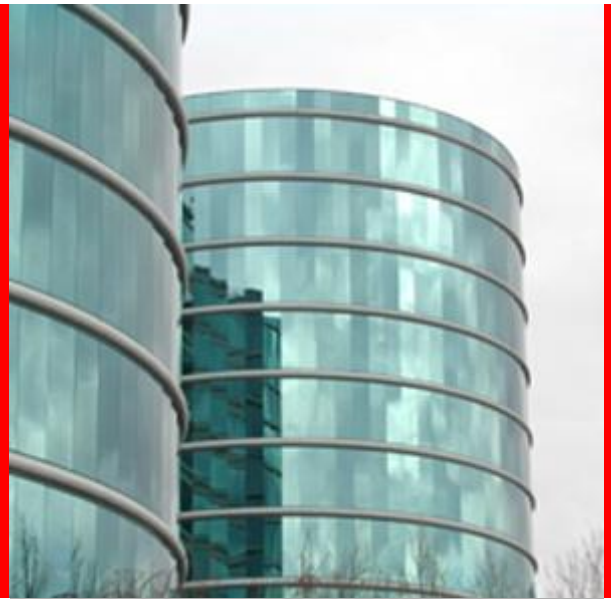


ORACLE®

MySQL Security In A Cloudy World

Dave Stokes @Stoker
David.Stokes @Oracle.com
MySQL Community Manager

slides: slideshare.net/davidmstokes

MySQL™

Safe Harbor Statement

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions.

The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

MySQL Securtiy In A Cloudy World

- How things *used* to be
  - Servers as pets
  - Relatively few users
  - Emphasis on costs
    - Disk Space
    - Disk Access
  - Quiet times for backup
  - Orderly data input
  - Structured data
  - Data base as heart of system

MySQL Securtiy In A Cloudy World

- How things ~~*used* to be~~ are
  - Servers as ~~pets~~ cattle
  - ~~Relatively few users~~ many users
  - Emphasis on ~~costs~~ speed, access, volume
    - Disk ~~Space~~ fast, spread out, increasing numbers
    - Disk ~~Access~~ ubiquity
  - Quiet times for backup GONE, replicate!
  - Orderly data input GONE, never toss any data!!!
  - Structured data GONE, shove in to JSON or API
  - Data base as ~~heart~~ of system

BOTTLENECK

Stokes's First Law

Nobody will run around complaining that the database is too fast!

Corolary

Slow is the new broken

Basic System Security for a Database

1. Database behind a firewall
2. Do not put other services on database server
3. Minimal number of login accounts on server
4. Keep OS up to date, keep MySQL up to date
5. Keep data replication between database servers on own network
6. Paranoia is your friend
7. Backups are need
    1. Database level
    2. Table level
    3. Row level
    4. Must be tested!!!!
8. Be Stingy With Accounts/Privileges

Problem 1 – The MySQL Login/Privs System

Clients talking to a MySQL Server are first checked for HOST permission

Then the User/Password combo is checked

Many instances have multiple logins for the same account on different networks
OR
A Convenient wildcard

This means that someone logging in from 10.10.10.99 can have permissions then when they login from home, another subnet, etc.

Recommendation: Check your mysql.user file periodically for duplicates.

Also check privs!

Problem 1 – The MySQL Login/Privs System -2

Username and password are masks into an in memory table, *mysql.user*.

Username expanded from 16 to 32 characters in 5.7,

Password column replaced by authentication_string

First match gets in!

May not be best match!

Rest of matching row contains privileges, limits, etc.

Other Login Tricks

Proxy Logins:

Users login with their own set of credentials that reference another account or specify defaults. Easy to add proxy users as needed.

```
CREATE USER "@" IDENTIFIED WITH ldap_auth AS 'O=Oracle, OU=MySQL';
```

Turn Off DNS lookups, go by IP Address. Don’t be taken down by a bad zone transfer!

Poor hide and seek with login credentials

You can “hide” login information in environmental variables and in configuration files. Your users may be sabotaging your database!

Check

- Login scripts
- ~/.my.cnf
- Source code
- Backup scripts

Use mysql_config_editor (5.6.6 and later)

Stores authentication credentials in an encrypted file

Shell> **mysql –login-path=production1 --host=prod**

No more clear text password to be sniffed!

Password locking, rotation, ageing

MySQL 5.7 will allow:

- Locking accounts
- You can specify complexity of passwords
- Set up password expiration policy
  - [mysqld]
    Default_password_lifetime=180
- Expired password -> Sandbox Mode

"Sorry, your password must contain a capital letter, two numbers, a symbol, an inspiring message, a spell, a gang sign, a hieroglyph and the blood of a virgin"

Authentication Plugins

- Native MySQL Password
- Old native (pre 4.1) removed 5.7.5
- SHA-256
- No-Login
  - Stored procedures, view with elevated privileges
- Clear-text (SSL/Private network)
- Socket peer-credential
- Test – Illustrates how to write your own authentication

SSL

As of 5.7.7, MySQL C Client wants to establish a SSL connection by default

- Connector/C
- Connector/C++
- Connector/ODBC

Problem 2 – Models

- Deployment Models
  - Virtual
  - DBaaS
  - Cloud hosted
  - Traditional
- Data Model
  - Relational
  - NoSQL
  - Hybrid

Problem 2 – Models

- Deployment Models
  - Virtual
  - DBaaS
  - Cloud hosted
  - Traditional
- Data Model
  - Relational
  - NoSQL
  - Hybrid

And you can easily have multiples of each!

Problem 2 – Models

- Deployment Models
  - Virtual
  - DBaaS
  - Cloud hosted
  - Traditional
- Data Model
  - Relational
  - NoSQL
  - Hybrid

And you can easily have multiples of each!

And how do you manage that?!

Problem 2 Continued- Getting data out there

How do you get data out to Virtual/DBaaS/Cloud?

- Automation tool
  - Docker, Ansible, Chef, etc.
- Loading the database itself is tricky
  - Copy live (mysqldbcopy, GTID replication) takes time/bandwith
  - Backup & Binary log for full/incremental
  - LVM snapshot & Binary log

Sprawl + Self-Service + Elastic = Headache ^N

- If servers with data can be spun up as needed to allow users to self fulfill data needs, how do you prevent data sprawl (digital landfill) ?
- How do you update N sets of disparate data to keep data current?
- Can you control sensitive to secret data as data sprawl grows? Do you have plans for a breach?
- What is you exposure / liability?

Slide to check if the Audience is Still Awake

Problem 3 – Data protection

- Data In
  - GIGO
    - Mainly application issue
    - Bulk loading
- Data Out
  - 5 W's
  - SQL Hacks
  - Company/Legal Policies
  - Planned Purge
- Data Access
  - How do you protect what is supposed to be protected?

MySQL Database

Performance, Reliability, Ease of Use

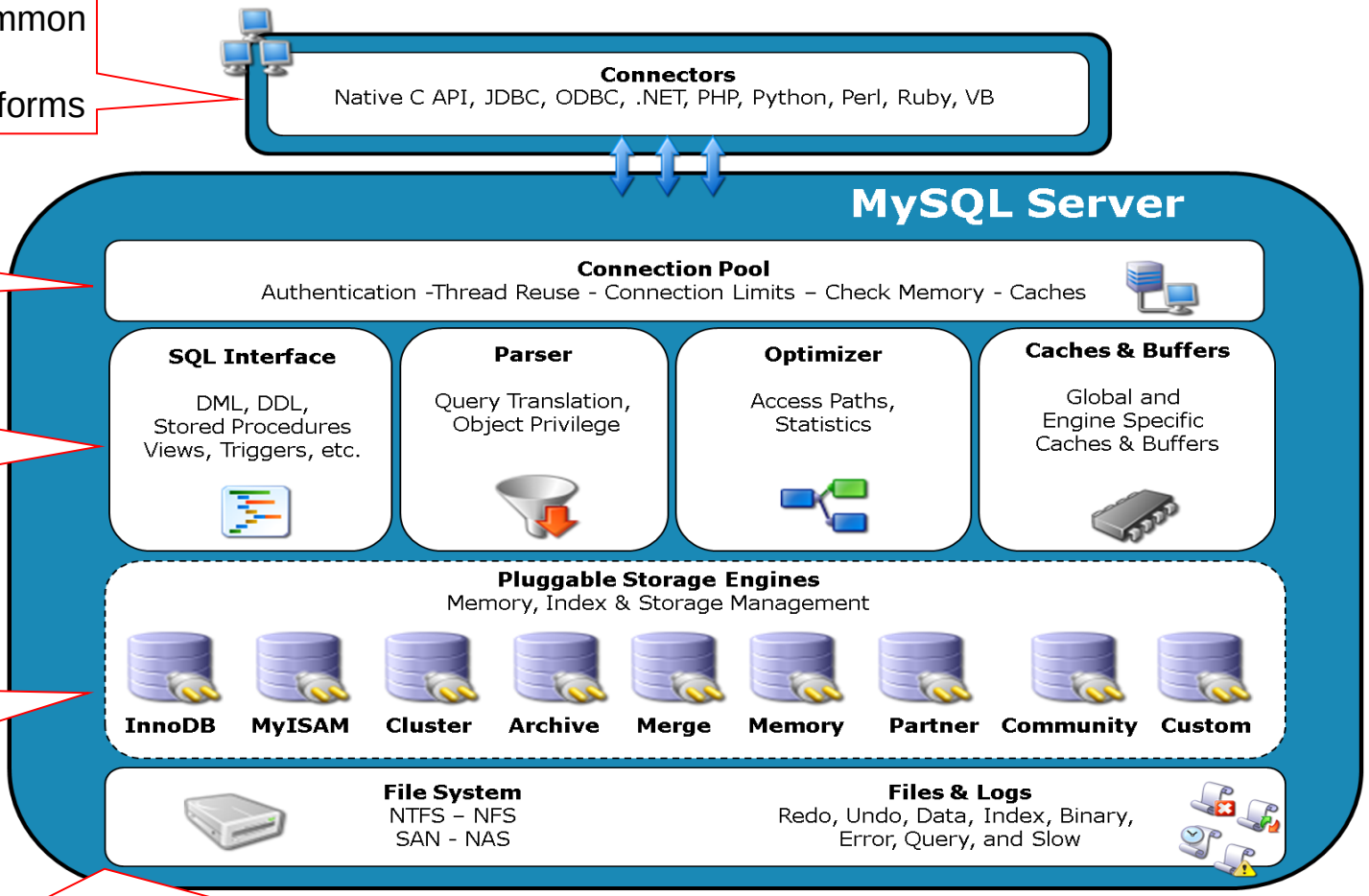
Support for common development languages/platforms

Efficient multi-threaded session handling

Full DML, DDL parsing, cost based optimizer, caching of queries and result sets

Flexible Storage Engine options for application specific storage needs

Flexible logging and physical storage options



Cloudy Virtual Servers?!?

So you can instantiate a new server on a whim but can you trust it?

Creating New MySQL 5.7 Instance

MySQL is ‘Secure by Default’

- No Anonymous Accounts
  - User =‘’ and password=‘’
- No test database
- Forced root password
  - Randomly generated
  - Or force insecure install

- Run mysql_secure_install (now a binary)

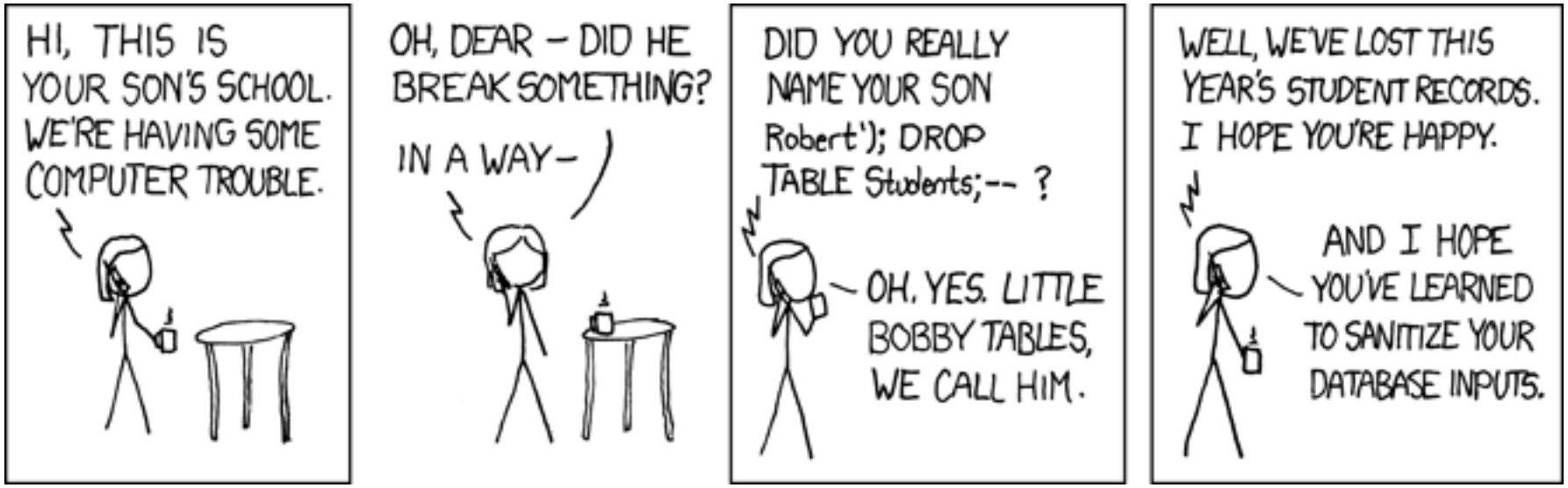
Problem 4 – Surface Area

The more the data is spread out, the more risk of unauthorized access increases.

The more Admins/DBAs/etc have access to your data, the higher risk of somebody doing something you do not want with that data.

The more networks the data travels the bigger potential exposure.

Sharing with other users on the equipment is asking for a felony in Murphy's law


HI, THIS IS YOUR SON'S SCHOOL. WE'RE HAVING SOME COMPUTER TROUBLE.
OH, DEAR – DID HE BREAK SOMETHING?
IN A WAY–
DID YOU REALLY NAME YOUR SON Robert'); DROP TABLE Students;-- ?
OH. YES. LITTLE BOBBY TABLES, WE CALL HIM.
WELL, WE'VE LOST THIS YEAR'S STUDENT RECORDS. I HOPE YOU'RE HAPPY.
AND I HOPE YOU'VE LEARNED TO SANITIZE YOUR DATABASE INPUTS.

Contracts not written in stone

If you do put your database in the cloud, know the terms of service *inside and out*!

Know how problems are communicated to and from cloud vendor.

Cover what happens if vendor is sold, goes bankrupt, changes country, or just is lousy.

And make sure your management knows of the above!

Problem 5 – DBA Still Needed?

Even if you outsource every database to the cloud, there has to be some management of the data.

- Quality control
- Are things working properly
- Schema changes
- Query monitoring and optimization
- Architecture
- DBA tasks – How much do you and how much for vendor? Who takes care of what can be in cloud??

MySQL™

ORACLE®

Thanks for attending!

Dave Stokes David.Stokes@Oracle.com
@stoker slideshare.net/davidmstokes