domain: bar.com

Service B

Trust bundle management module

Trust bundle config:
“foo.com”: local
“bar.com”: “https://endpoint_2”

Trust bundle config:
“bar.com”: local
“foo.com”: “https://endpoint_1”

https://endpoint_1

https://endpoint_2

cert for Interm. CA1
cert for Service A
cert for Interm. CA2
cert for Service B
Federation with SPIFFE Trust Bundle (2)

Service Mesh 1
Trust domain: foo.com

Trust bundle config:
“foo.com”: local
“bar.com”: “https://endpoint_2”

Service A

Retrieving trust bundles

https://endpoint_2

Trust management module

https://endpoint_1

Trust bundle config:
“bar.com”: local
“foo.com”: “https://endpoint_1”

SPIRE Server

Service B

Service Mesh 2
Trust domain: bar.com

cert for Interm. CA1

cert for Service A

cert for Interm. CA2

cert for Service B
Federation with SPIFFE Trust Bundle (2)

Service Mesh 1
Trust domain: foo.com

Service Mesh 2
Trust domain: bar.com

Trust bundle: foo.com ← bar.com ←
Trust bundle config:
“foo.com”: local
“bar.com”: “https://endpoint_2”

Trust bundle:
cert for Service A

cert for Interm. CA1

cert for Interm. CA2

cert for Service B

Trust management module

Trust bundle config:
“bar.com”: local
“foo.com”: “https://endpoint_1”

https://endpoint_1

https://endpoint_2

mTLS
Thank you!

Istio docs

[istio.io](https://istio.io)

Istio discussion board

[discuss.istio.io](https://discuss.istio.io)

Join Istio working groups


Contribute code

[github.com/istio](https://github.com/istio)