
KubeCon
CloudNativeCon
North America 2024

Venafi
A CyberArk Company

KubeCon 2024 **SALT LAKE CITY**

SPIFFE the Easy Way :

universal X.509 and JWT identities using cert-manager

Tim Ramlot
Senior Software Engineer

Ashley Davis
Staff Software Engineer

cert-manager

CNCF graduated project

427+ contributors

12k GitHub stars

20M monthly chart downloads

Universal identities

SPIFFE IDs are simple.

SVIDs are platform agnostic.

Supports both X.509 and JWT

X.509 and JWT have their differences, but together have very wide support

Simple Workload API

The SPIFFE Workload API standardizes how to obtain SVIDs and trust bundles.

Clients that support SPIFFE

- Any platform supporting JWT auth *
- Any platform supporting X.509 auth *
- cosign
- in-toto
- cilium operator
- Venafi
- Hashicorp Vault
- AWS / GCP / Azure / others
- More to come!

What are we trying to do?

- SPIRE is great!
- Good to explore alternative implementations
- Helps us learn
- Shows what can be improved

SPIRE

Specifications are best with multiple implementations

- Requires stateful DB deployment
  high operational overhead
- Platform agnostic
  also works outside of Kubernetes
- Supports JWT-SVID and X.509-SVID
- Provides SPIFFE Workload API

VS

cert-manager

- Operational simplicity
  state is kept in Kubernetes resources only
- Runs on Kubernetes only

Universal identities (X.509-SVID)

Obtaining an SVID from SPIRE

```
./spire-agent api fetch x509 \
   -socketPath /run/spire/sockets/agent.sock \
   -write /tmp/
```

Obtaining an SVID using openssl

```
openssl req -x509 -newkey rsa:4096 \
        -CAkey ca_key.pem -CA ca_cert.pem \
        -keyout leaf1_key.pem -out leaf1_cert.pem -noenc \
        -days 1 -subj "/" \
        -addext "subjectAltName = URI:spiffe://venafi.com/workload001"
```

OpenSSL
Cryptography and SSL/TLS Toolkit

Obtaining an SVID using cert-manager

```
apiVersion: cert-manager.io/v1
kind: Certificate
...
spec:
  ...

  uris:
    - spiffe://venafi.com/workload001

  issuerRef:
    name: ca-issuer
    kind: Issuer
    group: cert-manager.io
```

cert-manager

Obtaining an SVID using cert-manager csi-driver

```
apiVersion: v1
kind: Pod
…
spec:
 …
 volumes:
    - name: spiffe
      csi:
        driver: csi.cert-manager.io
        readOnly: true
        volumeAttributes:
          csi.cert-manager.io/issuer-name: workload-issuer
          csi.cert-manager.io/issuer-kind: ClusterIssuer
          csi.cert-manager.io/issuer-group: cert-manager.io

          csi.cert-manager.io/uri-sans: spiffe://venafi.com/workload001
```

cert-manager
csi-driver

Obtaining an SVID using cert-manager csi-driver-spiffe

```
apiVersion: v1
kind: Pod
…
spec:
 …
 volumes:
    - name: spiffe
      csi:
        driver: spiffe.csi.cert-manager.io
        readOnly: true
```

cert-manager
csi-driver-spiffe

SPIRE

- Requires stateful DB deployment
  high operational overhead
- Platform agnostic
  also works outside of Kubernetes
- Supports JWT-SVID and X.509-SVID
- Provides SPIFFE Workload API

VS

cert-manager

- Operational simplicity
  state is kept in Kubernetes resources only
- Runs on Kubernetes only
- **Supports X.509-SVID directly**

Exchanging identity proofs (X.509-SVID to JWT-SVID)

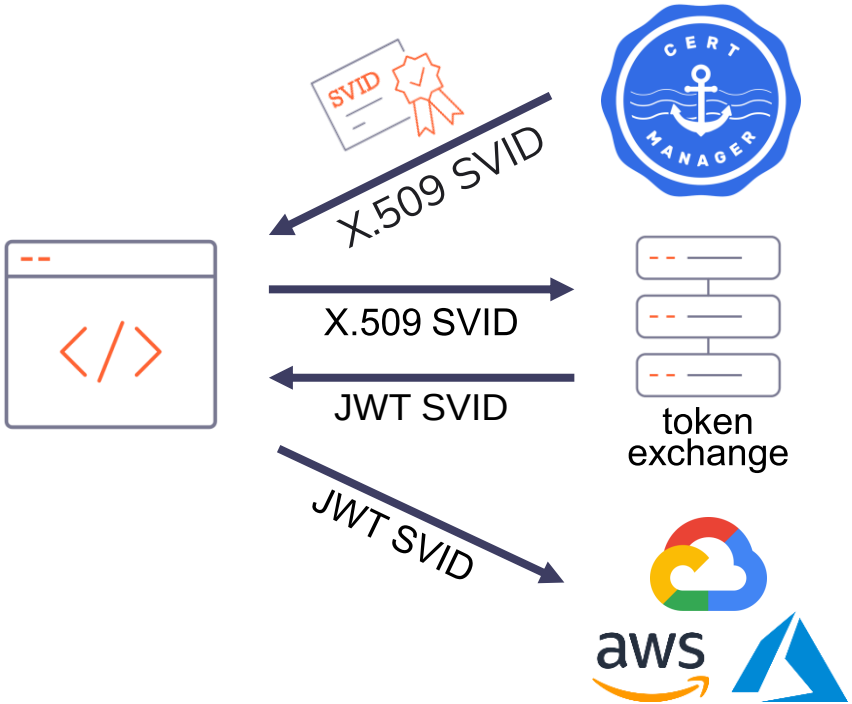
The JWT-SVID Problem

- cert-manager only handles X.509 certs
- No support for signing JWTs / handling JWT keys
- Ideally want to avoid doubling the keys we have to manage
- We have a potential solution…

token-exchange

What if we exchanged X.509 SVIDs for JWT SVIDs?

Following the SVID

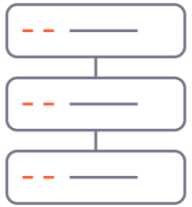


Converting X.509 SVIDs to JWT SVIDs

mTLS using
X.509 SVID

POST /token

token.<token-exchange>

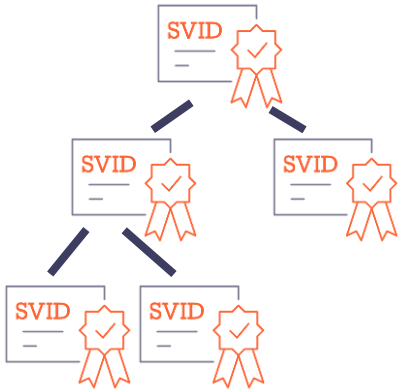


response
JWT SVID

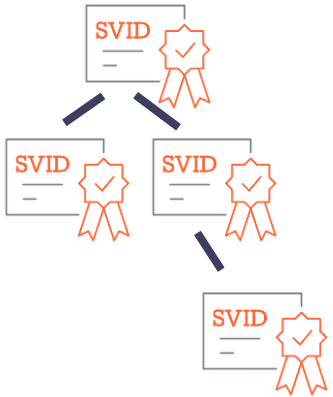
```
{
  "iss": "https://discover.<token-exchange>/2bf...30c",
  "sub": "spiffe://venafi.com/workload001",
  "aud": [
    "aws"
  ],
  "exp": 1730376027,
  "iat": 1730372427
}
```

Mapping X.509 CAs to JWKSs

`https://discover.<token-exchange>/<hash(root CA certificate)>`



`iss: https://discover.<token-exchange>/8898a8a3459079ed8a03f66c2a...`



`iss: https://discover.<token-exchange>/450dfbd984f8c69cc1a4c010ca...`

Trusting the OIDC endpoint in GCP/ AWS/ Azure

GET
JWKS by following
discovery endpoint

discover.<token-exchange>

/2bf...30c/.well-known/openid-configuration

/2bf...30c/.well-known/jwks

Follow the guide to setup your Cloud Provider: https://github.com/Venafi/token-exchange/tree/82ed21fbc6b5f24054d51113ccb0a426b3fe81de/client/guide

SPIRE

- Requires stateful DB deployment
  high operational overhead
- Platform agnostic
  also works outside of Kubernetes
- Supports JWT-SVID and X.509-SVID
- Provides SPIFFE Workload API

VS

cert-manager

- Operational simplicity
  state is kept in Kubernetes resources only
- Runs on Kubernetes only
- Supports X.509-SVID directly
  (+ supports JWT-SVID)

SPIFFE Workload API

SPIFFE Workload API

- Must be implemented so we can be SPIFFE “compliant”
- Specifies both X.509 and JWT endpoints
- For X.509 SVIDs, just return the cert from csi-driver
- For JWT SVIDs, use the cert to talk to token-exchange

SPIFFE Workload API

- Token-exchange enables the workload API
- We also need to handle trust, which we can do with trust-manager
  - Detail on trust-manager is out of scope of this talk
  - You should check trust-manager out!
- Token-exchange handles JWT trust for us

- Requires stateful DB deployment
  high operational overhead
- Platform agnostic
  also works outside of Kubernetes
- Supports JWT-SVID and X.509-SVID
- Provides SPIFFE Workload API

VS

cert-manager

- Operational simplicity
  state is kept in Kubernetes resources only
- Runs on Kubernetes only
- Supports X.509-SVID directly
  (+ supports JWT-SVID)
- **Provides SPIFFE Workload API**

DEMO

Demo Recap

- We only needed one cert-manager cert!
- Exchanged JWTs for each cloud provider
- Pluggable into any cluster running cert-manager
- Powerful approach that can scale

Trust Model

- SPIRE requires you to register a workload for a given SPIFFE ID
- csi-driver-spiffe forces SPIFFE IDs to match the pod
  - No registration step
- Pod mappings can be restricted using approver-policy
  - Out of scope for this talk
  - Enables secure use of csi-driver
- We move trust from the SPIRE database to Kubernetes RBAC

Feedback?

Venafi

Thank you!

Reach out to us on Kubernetes Slack/ via your Venafi contact.
Code can be found on https://github.com/venafi/token-exchange

Tim Ramlot
Senior Software Engineer

Ashley Davis
Staff Software Engineer