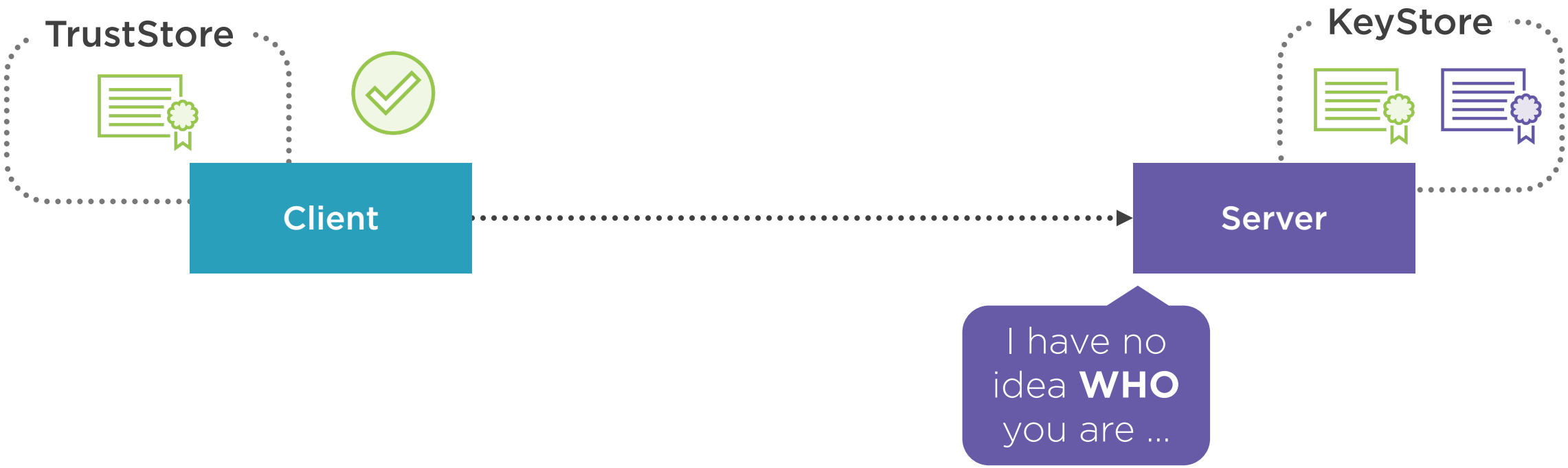
Authenticating Clients

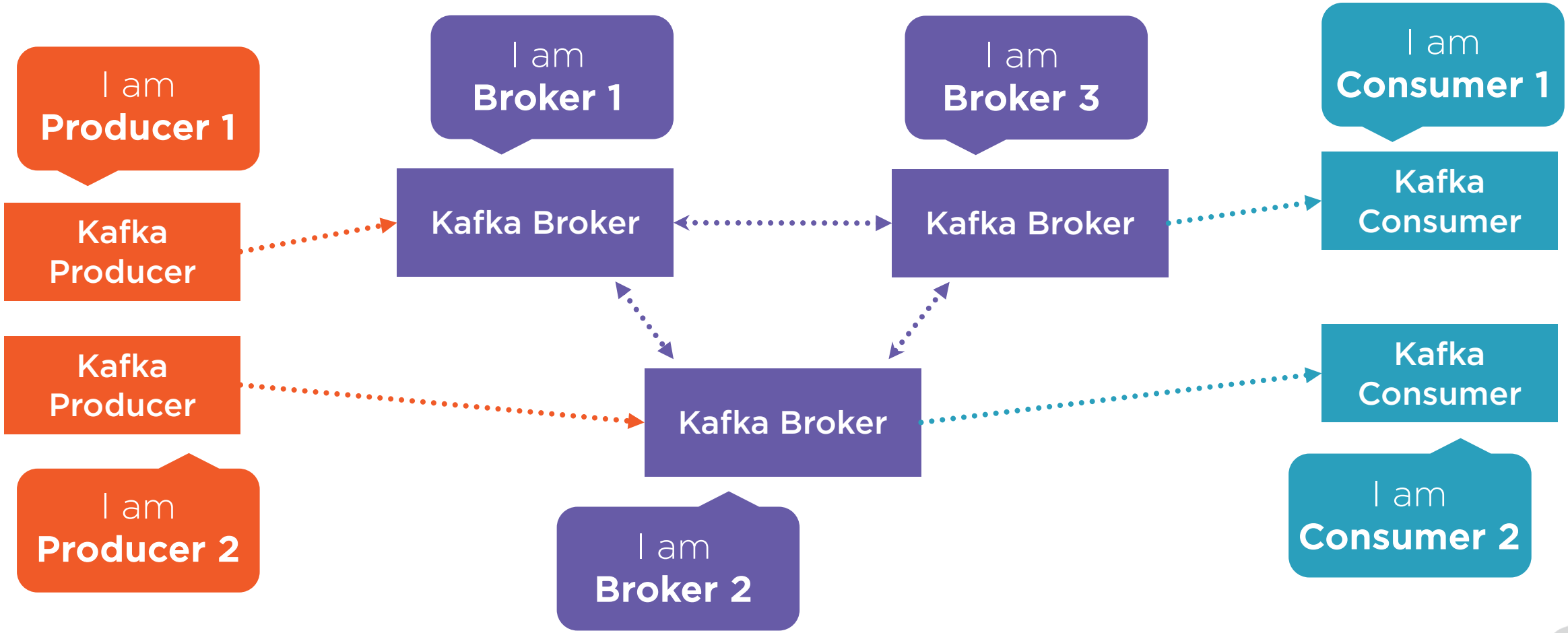
Bogdan Sucaciu
SOFTWARE ENGINEER

@BSucaciu bsucaciu.com

Authenticating Clients



Authenticating Clients



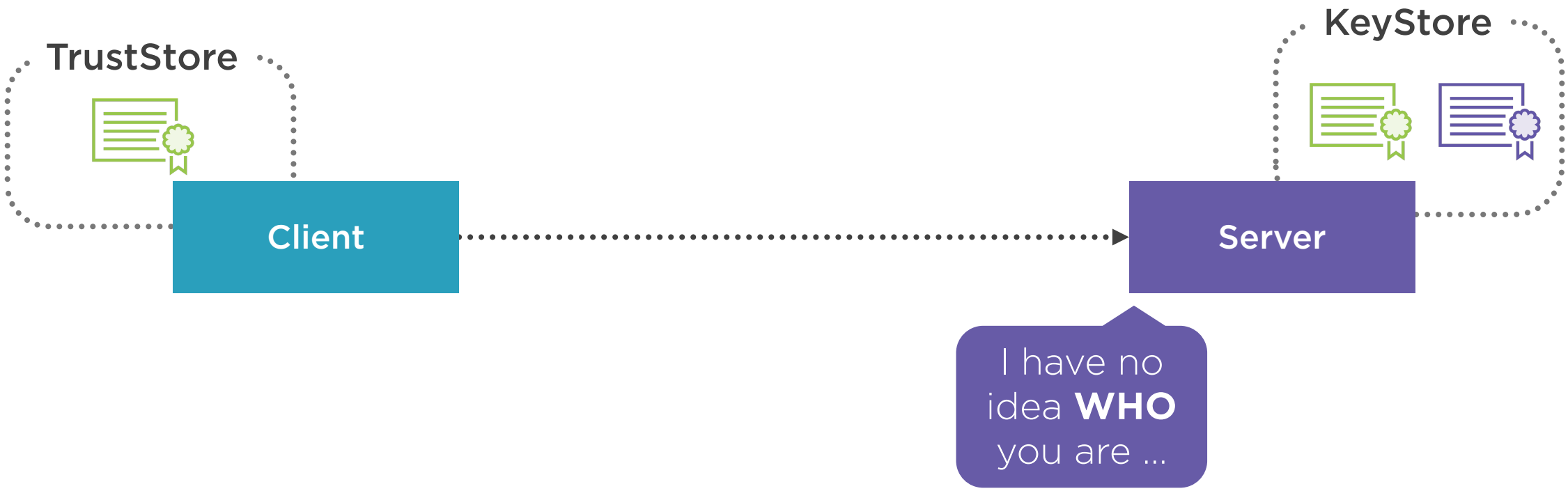
Kafka Authentication Frameworks

mTLS

SASL

kafka.apache.org/documentation/#security_overview

Mutual TLS


TrustStore
KeyStore
Client
Server
I have no idea **WHO** you are ...

Mutual TLS

Demo

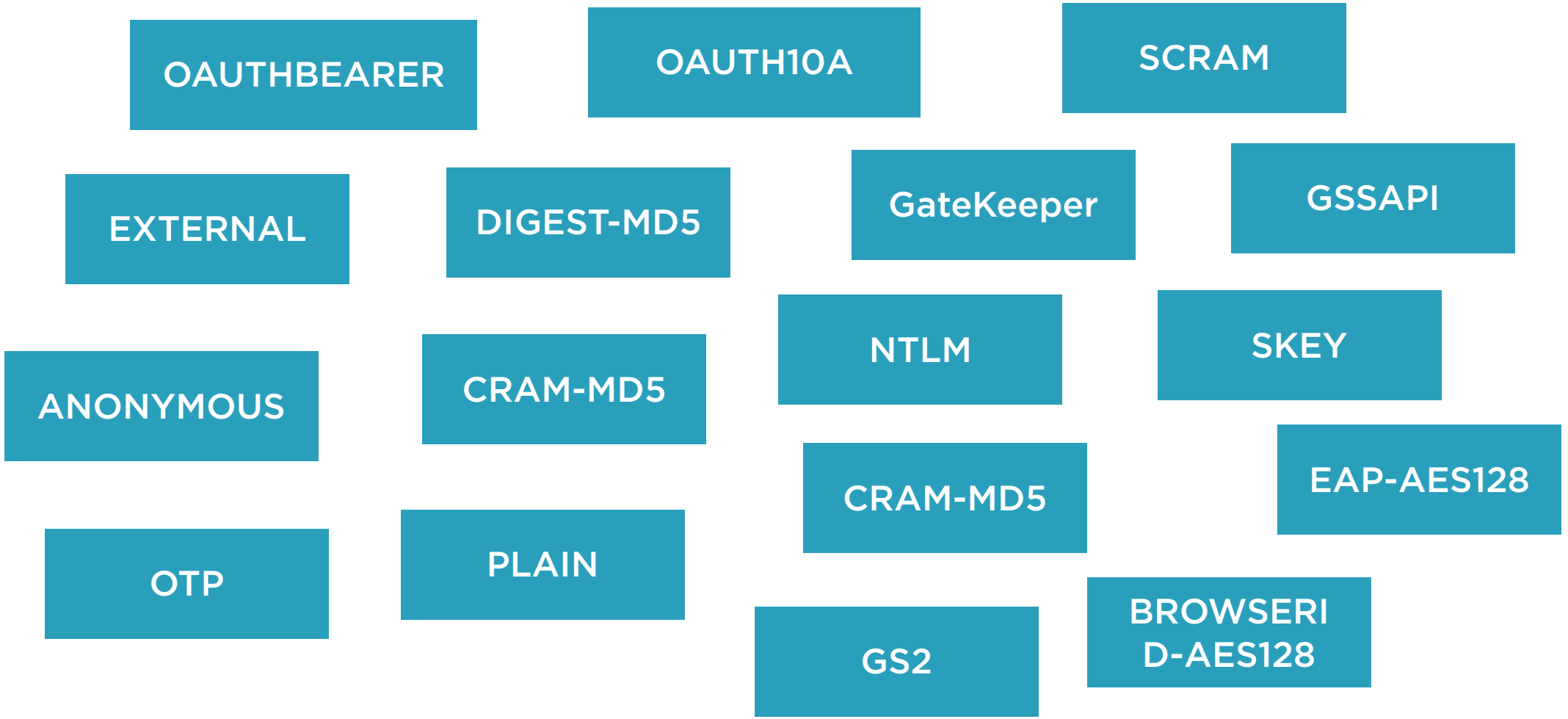
Set up mTLS

SASL

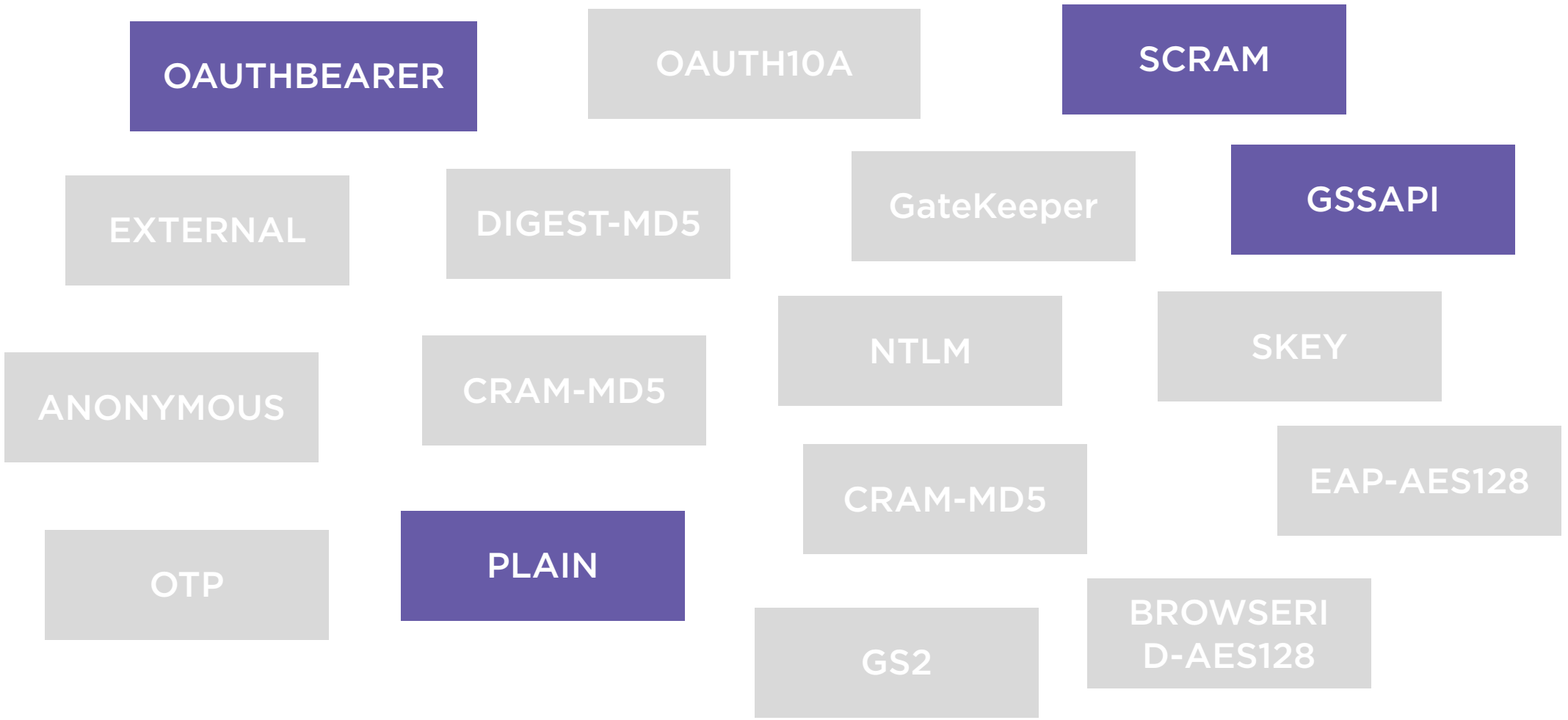
Simple **A**uthentication

and **S**ecurity **L**ayer

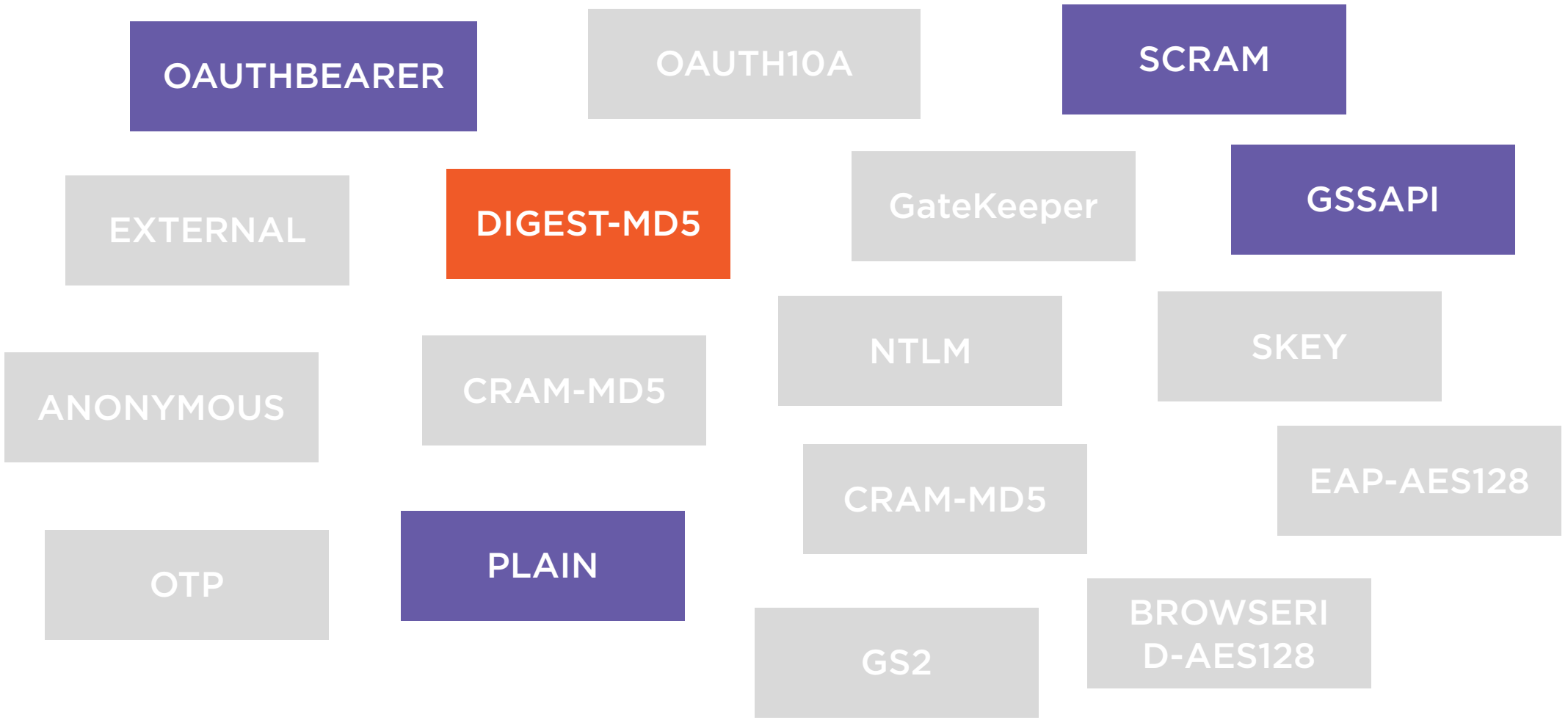
SASL Mechanisms



SASL Mechanisms



SASL Mechanisms



JAAS

Java **A**uthentication and **A**uthorization **S**ervice is the Java implementation of the Pluggable Authentication Module (PAM)

JAAS Configuration Files

Authentication

*.conf

Authorization

*.policy

JAAS Configuration Files

PLAIN

SCRAM

OAUTHBEARER

GSSAPI

```
KafkaServer {


};
```

kafka_jaas.conf

JAAS Configuration Files

PLAIN | SCRAM | OAUTHBEARER | GSSAPI

```
KafkaServer {
      org.apache.kafka.common.security.plain.PlainLoginModule required
      .....
};
```

kafka_jaas.conf

JAAS Configuration Files

PLAIN | SCRAM | OAUTHBEARER | GSSAPI

```
KafkaServer {
      org.apache.kafka.common.security.scram.ScramLoginModule required
      .....
};
```

kafka_jaas.conf

JAAS Configuration Files

PLAIN | SCRAM | **OAUTHBEARER** | GSSAPI

```
KafkaServer {
      org.apache.kafka.common.security.oauthbearer.OAuthBearerLoginModule required
      ......
};
```

kafka_jaas.conf

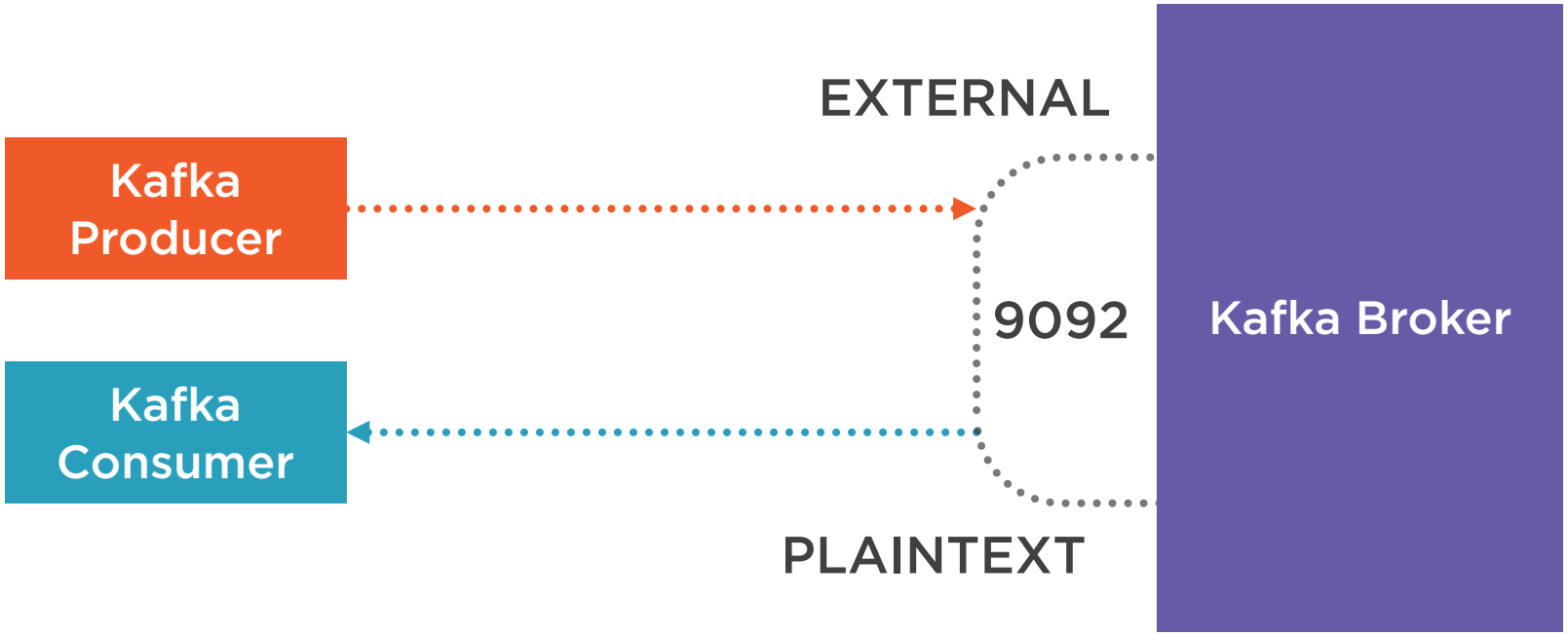
JAAS Configuration Files

PLAIN | SCRAM | OAUTHBEARER | GSSAPI

```
KafkaServer {
      com.sun.security.auth.module.Krb5LoginModule required
      ......
};
```

kafka_jaas.conf

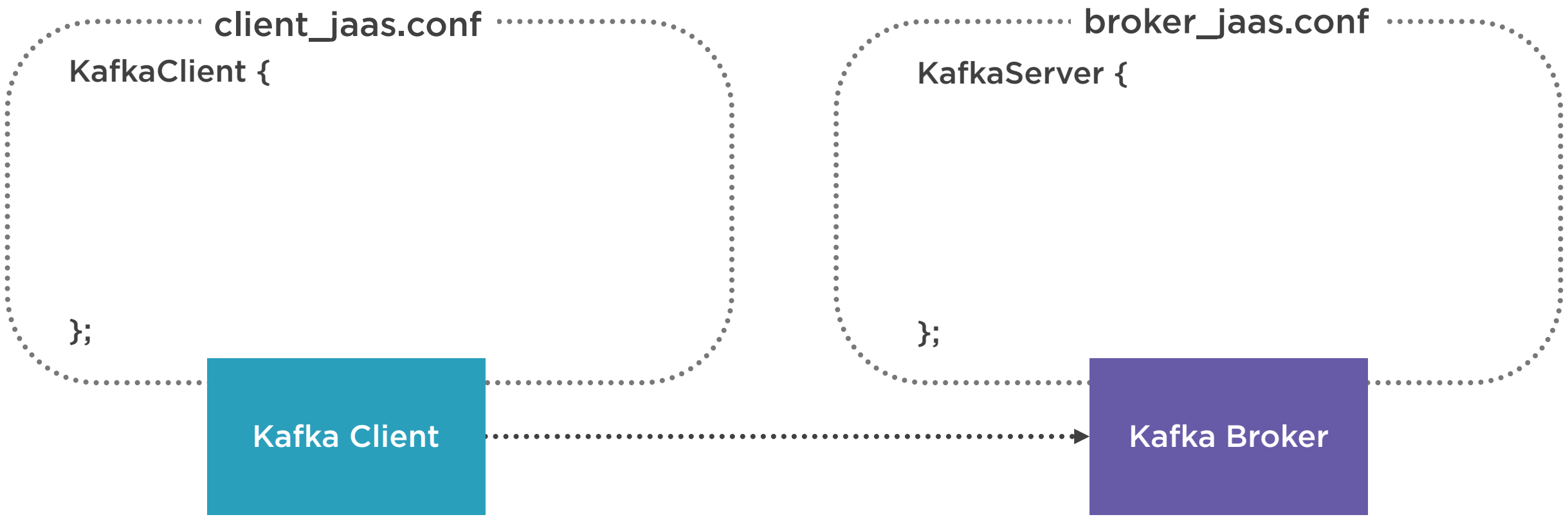
Kafka Listeners



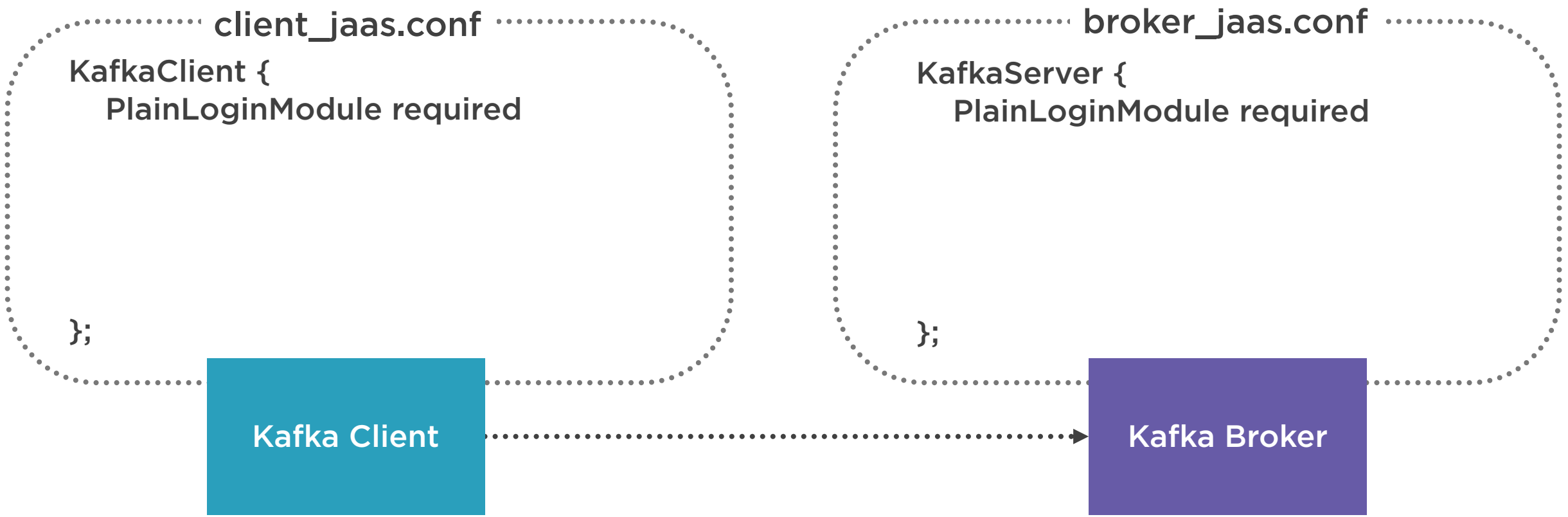
PLAINTEXT | SSL | SASL_PLAINTEXT | SASL_SSL

SASL PLAIN

SASL PLAIN


client_jaas.conf
KafkaClient {
};
broker_jaas.conf
KafkaServer {
};
Kafka Client
Kafka Broker

SASL PLAIN


client_jaas.conf
KafkaClient {
PlainLoginModule required
};
broker_jaas.conf
KafkaServer {
PlainLoginModule required
};
Kafka Client
Kafka Broker

SASL PLAIN

client_jaas.conf

```
KafkaClient {
    PlainLoginModule required



};
```

broker_jaas.conf

```
KafkaServer {
    PlainLoginModule required
    user_producer="producer-secret"



};
```

SASL PLAIN

client_jaas.conf

```
KafkaClient {
    PlainLoginModule required
    username=”producer"
    password=”producer-secret";


};
```

broker_jaas.conf

```
KafkaServer {
    PlainLoginModule required
    user_producer="producer-secret"


};
```

SASL PLAIN

client_jaas.conf

```
KafkaClient {
    PlainLoginModule required
    username=”producer"
    password=”producer-secret";
};
```

broker_jaas.conf

```
KafkaServer {
    PlainLoginModule required
    user_producer="producer-secret"
    user_consumer=”consumer"
    user_broker =”broker-secret”


};
```

SASL PLAIN

client_jaas.conf

```
KafkaClient {
    PlainLoginModule required
    username=”producer"
    password=”producer-secret";
};
```

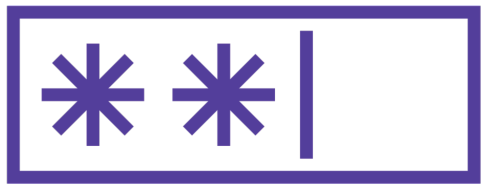
broker_jaas.conf

```
KafkaServer {
    PlainLoginModule required
    user_producer="producer-secret"
    user_consumer=”consumer"
    user_broker =”broker-secret”
    username=”broker"
    password="broker-secret"
};
```

SASL PLAIN Considerations

Transport Layer

Use SSL in production environments



Password Storage

Avoid storing clear text password on disk

External Authentication

Delegate authentication to external server

Demo

Set up SASL PLAIN

JAAS Configuration

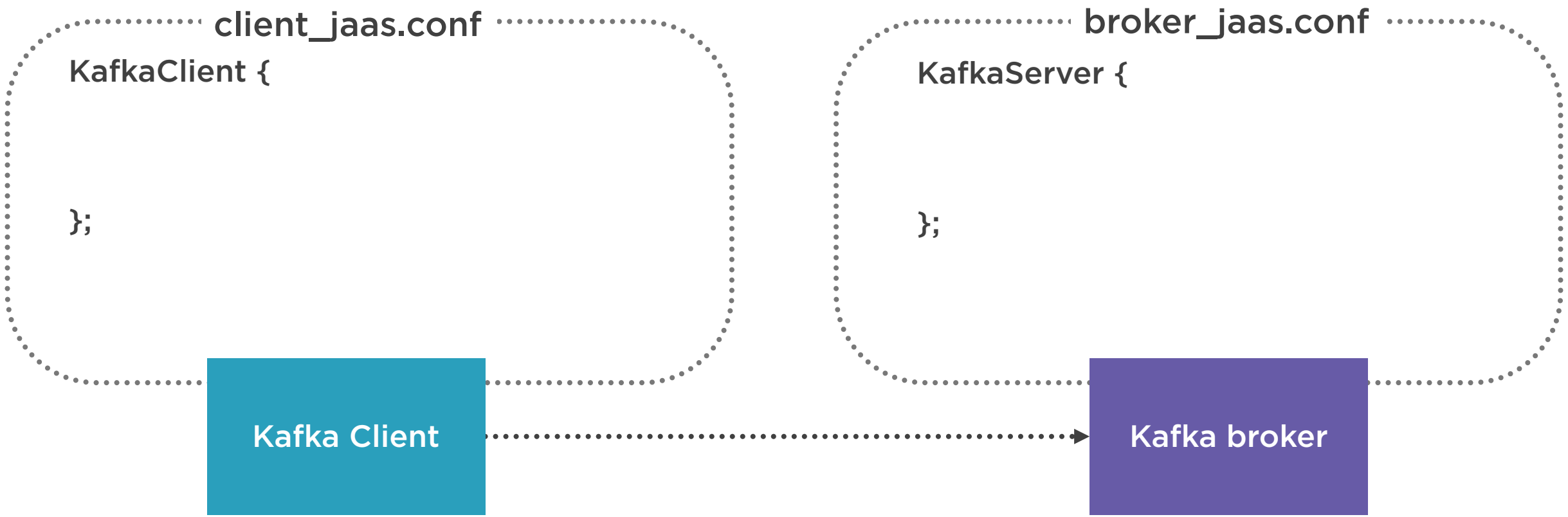
SASL SCRAM

SASL SCRAM

SHA-256

SHA-512

SASL SCRAM


client_jaas.conf
KafkaClient {
};
broker_jaas.conf
KafkaServer {
};
Kafka Client
Kafka broker

SASL SCRAM

client_jaas.conf

```
KafkaClient {
    ScramLoginModule required


};
```

broker_jaas.conf

```
KafkaServer {
    ScramLoginModule required


};
```

SASL SCRAM

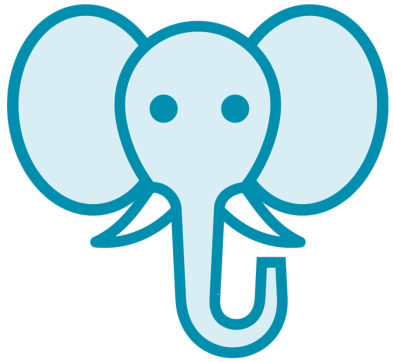
client_jaas.conf

```
KafkaClient {
    ScramLoginModule required
    username=”producer"
    password=”producer-secret";
};
```

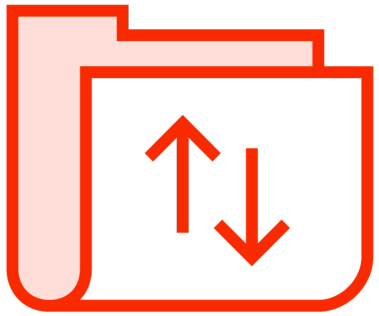
broker_jaas.conf

```
KafkaServer {
    ScramLoginModule required
    username=”broker"
    password="broker-secret"
};
```

SASL SCRAM Considerations

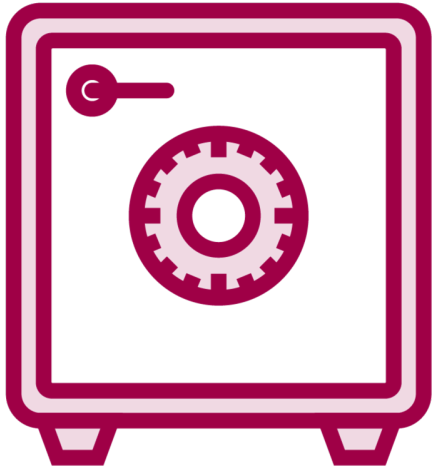


Secure Zookeeper



Transport Layer

Strong passwords



External credentials store

Demo

Set up SASL SCRAM

SASL OAUTHBEARER

SASL OAUTHBEARER

client_jaas.conf

```
KafkaClient {
  OAuthBearerLoginModule required

};
```

broker_jaas.conf

```
KafkaServer {
  OAuthBearerLoginModule required

};
```

SASL OAUTHBEARER

client_jaas.conf

```
KafkaClient {
  OAuthBearerLoginModule required
  unsecuredLoginStringClaim_sub=”admin";
};
```

broker_jaas.conf

```
KafkaServer {
  OAuthBearerLoginModule required
  unsecuredLoginStringClaim_sub=”broker";
};
```

SASL OAUTHBEARER

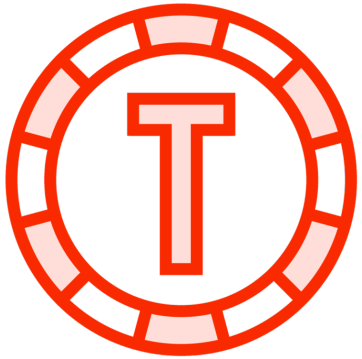
client_jaas.conf

```
KafkaClient {
  OAuthBearerLoginModule required
  unsecuredLoginPrincipalClaimName=alt
  unsecuredLoginStringClaim_alt=”admin”;
};
```

broker_jaas.conf

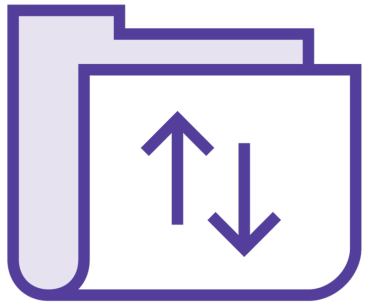
```
KafkaServer {
  OAuthBearerLoginModule required
  unsecuredLoginStringClaim_sub=”broker";
};
```

SASL OAUTHBEARER Considerations



Unsecured JWT

Empty JWT Signature



Transport Layer

Use SSL in production environments

Custom Implementation

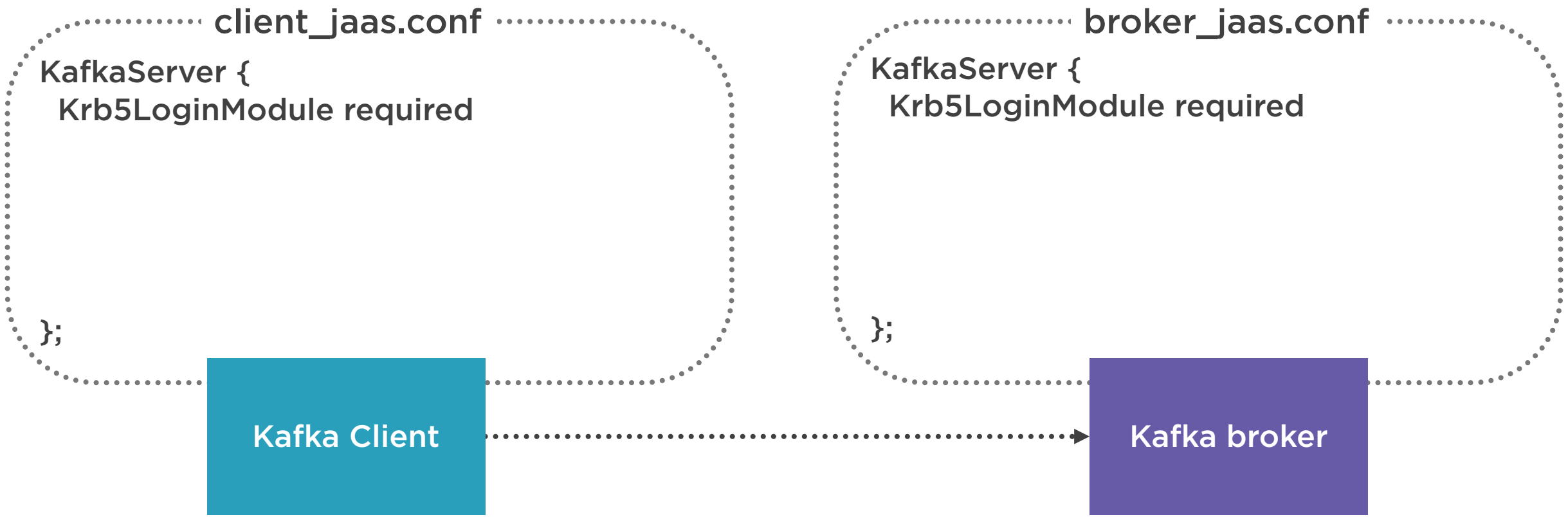
Provide secure implementation

Demo

Set up SASL OAUTHBEARER

SASL GSSAPI

SASL GSSAPI


client_jaas.conf
KafkaServer {
Krb5LoginModule required
};
broker_jaas.conf
KafkaServer {
Krb5LoginModule required
};
Kafka Client
Kafka broker

SASL GSSAPI

client_jaas.conf

```
KafkaServer {
  Krb5LoginModule required
  useKeyTab=true
  storeKey=true
  keyTab=”/path/to/consumer.keytab"
  principal=
      ”cons/cons-1@PLURALSIGHT.COM";
};
```

broker_jaas.conf

```
KafkaServer {
  Krb5LoginModule required
  useKeyTab=true
  storeKey=true
  keyTab=”/path/to/broker1.keytab"
  principal=
      "kafka/broker-1@PLURALSIGHT.COM";
};
```

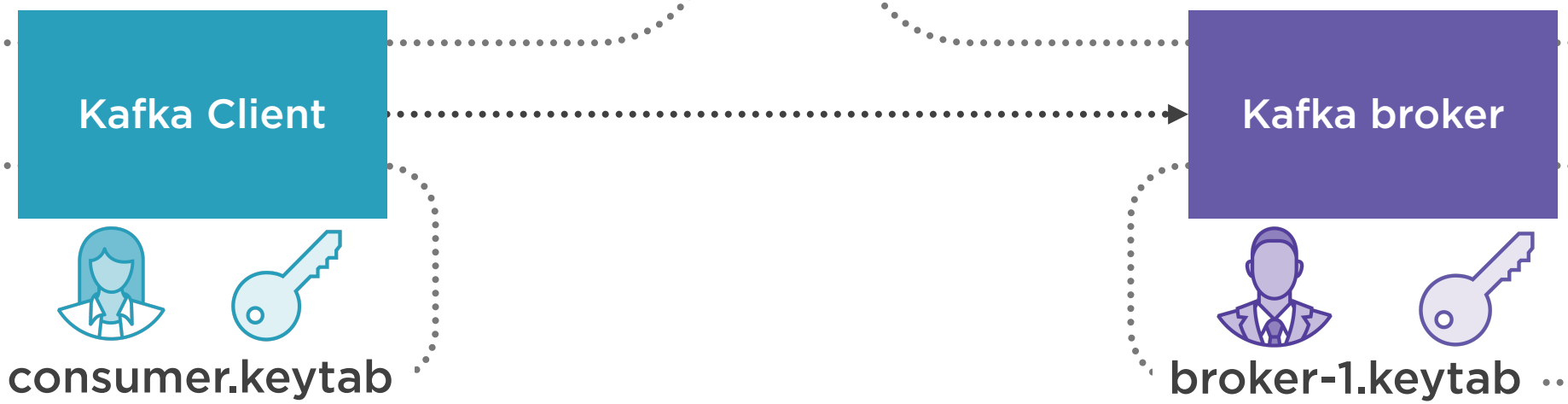
SASL GSSAPI

client_jaas.conf

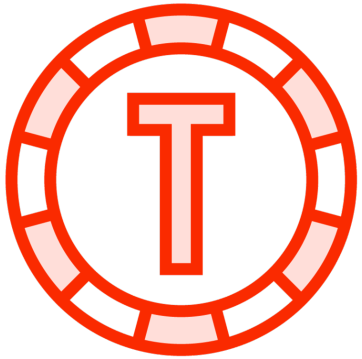
```
KafkaServer {
  Krb5LoginModule required
  useKeyTab=true
  storeKey=true
  keyTab=”/path/to/consumer.keytab"
  principal=
      ”cons/cons-1@PLURALSIGHT.COM";
};
```

broker_jaas.conf

```
KafkaServer {
  Krb5LoginModule required
  useKeyTab=true
  storeKey=true
  keyTab=”/path/to/broker1.keytab"
  principal=
      "kafka/broker-1@PLURALSIGHT.COM";
};
```

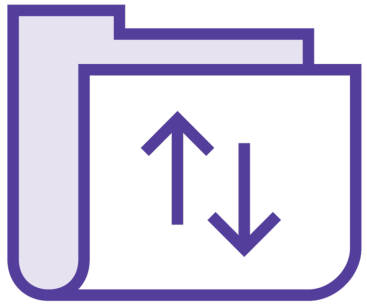


SASL GSSAPI Considerations



Pre-existing Kerberos

Check for Kerberos setups; e.g: Active Directory



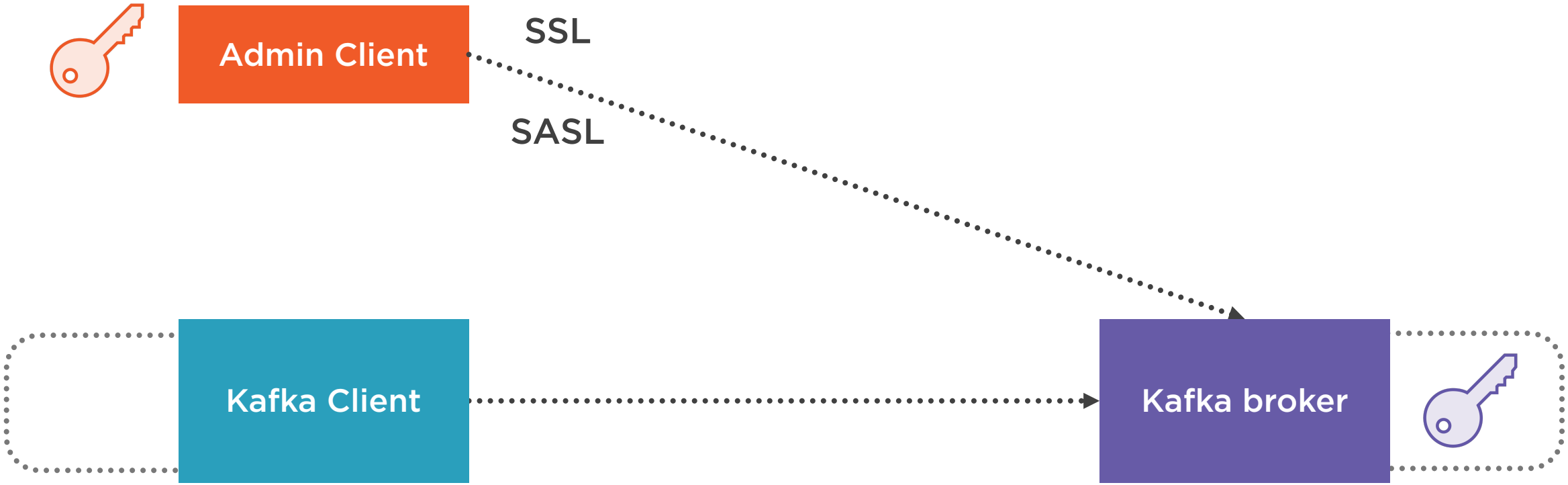
Transport Layer

Use SSL in production environments

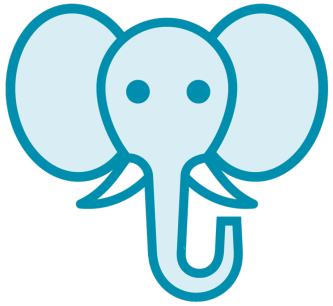
Hosts reachable using hostnames

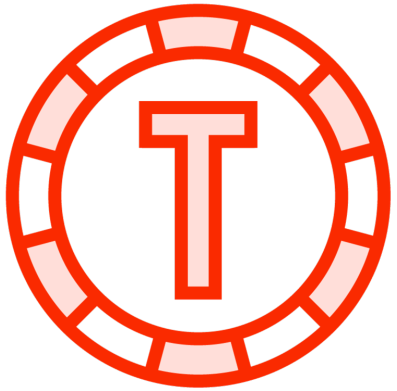
Kerberos relies heavily on hostnames

Delegation token



Delegation Token Considerations





Shared Secret

All brokers must be configured with the same secret key

Secured Zookeeper

Token details are stored in Zookeeper

Token Renewal

Default 24h renewal period for up to 7 days

Demo

Set up SASL GSSAPI