Troubleshooting

> Identifying and solving problem

Questions to ask:

- What are the symptoms?
- Where is the problem happening?
- When does the problem happen?
- Under which conditions does the problem happen?
- Is the problem reproducible?

Common problems are caused by:

- Poor performance:
    - Inadequate hardware (Scaling, horizontal or vertical)
    - Server or db configuration
    - Network connectivity
    - Query and application logic
- Improper configuration
    - Client configuration
        - Check there:
            - Incorrect login, pw or authentication type
            - Incorrect connection config
            - Incorrect driver version
    - Server configuration
        - Check these:
            - Memory
            - Disc space
            - Processing power
            - defragmentation
            - improve storage configuration
            - bugs in Os and RDBMS
    - Database configuration
        - Check these:
            - Database connection (no of connections allowed)
            - Insufficient buffering
            - Indexing
- poor connectivity
    - Server cannot be reached
    - DB instance cannot be reached
    - Client login credentials or security are incorrect
    - Client configuration incorrect
    - Check:
        - Verify server

    - Verify instance of db is running
    - Verify connection(use ping to server ip)
    - Verify client and config

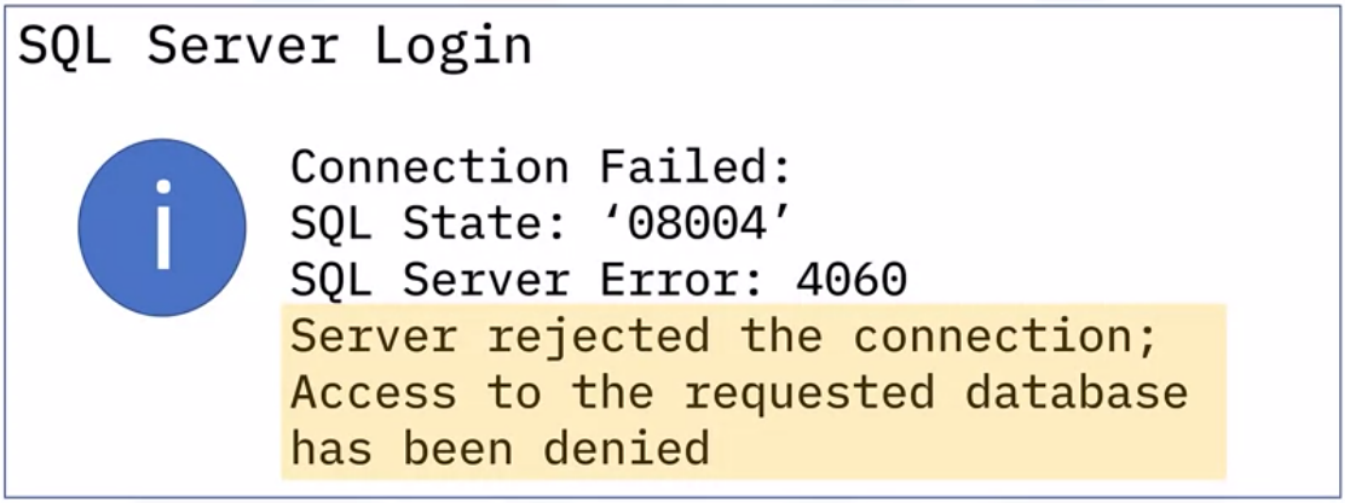
Tools:

- Monitoring tools
- Dashbords and reports
- Logs

Status variables, error codes and documentation

- From CMD:
    - `SERVICE MYSQL STATUS`
    - `SHOW STATUS` #global status or session status variable
    - `SHOW STATUS LIKE 'Key%';` #this can be used too
- GUI:
    - Activity monitor of SQL server
- Error logs:
    - Server and OS log
    - DB error logs:
        - error log
        - event log
        - trace log(optional)

Error codes:



- Search error code tables on documentation

Using logs for troubleshooting

> Diagnostic logs track events and errors on db when processing request, used for troubleshooting

- Types:
    - Server logs

    - Device logs
    - Network logs
    - Operating system logs
    - Database logs <== DBA should knooow this
    - Application logs

- Working with log fies:

    - location might be configurable
    - many logs in plain text format
    - some logs may require special tools to read and filter

- Component:

    - Type (error or event)
    - Error message
    - Where
    - Timestamp
    - User's Ip and User Agent
    - Additional details

```
2006-02-09-18.07.31.059000-300 I1H917            LEVEL: Event
PID     : 3140                 TID  : 2864       PROC : db2start.exe
INSTANCE: DB2                  NODE : 000
FUNCTION: Db2, RAS/PD component, _pdlogInt, probe:120
START   : New Diagnostic Log file
DATA #1 : Build Level, 124 bytes
Instance "DB2" uses "32" bits and Db2 code release "SQL09010"
with level identifier "01010107".
Informational tokens are "Db2 v9.1.0.190", "s060121", "", Fix Pack "0".
DATA #2 : System Info, 1564 bytes
System: WIN32_NT MYSRVR Service Pack 2 5.1 x86 Family 15, model 2, stepping 4
CPU: total:1 online:1 Cores per socket:1 Threading degree per core:1
Physical Memory(MB): total:1024 free:617 available:617
Virtual  Memory(MB): total:2462 free:2830
Swap     Memory(MB): total:1438 free:2213
Information in this record is only valid at the time when this file was created
(see this record's time stamp)
```

- In MYSQL
    - general query log
    - slow query log
    - error log

MySQL error log

Configure how much is stored in the log:

Level	Includes	Default
1	Errors	
2	Errors and warnings	
3	Errors, warnings and notes	Default

- Use `log_err_verbosity`:

```
[mysqld]
log_error_verbosity=2 # error and warning messages only
```

Automation

> Unattended processes and self-updating procedures, routine jobs, ..

- Version control is necessary
- Scripts can be used to automate
- Cron Jobs, shell scripts

Advantages

- throughput and productivity
- Improve quality or increase predictability of quality
- Improve consistency of process or product
- Increase consistency of outputs or results
- To free up staff
- provide higher level jobs in automated processes

Examples:

- Health check
- DB tasks
- Alert log files cleanup
- Trace file cleanup
- Data dictionary statistics (metadata)
- Data configuration check
- Schema object check
- Routine daily tasks (maybe using GUI)

Automating db testing

> Ensure information in db is correct and running properly within controlled testing environment

- Check schema, table, triggers
- Prevents data loss
- saves aborted transaction data
- prohibits unauthorized access
- Checks data integrity and consistency
- Cheaper , faster , secure

Automating reports and alerts

- Reports: Health of db, adress issues/problems, keep track of trends, predict future needs, regular schedule
- Notifications: bring event to attention, raise awareness but not critical issue
- Alerts: urgent issues brought to attention, threshold(warning and critical)
- RDBMS (most) have GUI or CUI or scrips option
- Varies based on RDBMS