Enhancements in MySQL Server Security

PreFOSDEM MySQL Belgium Days 2025

Harin Vadodaria
Security Lead
Heatwave MySQL Engineering

January 31, 2025

Who Am I?

- Part of MySQL server engineering team since 2012
- Focus on development of security features for MySQL server and libmysql

MySQL Server Security Enhancement

Rationale and Overview of security features/enhancements introduced in last couple of years

- And what else do they bring?

Keyring Components

Authentication

TLS Enhancements

Account Management

Deprecation/Removal

Usability

Authentication Enhancements

Moving toward MFA support and more …

Multi Factored Authentication Support

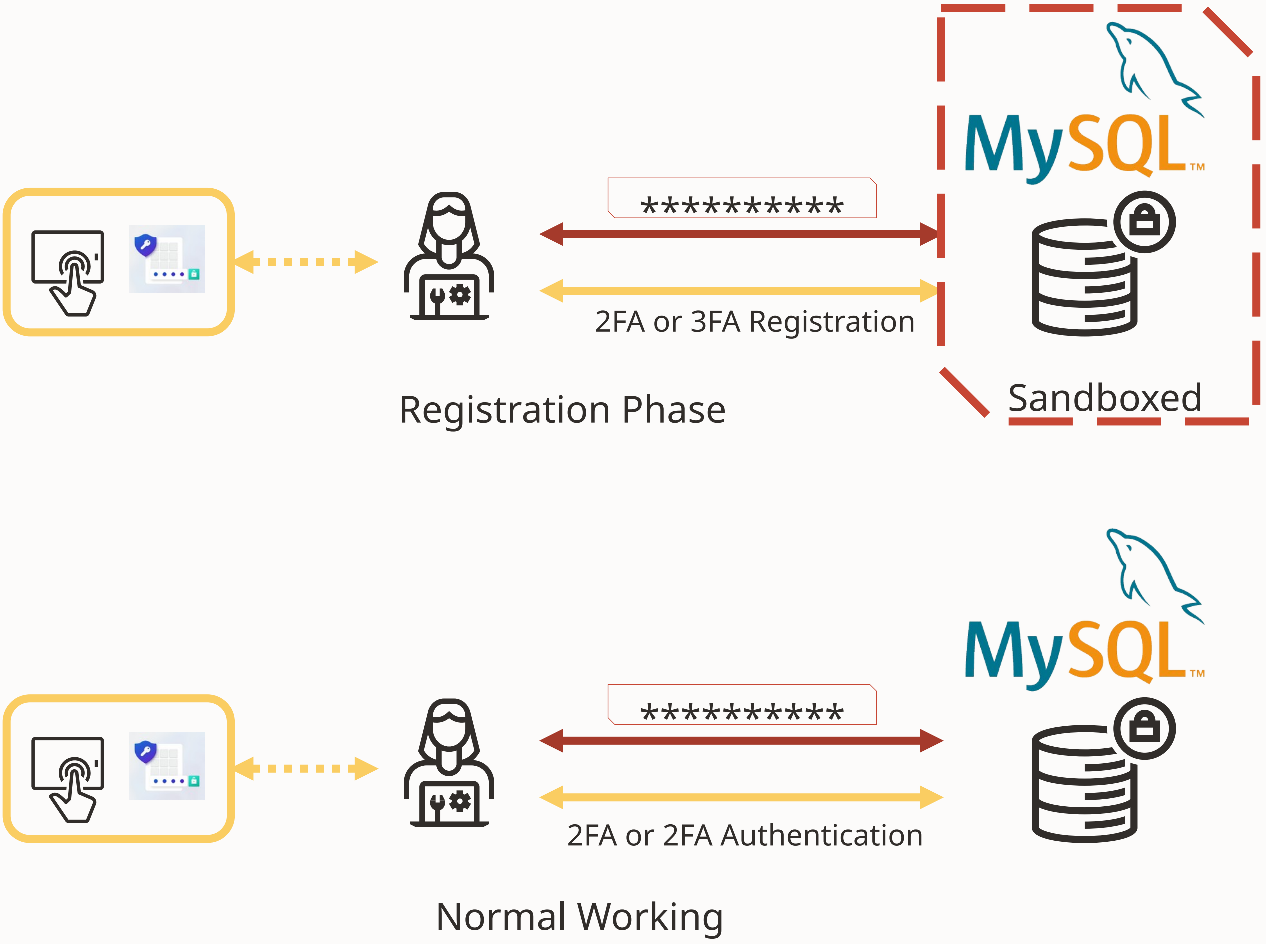
Protect administrative accounts better

- Password alone is not sufficient
- Authenticate using
  - Something you know (password)
  - Something you have (Yubikey) and/or Something you are (fingerprint)
- Utilizes libfido2. Supports:
  - Yubikeys
  - Windows Hello
- Cannot be used in non-interactive manner
- Two modes:
  - Registration mode - Sandboxed
  - Normal working

Multi Factored Authentication Support

- Requires
  - Configuring authentication policy
  - Configuring MFA for accounts
  - Using right tools at client side
- Typical order of authentication
  - Credentials
  - Fido2 OR Windows Hello
- Supports for upto 3 factors

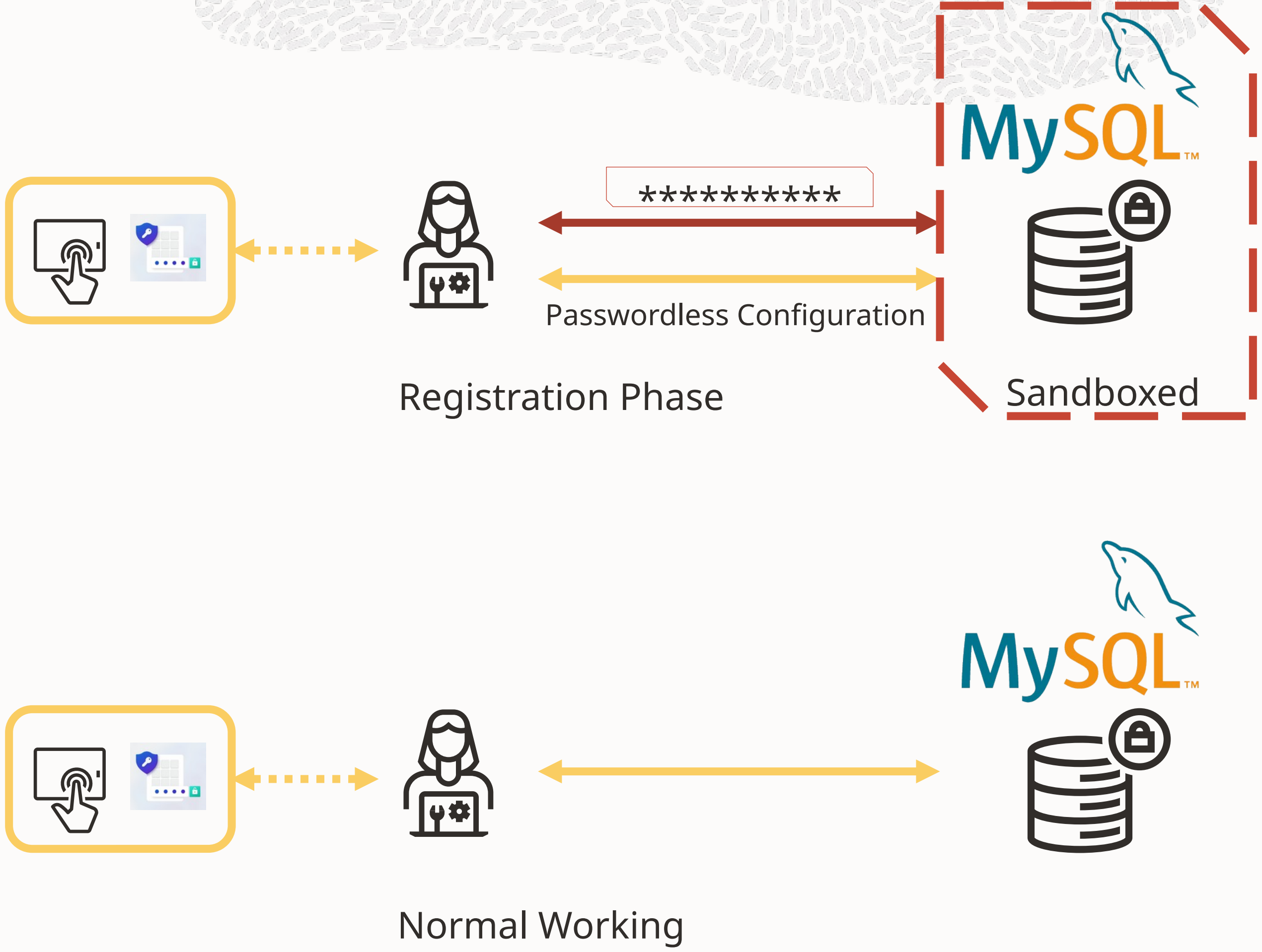
```
CREATE USER alice
IDENTIFIED WITH caching_sha2_password BY ‘<redacted>’
AND IDENTIFIED WITH authentication_webauthn;
```



Is That It? Passwordless Support

- Eliminate passwords altogether
- Use fido2 device or Windows Hello to login to MySQL server
- Uses MFA infrastructure for initial configuration
- As usual: Sandbox mode until configured properly

```
CREATE USER alice
IDENTIFIED WITH authentication_webauthn
INITIAL AUTHENTICATION IDENTIFIED BY
RANDOM PASSWORD;
```



****

Passwordless Configuration

Registration Phase

Sandboxed

Normal Working

Supporting Cloud Identity Providers

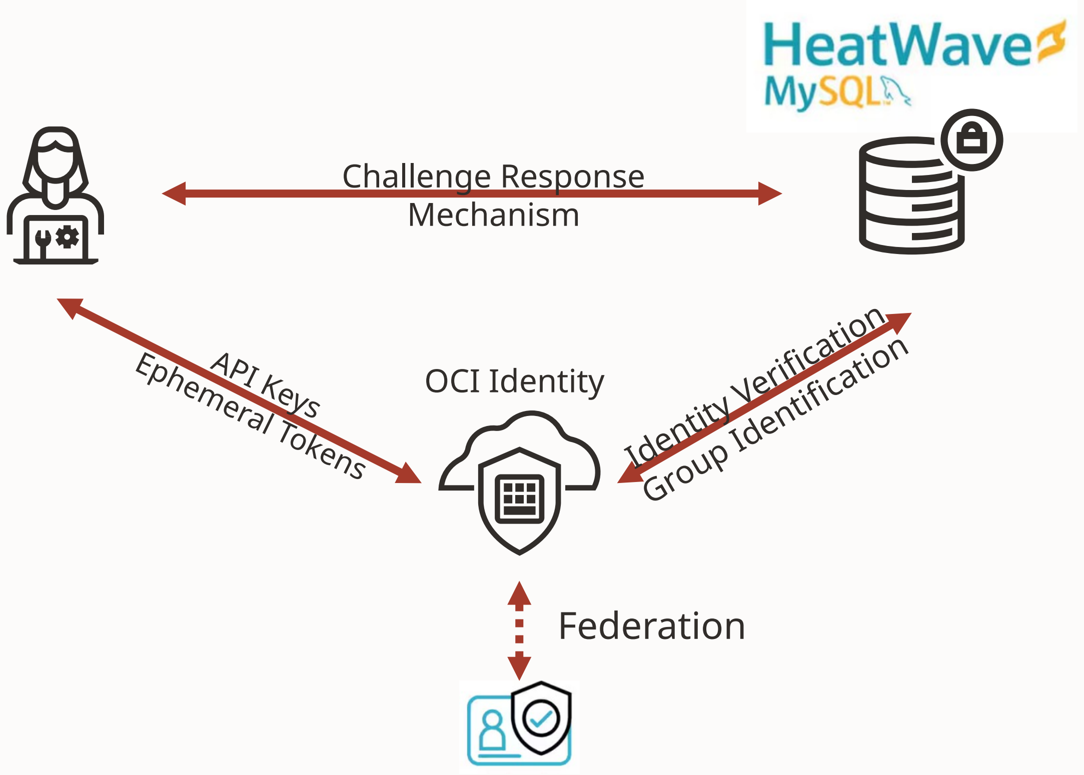
Shift: All cloud service providers have centralized identity management service

- MySQL integrates with
  - LDAP, Kerberos, PAM, Windows authentication
- Cloud identity providers: Supports federation to integrate with on-premise authentication server
- Need to integrate with cloud identity providers

Integration with OCI Identity

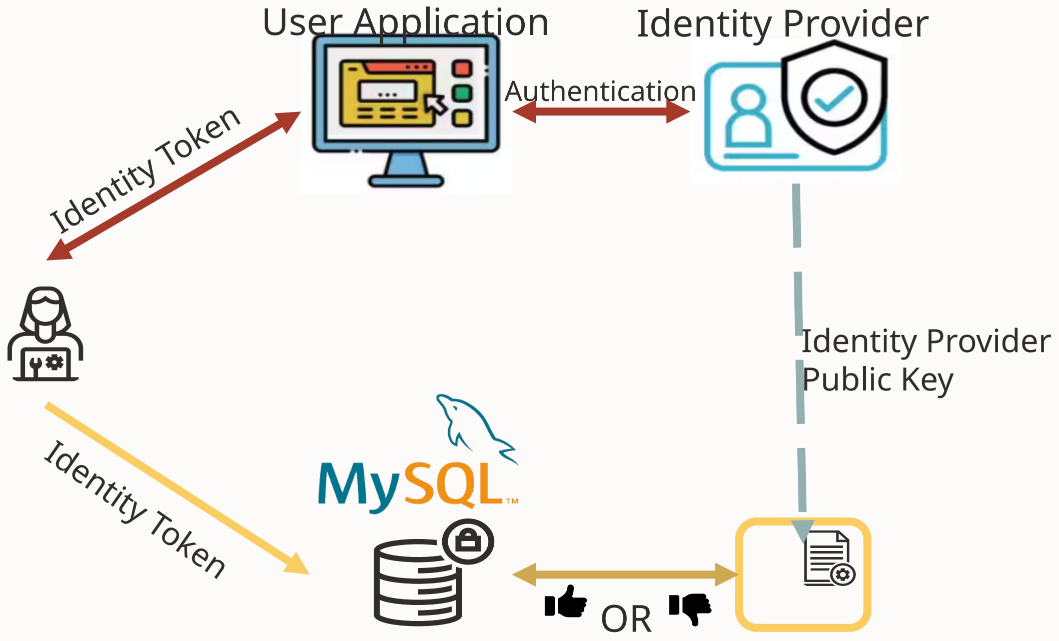
Available on Heatwave service on OCI

- Supports multiple authentication modes
  - API Keys
  - Ephemeral tokens
- Proxy support: Mapping to OCI groups
- Federation: Integrate with on-premise authentication server
  - OCI Identity domains



Is That It?OpenID Connect Support

- Allow on-premise instances to leverage cloud Identity providers
- OpenID Connect
- Supported by: All major cloud service providers
- Requires:
  - Server to recognize providers
  - Generate and supply token to client
  - A secure connection between server-client because Token => Credential
- Cannot support proxy: OpenID Connect does mandate group information in identity token



Account Security Improvements

Better controls for managing passwords …

Better Control Over Password Change

Knowing existing password is mandatory to change the password

- PASSWORD REQUIRE CURRENT : Prevent password change without existing password
- Configurable
  - Mandatory
  - Follow the system variable (password_require_current)
  - Make it optional
- Does not impact external authentication plugins

Is That It?Password Validation Enhancements

Prevent working around the password policy

- Mandate changed character percentage on password change
- Case insensitive and position agnostic
  - My$tr0ngpassword = mY$TR0ngPaSSWORD
  - Str0ngP@ssword = P@sswordStr0ng
- Requires REQUIRE CURRENT PASSWORD set for the account

MySQL NEVER stores password – only the hash transformation

Keyring Components

Changing the security model …

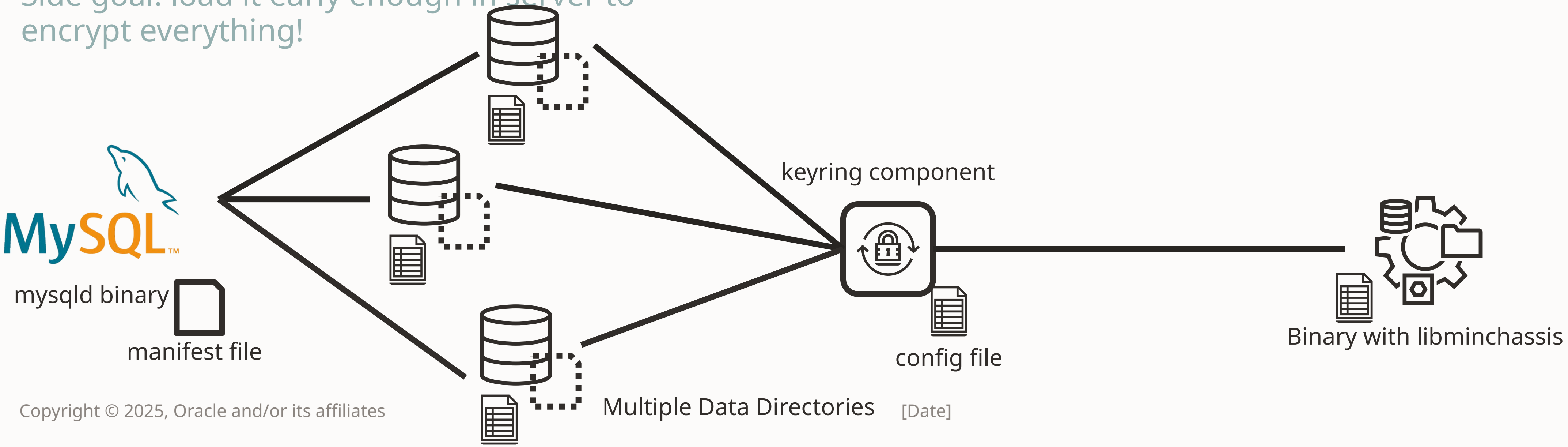
Keyring Components

Make it hard to change configuration (accidently)

- limit location to read configuration
  - Rely on file system security
- Manifest files
  - Global: alongside mysqld
  - Local: in data directory
- Side goal: load it early enough in server to encrypt everything!

Make it reusable for binaries other than server

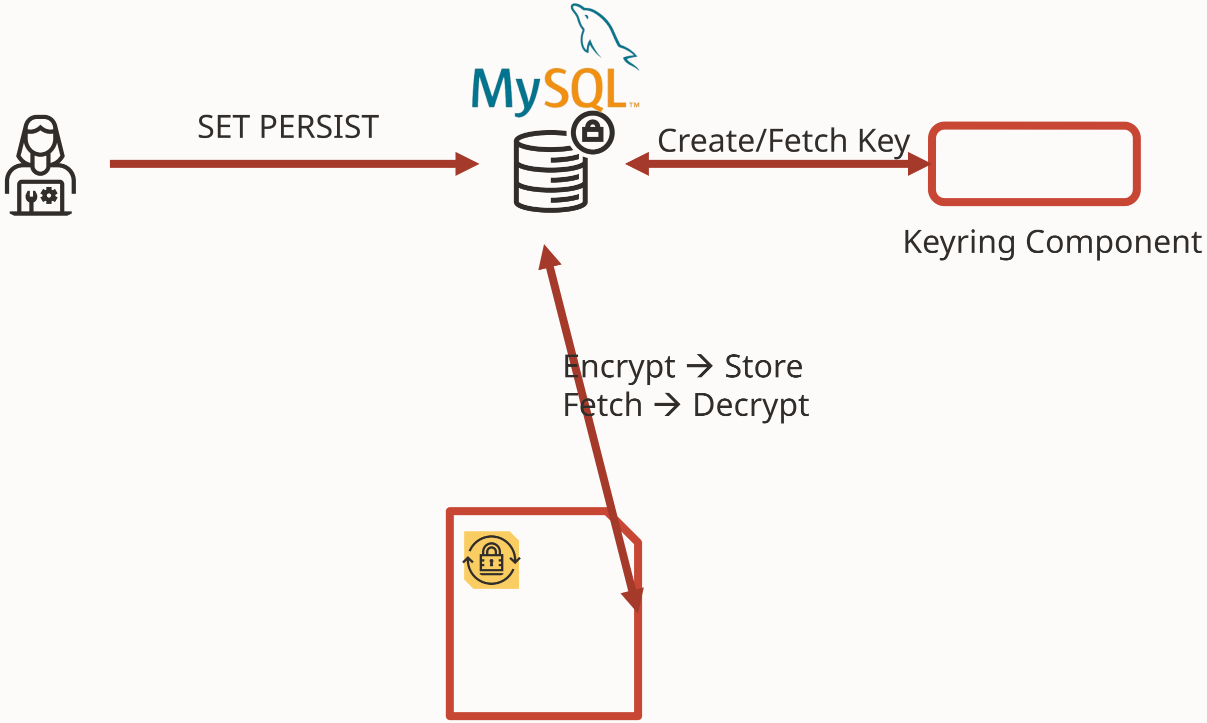
- Component configuration
  - Dedicated config file
  - Co-located with shared library
    - Can be configured to other path such as data directory



Is That It? Sensitive Variables Support

Secure storage for sensitive variables in server

- Infrastructure to store variables in encrypted form in mysqld-auto.cnf
- Supported through SET PERISTS | SET PERSIST_ONLY
- Requires keyring component
  - Plugin uses system variables: Chiken-and-egg
- Useful for variables related to e.g. passwords
- Restricted read access: Only privileged user can see value
- Supports
  - Server and component variables
  - Static or dynamic variables



Is That Really It?

What They Didn’t Tell You About Manifest file …

Can be used to load *ANY* component

- Server loads them early… very early
  - Even before persisted variables
- Caveat: You cannot depend on system variables
  - BYOConfig!
- Example: something that helps you orchestrate and/or monitor an instance
- Remember: A component need not provide *any* service

Communication Security Improvements

Stronger cipher supports, TLS Enhancements

Simplified And More Performant TLS Connection Establishment

Push toward OpenSSL 3.x+ APIs

- Ability to load global OpenSSL configuration
  - Simplified constraint enforcements – e.g. FIPS
  - Transparent support for custom providers
- Reduce context creation overhead: Cache and reuse
- Favor ECDSA over RSA
  - Equivalent security with lesser key size
  - Faster encryption/decryption speed
- Favor automatic DH parameter configuration over hardcoded one
- Revert to legacy APIs for SHA2/MD5 computation (See: https://bugs.mysql.com/bug.php?id=116393)
  - OpenSSL EVP APIs are slower than legacy APIs: https://github.com/openssl/openssl/issues/25858

Is That It? Hardening Ciphersuite Support

Continued evolution to support the strongest possible ciphers

- 8.4+ supports TLSv1.2/TLSv1.3 ciphers with following traits
  - Uses AEAD (Authenticated Encryption with Associated Data)
  - Strong hashing technique (SHA2)
  - DHE or ECDHE based key exchange
  - Use of ECDSA or RSA based keys

```
Value for option 'ssl_cipher' contains cipher 'EDH-RSA-DES-CBC3-SHA' that is blocked
```

- libmysqlclient: Supports legacy ciphers for interoperability PoV

Migration To Components

Services that are important for security features …

Plugin-To-Component Migration

Eat your own dogfood!

- Keyrings (all except keyring_okv - WIP)
  - Plugins are supported but deprecated
- Password validation
- Enterprise encryption functions
- Data masking
- Move audit event generation to component service APIs
  - Component service to plugin API: Through a bridge implementation in server component
  - To be done: Migrating audit plugins, firewall plugins
- Connection control

Is That It?Utilize Component Services

Opens Door For Component Development (Please use them!)

- Event Tracking Services can track
  - Authentication events (Create/Alter/Drop User, FLUSH PRIVILEGES, …)
  - Command and Query events (COM_* monitoring, Query execution monitoring)
  - Connection events (Connects/Disconnects/Change User)
  - Global Variable events (Change in configuration)
  - Server lifecycle events (Start/Stop)
  - Parse event (Enables query rewriting)
  - Execution state events (Success, Error, …)
- Password Validation
  - Create your own validation engine (More on this during FOSDEM!!)

Doxygen: https://dev.mysql.com/doc/dev/mysql-server/latest/group__group__components__services__inventory.html

General Improvements

Secure Choices, Customization, Deprecation …

Symmetric Encryption: Key Derivation Support

Key derivation function support in AES Encrypt/Decrypt

- Support for HKDF and PBKDF2_HMAC
- More robust than current method based on XOR
- Uses OpenSSL APIs
- Supports iterations as a configurable parameter
  - Strengthens key generation as per requirement
- Caution: Salt, Info, Iterations information must be retained for decryption

Enhanced Manageability

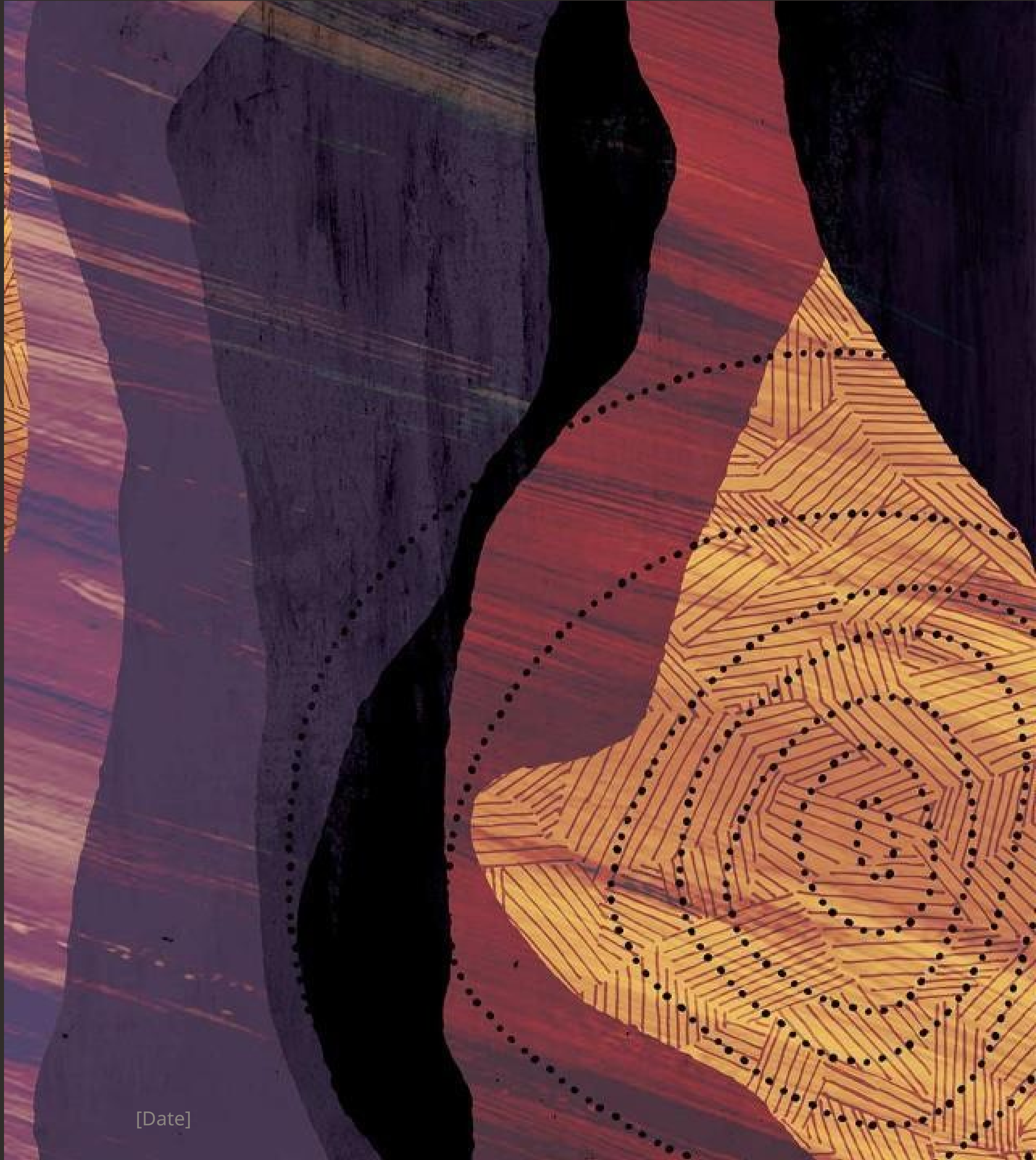
Custom schema support

- Configurable schema to hold policy metadata for
  - Audit
  - Firewall
  - Data masking
- Allows granular control
  - E.g. replication preferences/filter configuration
  - Allows user to setup more relaxed permissions (Required for data import) without compromising mysql schema

Deprecations/Removals

- Native password plugin
    - Deprecated in 8.0 → Disabled in 8.4 → Removed from 9.0
- Keyring plugins for which components are available
- FLUSH PRIVILEGES
- Legacy grant behavior
    - foo@hostname won’t inherit grants from foo@%
        - Obscure feature
        - Use SQL roles instead
    - Treating _ and % as wildcard in database grants: Convenient but can be easily misconfigured
        - Already unsupported if –partial_revokes is ON
- FIPS mode made READ-ONLY
    - OpenSSL 3.0+: Rely on systemwide configuration

Thank you for using MySQL™



ORACLE

Our mission is to help people see data in new ways, discover insights, unlock endless possibilities.