Single Sign On for MySQL, PostgreSQL and MongoDB with Kerberos

Percona University - Bangkok, Thailand - March 11

About Me

- Jaime Sicam
- Senior Database Performance Engineer at Percona
- APAC Team
  - Australia
  - India
  - Philippines
  - Vietnam
- Lives in Metro Manila, Philippines

Single Sign-On

- User authenticates one time
- Authentication is trusted by multiple services
- The users does not need to log in again
- Login once - Access many services

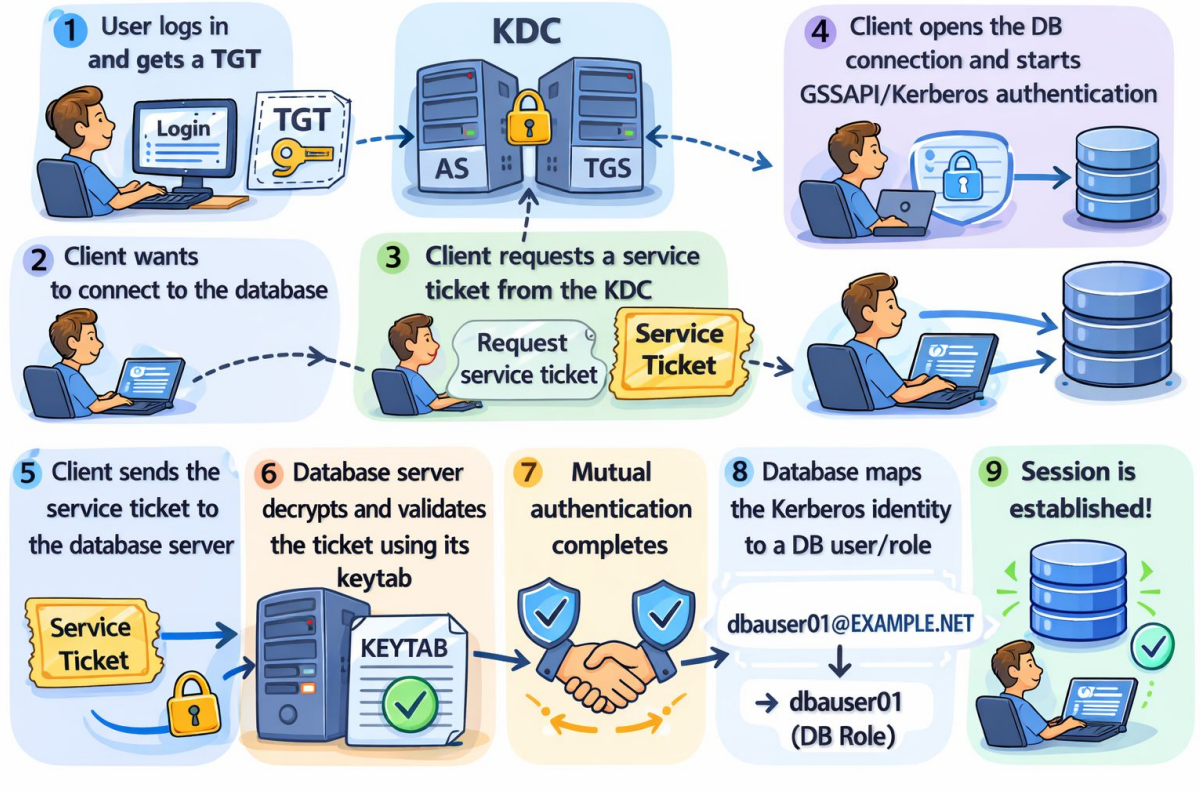
Kerberos

- Kerberos is a network authentication protocol
- Designed to provide secure authentication over insecure networks
- Uses tickets instead of sending passwords
- Used in enterprise environments
  - Active Directory
  - Linux systems
  - Database authentication

Kerberos components

- Client side
  - kinit
  - Kerberos credential cache
  - DB Driver (MySQL plugin, libpq, MongoDB driver)
- Key Distribution Center
  - Authentication Server(AS)
  - Ticket Granting Server(TGS)
- Database server
  - Service principal(SPN)
  - Keytab file
  - GSSAPI support

Authentication Sequence



Setting up a test Kerberos authentication system

- Windows AD
- Kerberos packages
- Samba(my favorite for labs)
  - DNS
  - LDAP protocol
  - Easy user/server management with **samba-tool**

Samba installation and configuration

Setup Docker network

```
docker network create example-network  --driver bridge  --subnet 172.100.0.0/24 --gateway 172.100.0.1
```

Create Samba container with ubuntu:22.04 image

```
docker run -it --detach --cpus=8 --memory=2G --ip=172.100.0.10 --privileged --hostname samba.example.net --domainname example.net --network
example-network --name samba ubuntu:22.04
```

Enter Samba container

```
docker exec -it samba bash
```

Install Samba and Kerberos utilities

```
apt update && DEBIAN_FRONTEND=noninteractive apt -y install samba net-tools winbind bind9-dnsutils vim iputils-ping ldap-utils krb5-user
```

```
REALM=EXAMPLE.NET
DOMAIN=EXAMPLE
ADMIN_PASSWORD=MyPassword2026
REQUIRE_TLS=no
OPTIONS=""
rm -f /etc/samba/smb.conf
if [ "$REQUIRE_TLS" = "yes" ]
then
 OPTIONS='ldap server require strong auth=Yes'
else
 OPTIONS='ldap server require strong auth=No'
fi
```

Samba installation and configuration

Provision Samba domain

```
samba-tool domain provision --realm $REALM --domain $DOMAIN --admin=$ADMIN_PASSWORD --option="$OPTIONS"
```

Create users with their passwords

```
samba-tool user add dbauser01 --surname=User01 --given-name=Dba --mail-address=dbauser01@example.net DbaPassword1
samba-tool user add dbauser02 --surname=User02 --given-name=Dba2 --mail-address=dbauser02@example.net DbaPassword2
samba-tool user add dbauser03 --surname=User03 --given-name=Dba3 --mail-address=dbauser03@example.net DbaPassword3
samba-tool user add devuser01 --surname=User01 --given-name=Dev --mail-address=devuser01@example.net DevPassword1
samba-tool user add devuser02 --surname=User02 --given-name=Dev2 --mail-address=devuser02@example.net DevPassword2
samba-tool user add devuser03 --surname=User03 --given-name=Dev3 --mail-address=devuser03@example.net DevPassword3
```

Create server principals for MongoDB and PostgreSQL. Know that you should use the hostnames as names for the server principals. Export each principal into their own keytabs that will be used by each server:

```
samba-tool user create mongodbsvc 'MyMongoDBPassword2026'
samba-tool spn add mongodb/mongodb.example.net mongodbsvc
samba-tool spn list mongodbsvc
samba-tool domain exportkeytab /root/mongodb.keytab --principal=mongodb/mongodb.example.net

samba-tool user create postgressvc 'MyPostgresPassword2026'
samba-tool spn add postgres/postgres.example.net postgressvc
samba-tool spn list postgressvc
samba-tool domain exportkeytab /root/postgres.keytab --principal=postgres/postgres.example.net
```

Samba installation and configuration

View contents of the keytabs

```
klist -k /root/mongodb.keytab
klist -k /root/postgres.keytab
```

Copy Kerberos config generated by samba to **/etc/krb5.conf**

```
cp -f /var/lib/samba/private/krb5.conf /etc/krb5.conf
```

Start Samba service

```
/etc/init.d/samba-ad-dc start
```

Check if Samba's DNS server resolution is working:

```
host example.net 127.0.0.1
host google.com 127.0.0.1
```

If it works, use it as the server's default DNS server. This DNS server will be used by database containers that will be created later

```
echo "nameserver 127.0.0.1" > /etc/resolv.conf
```

Samba installation and configuration

Test Kerberos authentication

```
kinit dbauser01
klist
kdestroy -A
```

Exit container

```
exit
```

Copy the config files from the samba container. They will be used by the database containers:

```
docker cp samba:/root/mongodb.keytab .
docker cp samba:/root/postgres.keytab .
docker cp samba:/etc/krb5.conf .
```

Links to Kerberos integration with databases

- https://dev.mysql.com/doc/refman/8.0/en/kerberos-pluggable-authentication.html
- https://www.postgresql.org/docs/current/gssapi-auth.html
- https://www.mongodb.com/docs/manual/tutorial/control-access-to-mongodb-with-kerberos-authentication/
- https://docs.percona.com/percona-server-for-mongodb/8.0/kerberos.html

Example Login

- Login to Kerberos
  kinit dbauser01@EXAMPLE.NET
- Login to Database servers

```
psql -h postgres.example.net -U dbauser01

mongosh --host mongodb.example.net --authenticationMechanism=GSSAPI
--authenticationDatabase='$external' --username  dbauser01@EXAMPLE.NET

mysql --host mysql.example.net --user=dbauser
--plugin_Authentication_kerberos_client_mode=GSSAPI
```

Databases that support Kerberos

- Percona Server for MongoDB
- PostgreSQL and Percona Server for PostgreSQL
- Percona Server for MySQL(via Percona PAM)
- MySQL Enterprise
- MongoDB Enterprise

PERCONA
for PostgreSQL

PERCONA
for MySQL

PERCONA
for MongoDB

Newer SSO solution(OIDC)

- Percona Server for MongoDB
- Percona Server for PostgreSQL(with pg_oidc_validator)
- MySQL Enterprise
- MongoDB Enterprise

PERCONA
for PostgreSQL

PERCONA
for MongoDB

Thank you!

Questions?