



Optimize MySQL for Production Workloads

You won't BELIEVE these 10 MySQL Hacks!

Brian Welcker, Principal Product Manager Amazon RDS

May 23, 2018



Why MySQL?



Ease of use

Triggers, stored procedures, updatable views, JSON

Many available tools and utilities

Performance

B-tree disk tables with index compression, thread-based memory allocation, and optimized nested-loop joins

Reliability

Transactional storage engine adheres to ACID model

Broad language support

Python, Java, .NET, Node.js, C/C++, Perl, PHP, Tcl, Ruby, ODBC, Erlang, Objective-C, more

Security

Authentication and granular access control

Full encryption of database traffic with TLS

Running MySQL on AWS

Self-managed on EC2



AWS managed DB services

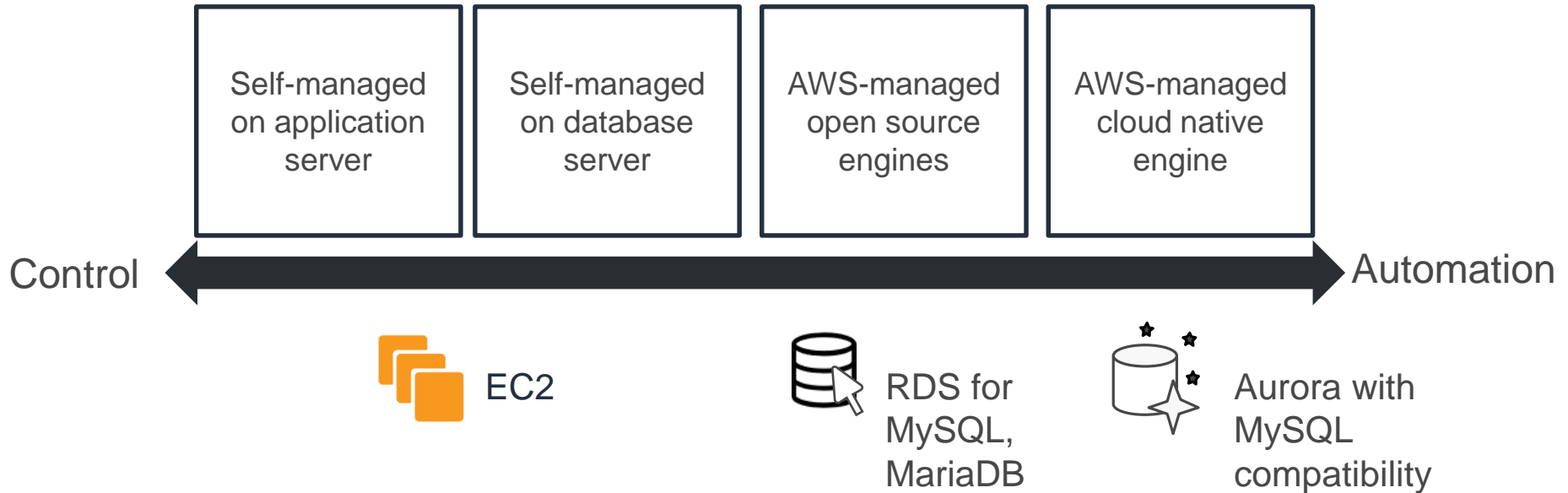


RDS for MySQL, MariaDB



Aurora with MySQL-compatibility

Choosing your AWS MySQL deployment strategy



Managed MySQL engines in AWS


Standard
The open source standard MySQL

 MariaDB
Community
The popular community choice

Amazon Aurora
Performance
The fastest MySQL compatible engine

Amazon RDS for MySQL

Support for MySQL Community Edition versions 5.5, 5.6, and 5.7
InnoDB and MyISAM storage engines

Version 5.7—new features

- JSON support
- Query optimizer improvements
- GIS extensions
- Improved parallel replication
- Dynamic buffer pool resizing

Version 8.0 coming soon



Amazon RDS for MariaDB

Support for versions 10.0,10.1, and 10.2

Same instance, regions, pricing as RDS for MySQL (including free tier)

Differences from RDS MySQL

- Thread Pooling
- GTID for Replication

Version 10.2 – new features

- Window Functions
- Spatial Indexes
- JSON Functions
- Common Table Expressions



Amazon Aurora



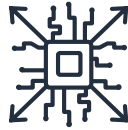
- Built from the ground up to leverage AWS
- MySQL 5.6 and 5.7 compatible with up to 5x better performance on the same hardware: 100,000 writes/sec & 500,000 reads/sec
- Scalable with up to 64 TB in single database, up to 15 read replicas
- Highly available, durable, and fault-tolerant custom SSD storage layer: 6-way replicated across 3 Availability Zones
- Transparent encryption for data at rest using AWS KMS
- Stored procedures in Amazon Aurora can invoke AWS Lambda functions

Optimizing MySQL in Amazon RDS

Here come the hacks!

Hack #1: Scale your compute and storage

Choose the right configuration for your workload



Scale compute up to handle increased load or down to save costs

- As little as 1vCPU / 1 GiB of RAM
- Up to 64 vCPU and 488 GiB of RAM
- Minimal disruption for compute scaling



Scale storage for larger data sets

- Scalable Amazon EBS storage up to 16 TiB
- General Purpose SSD or Provisioned IOPS SSD storage
- No downtime for storage scaling
- Amazon Aurora automatically scales to 64 TiB

Hack #2: Turn on automated backups

Point-in-time recovery for your DB instance

- Scheduled daily volume backup of entire instance
- Archive database change logs
- 35-day maximum retention
- Negligible impact on database performance
- Taken from standby when running Multi-AZ

DB instance status

available

Multi AZ

Yes

Secondary zone

us-east-1d

Automated backups

Enabled (7 Days)

Latest restore time

March 22, 2018 at 10:25:00 AM
UTC-7



Every day during your backup window, RDS creates a storage volume snapshot of your instance

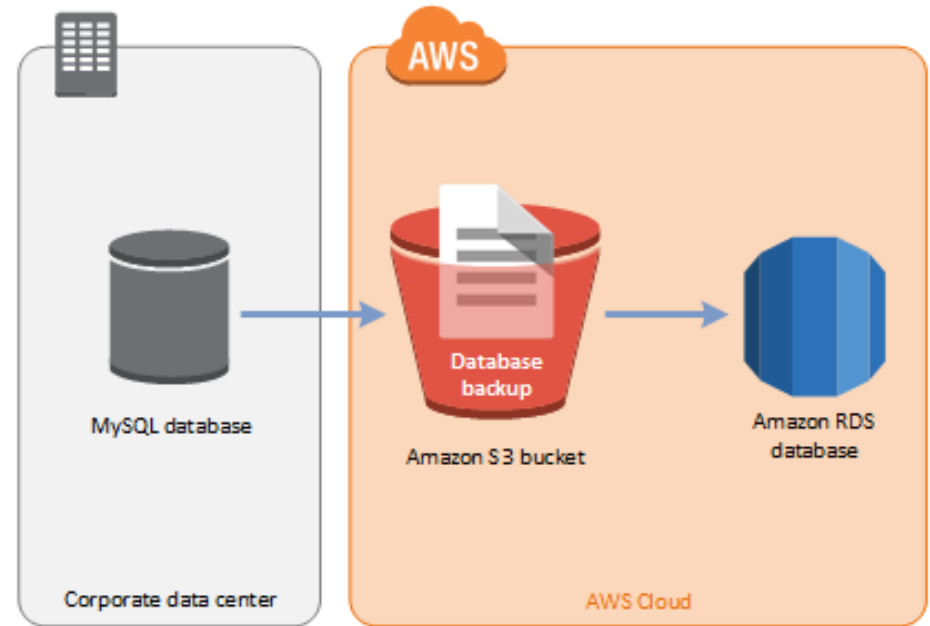


Every five minutes, RDS backs up the transaction logs of your database

Hack #3: Import Backups from Amazon S3

Easiest way to get your database into RDS

- Create full or incremental backup with Percona XtraBackup
- Upload backup to S3 bucket
- Provide AWS IAM role to access S3 bucket
- Restore backup to RDS MySQL or Amazon Aurora
- Use replication to catch up to changes in the source database (if necessary)



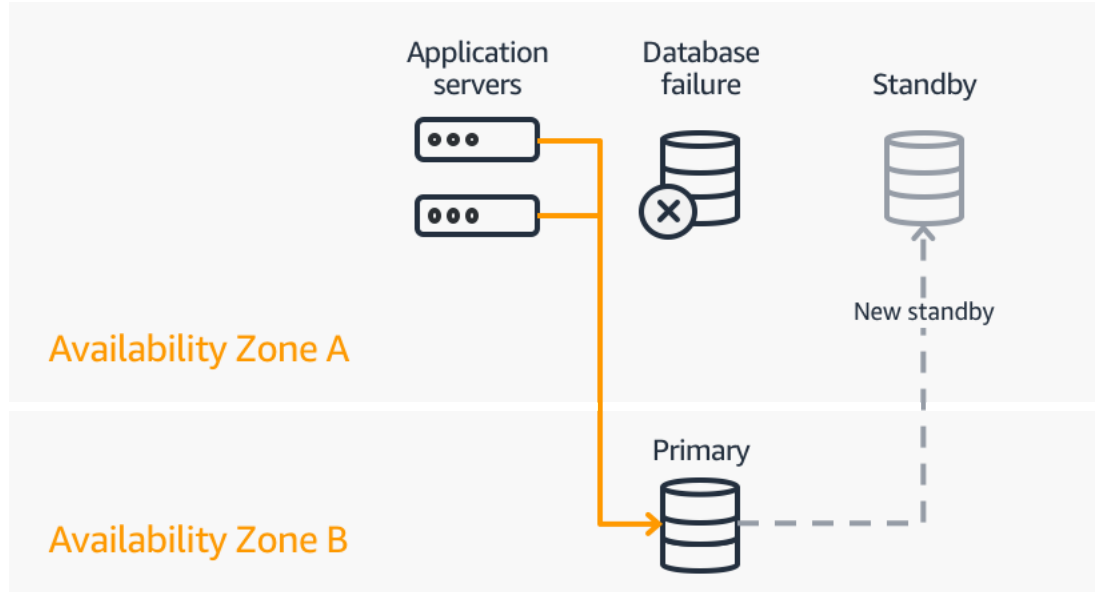
Hack #4: Enable Multi-AZ configuration

Enterprise-grade high availability

Fault tolerance across multiple data centers

- Automatic failover
- Synchronous replication
- Enabled with a few clicks

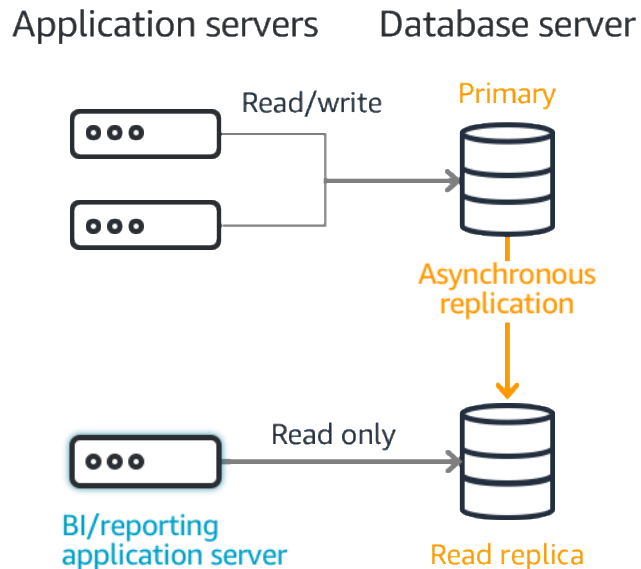
Redirection to the new primary instance is provided through DNS



Hack #5: Create read replicas

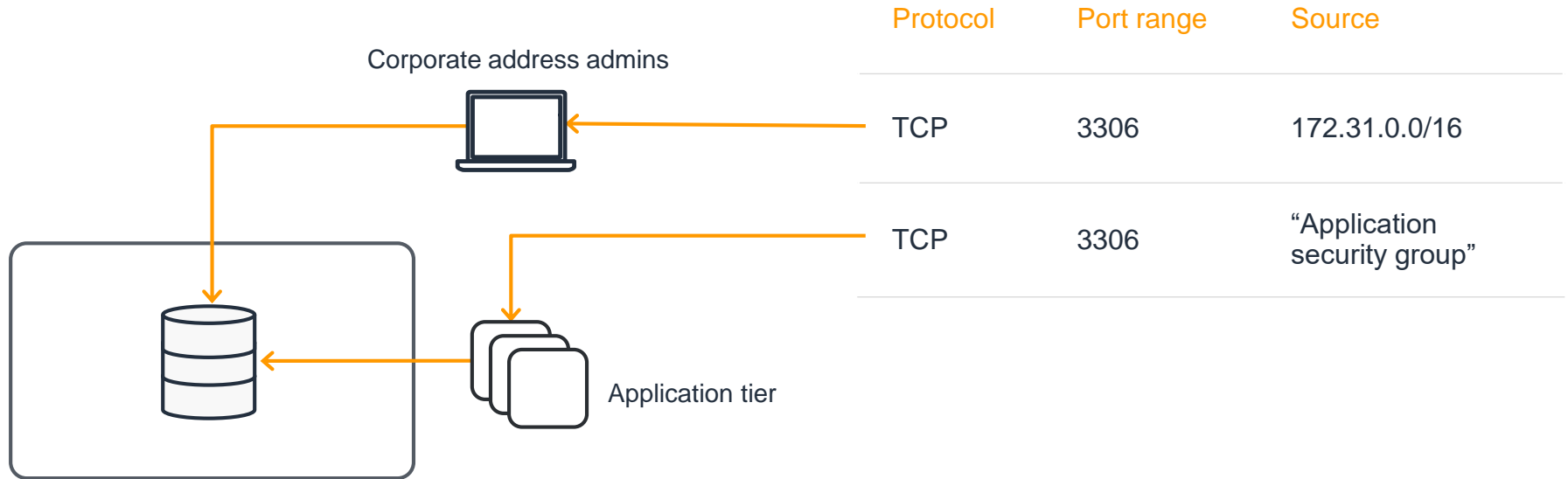
Read scaling and disaster recovery

- Relieve pressure on your master node with additional read capacity
- Bring data close to your applications in different regions
- Promote a read replica to a master for faster recovery in the event of disaster
- Low latency replicas for Amazon Aurora can be used for both read scaling and failover targets



Hack #6: Secure network access with Amazon VPC

Security groups provide a network firewall for your database server



Hack #7: Turn on encryption

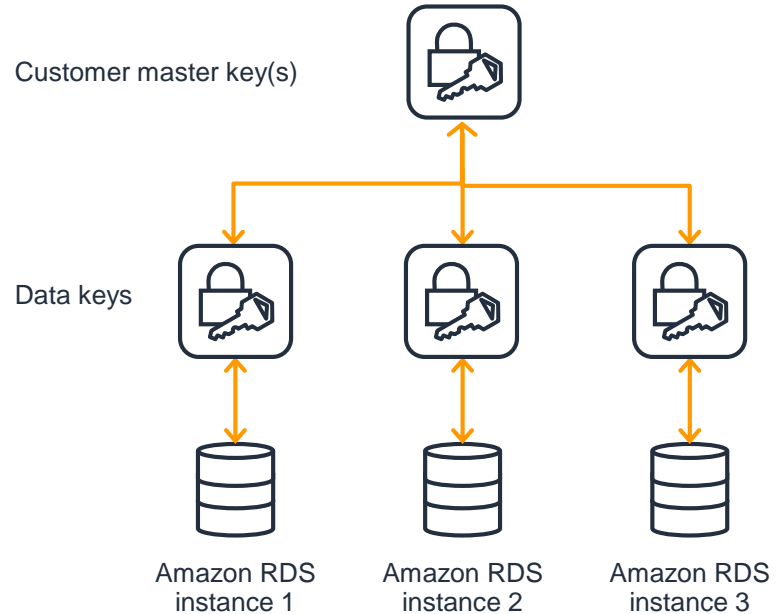
Encrypt data at rest with AWS Key Management Service

Two-tiered key hierarchy using envelope encryption

- Unique data key encrypts customer data
- AWS KMS master keys encrypt data keys
- No impact on performance

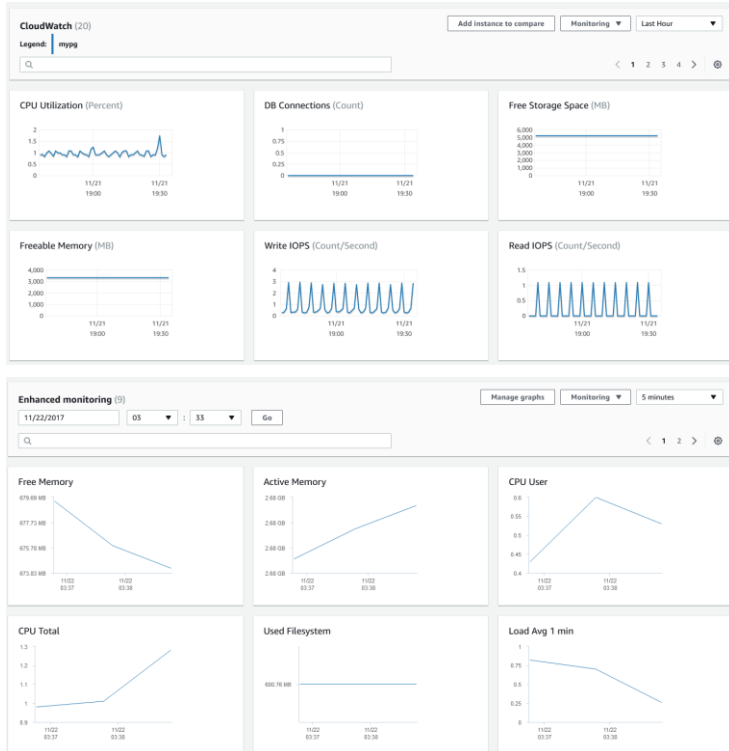
Benefits

- Limits risk of compromised data key
- Better performance for encrypting large data
- Easier to manage small number of master keys than millions of data keys
- Centralized access and audit of key activity



Hack #8: Monitor your MySQL DB instance

Detect problems before a failure occurs



Amazon CloudWatch Metrics

- Displayed in the RDS Console or personalized CloudWatch dashboards
- As low as 1 minute intervals

Amazon CloudWatch Alarms

- Trigger actions based on a metric value relative to a threshold you set

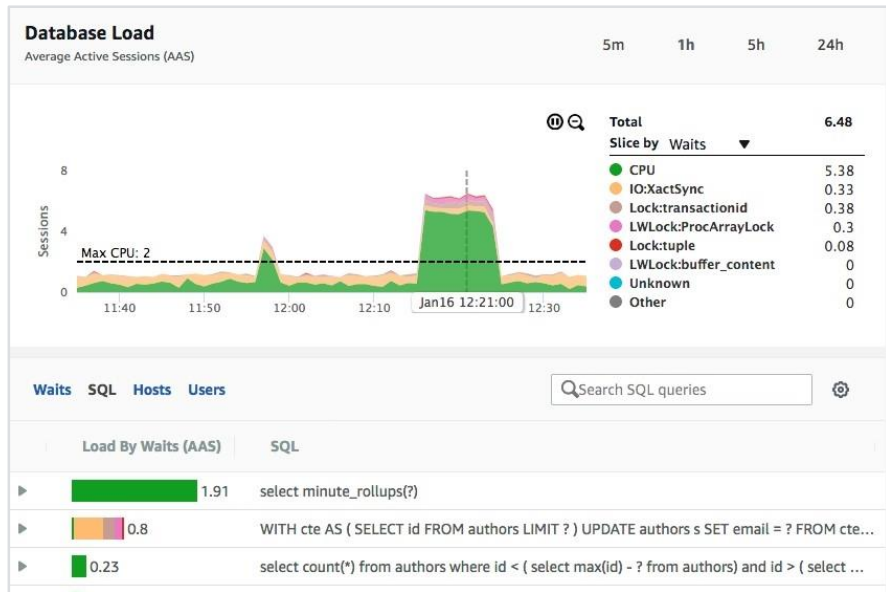
Enhanced Monitoring for Amazon RDS

- Access to over 50 CPU, memory, file system, and disk I/O metrics
- As low as 1 second intervals

Integration with third-party monitoring tools

Hack #9: Enable RDS Performance Insights

Built-in database performance monitoring

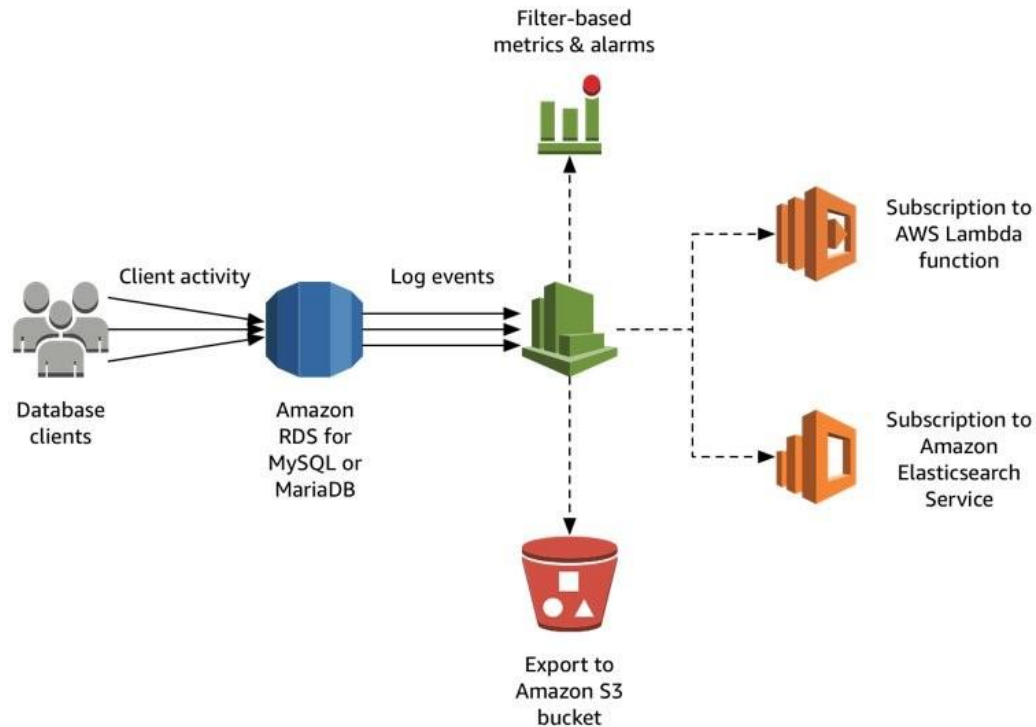


- Measures database load to help you identify database bottlenecks
- Top SQL/most intensive queries
- Enables problem discovery
- Adjustable timeframe
 - Hour, day, week, and longer

Hack #10: Publish Logs to CloudWatch Logs

Consolidate and archive database events

- Real time publishing for RDS for MySQL, MariaDB and Amazon Aurora
- Supported logs
 - General log
 - Slow query log
 - Audit log
 - Error log
- Set up CloudWatch Alarms
- Specify a retention period



Bonus Hack: Enable the Audit plug-in

- Provides customer configurable event logging for database activity
 - Auditable events include logins, queries, and tables accessed
 - Individual users can be included or excluded from the audit
- The audit plug-in is supported on the following RDS engines
 - MySQL 5.6, 5.7
 - MariaDB 10.0, 10.1, 10.2
 - Amazon Aurora
- Available via RDS option group
- Test for impact on server performance



Hundreds of thousands of customers use Amazon RDS



**Learn
more**

aws.amazon.com/rds

Thank you!