Hướng dẫn nâng version MySQL 5.7 lên MySQL 8.0 trên AWS RDS

**1. Vấn đề**

- Do phiên bản MySQL 5.7 đã cũ và đã không còn nhiều bên còn hỗ trợ ,cần upgrade lên version 8  
- Mỗi phiên bản của không chỉ MySQL mà đa số các phần mềm đều có định dạng như sau MAJOR.MINOR.PATCH, do đó việc upgrade MySQL 5.7 → 8.0, đồng nghĩa với việc bạn thực hiện một MAJOR upgrade và buộc phải upgrade bằng tay.

**2. Kiểm tra trước nâng cấp**

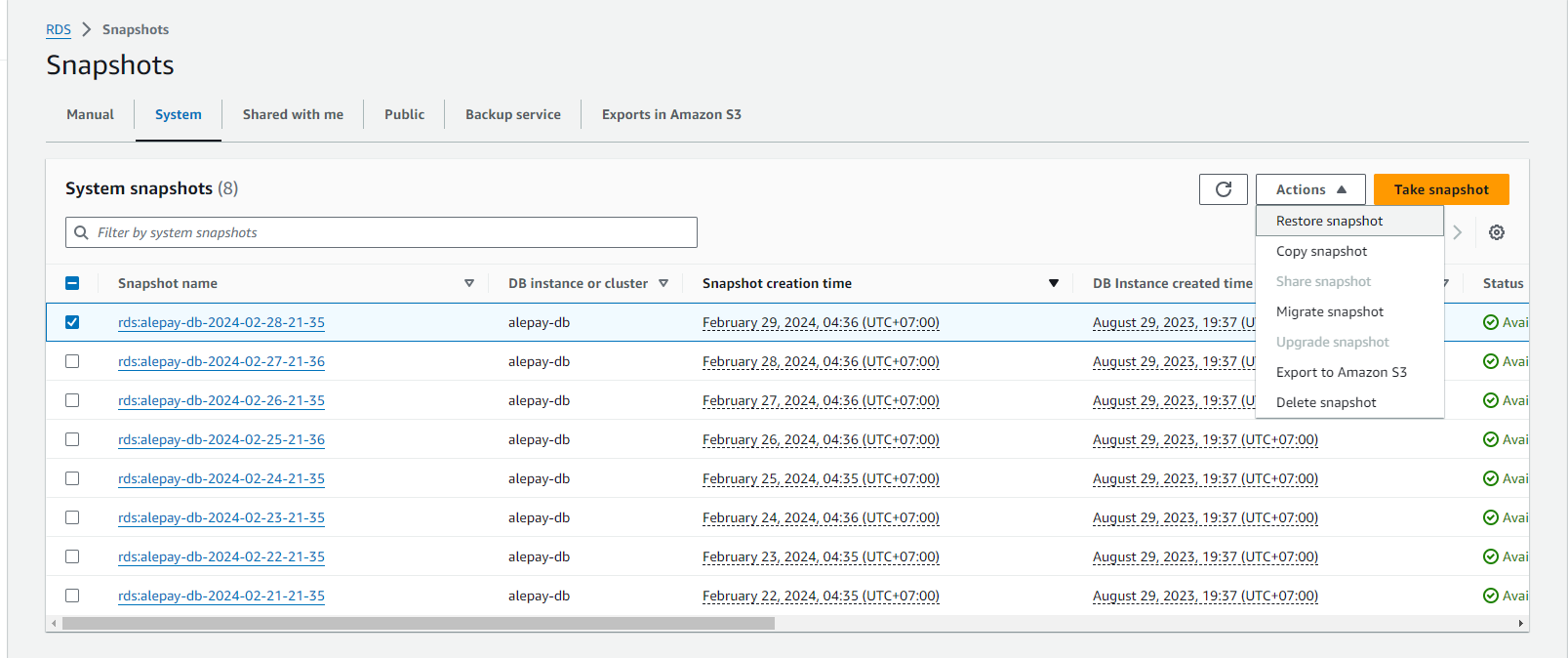
Để đảm bảo việc nâng cấp suôn sẻ thì bạn cần đảm bảo các yêu cầu sau:

* Không có tables nào sử dụng các kiểu dữ liệu và hàm đã lỗi thời. Nếu có thì bạn cần sử dụng REPAIR TABLE trước khi nâng cấp sang bản 8.0.
* Không tồn tại các orphan \*.frm files, keyword violations
* Trigger nếu có phải đảm bảo không bị thiếu
* Không có bảng nào trong MySQL 5.7 mysql system database có cùng tên với table MySQL 8.0 data dictionary
* Không được có bảng hoặc stored procedures với các phần tử riêng lẻ ENUMhoặc SETcột có độ dài vượt quá 255 ký tự hoặc 1020 byte
* Cấu hình cài đặt MySQL 5.7 của bạn không được sử dụng các tính năng không được hỗ trợ trong MySQL 8.0 (<https://dev.mysql.com/doc/refman/8.0/en/mysql-nutshell.html#mysql-nutshell-removals>)
* Không được có tên ràng buộc khóa ngoại dài hơn 64 ký tự
* Để cải thiện hỗ trợ Unicode, hãy xem xét chuyển đổi các đối tượng sử dụng utf8mb3bộ ký tự để sử dụng utf8mb4bộ ký tự. Bộ utf8mb3ký tự không được dùng nữa. Ngoài ra, hãy cân nhắc sử dụng utf8mb4cho các tham chiếu bộ ký tự thay vì utf8, vì hiện tại utf8là bí danh cho utf8mb3bộ ký tự.

Chi tiết hơn xem tại [(https://dev.mysql.com/doc/refman/8.0/en/upgrade-prerequisites.html](https://dev.mysql.com/doc/refman/8.0/en/upgrade-prerequisites.html))

**Bật sao lưu dự phòng**

**Tạo snapshot trước khi upgrade sẽ giúp giảm thời gian upgrade đi đáng kể**

[](https://book.saobang.vn/uploads/images/gallery/2024-02/6CC3PgFa50PrMVAN-image-1709195322512.png)

Nếu dùng AWS CLI ta chạy lệnh sau :

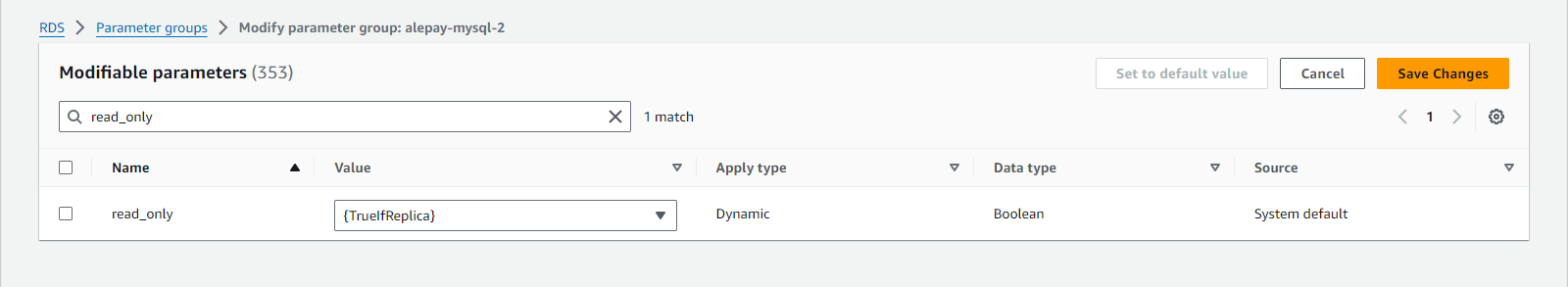
aws rds create-db-snapshot \

--db-instance-identifier $db\_instance\_name \

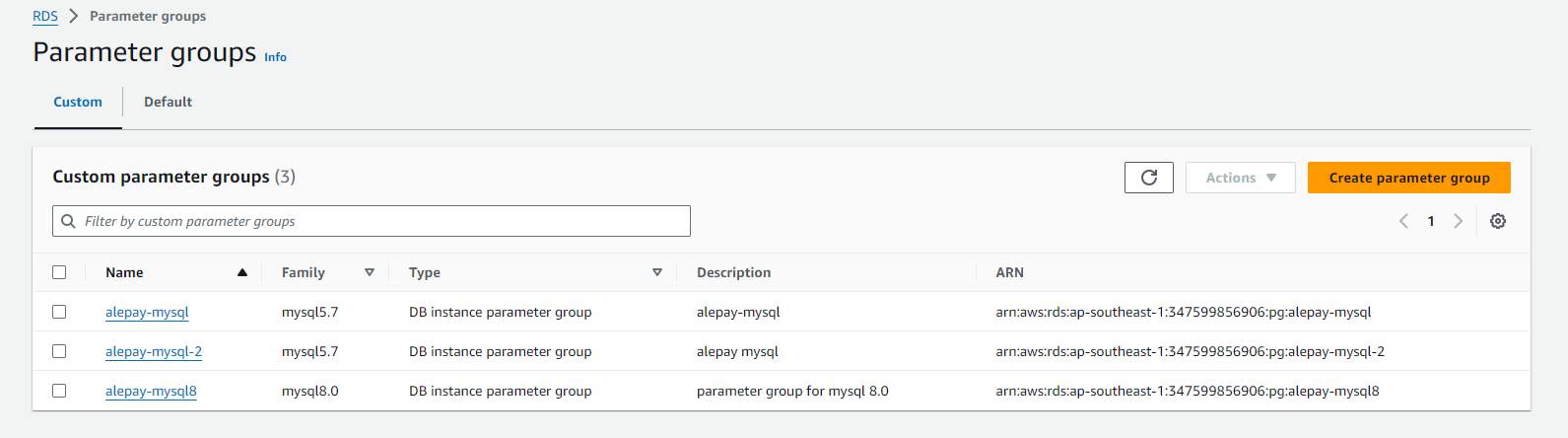
--db-snapshot-identifier $db\_snapshot\_name

Để đảm bảo sự toàn vẹn của dữ liệu và không xảy ra mất mát dữ liệu, cấu hình tham số read\_only trong parameter group về 1

để tránh việc ghi dữ liệu vào database primary trong quá trình upgrade

[](https://book.saobang.vn/uploads/images/gallery/2024-02/ydA1bZafmXFjLPJe-image-1709195374898.png)

Tạo parameter group chuẩn bị trước cho RDS MySQL 8.0.36 để khi nâng cấp thành công sẽ gán DB cho parameter group mới tương thích với version 8.0.36

[](https://book.saobang.vn/uploads/images/gallery/2024-02/RpY6YpmFfhNPlnwU-image-1709196087295.png)

**3. Tiến hành upgrade**

**Chúng ta sẽ tiến hành upgrade theo  2 phương pháp chính như sau**

* **Nâng cấp trực tiếp**

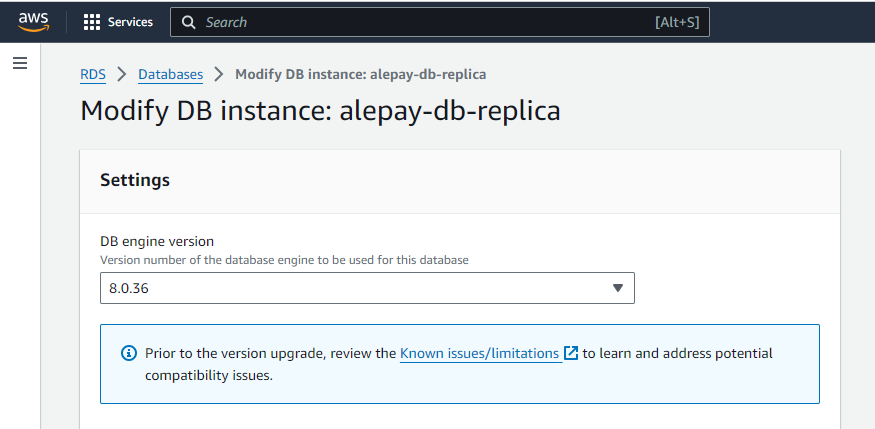
aws rds modify-db-instance \

--db-instance-identifier $db\_instance\_name \

--engine-version $new\_mysql\_version \

--allow-major-version-upgrade \

--apply-immediately

[](https://book.saobang.vn/uploads/images/gallery/2024-02/UOCZqDQeujZPRagJ-image-1709195431191.png)

Nâng cấp công cụ cơ sở dữ liệu yêu cầu thời gian ngừng hoạt động. Khoảng thời gian ngừng hoạt động thay đổi tùy theo kích thước phiên bản DB của bạn.

Quá trình nâng cấp phiên bản chính của MySQL thường hoàn tất sau khoảng 10 phút. Một số nâng cấp có thể mất nhiều thời gian hơn do kích thước lớp phiên bản DB hoặc do phiên bản không tuân theo các nguyên tắc vận hành nhất định trong Thực tiễn tốt nhất cho Amazon RDS

Để kiểm tra DB instance đã được upgrade lên đúng phiên bản mong muốn, sử dụng lệnh sau :

aws rds describe-db-instances \

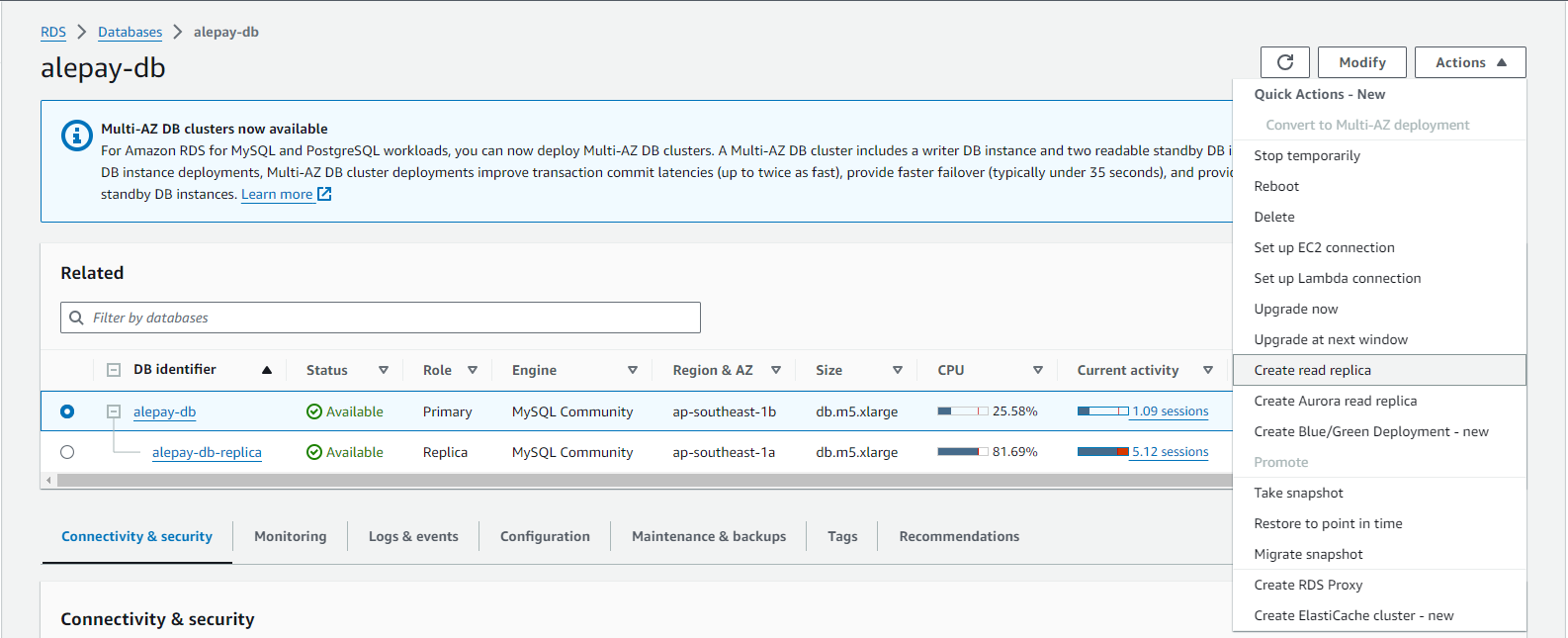
--db-instance-identifier $db\_instance\_name \

--query 'DBInstances[\*].{DBInstanceIdentifier:DBInstanceIdentifier,DBInstanceStatus:DBInstanceStatus,EngineVersion:EngineVersion}'

* **Sử dụng Replica DB chỉ có quyền đọc để giảm thời gian ngừng hoạt động**

Để xây dựng môi trường test chúng ta có thể làm theo các bước sau :

Tạo bản sao replica của phiên bản MySQL 5.7 của bạn.

[](https://book.saobang.vn/uploads/images/gallery/2024-02/aahjPr6yyikx4npg-image-1709195528510.png)

read\_replica\_db\_instance\_name="read-replica-rds-upgrade-testing"

​

# Create read replica

aws rds create-db-instance-read-replica \

--db-instance-identifier $read\_replica\_db\_instance\_name \

--source-db-instance-identifier $db\_instance\_name

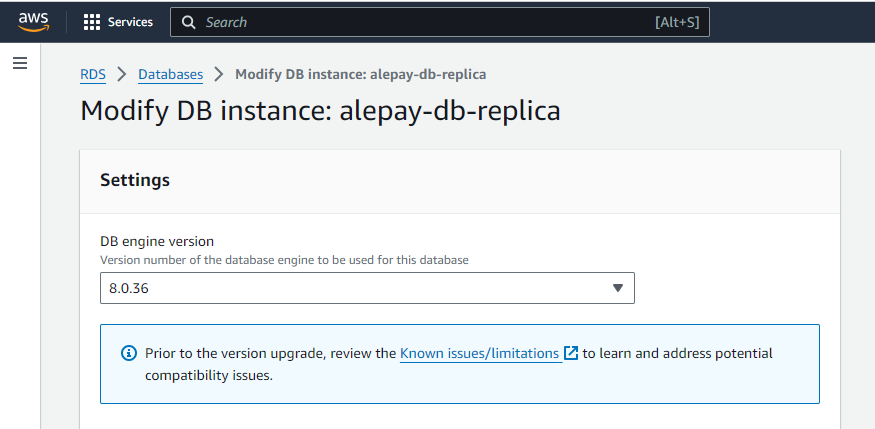
​

# Wait util read replica db instance becomes available (~8 minutes)

aws rds wait db-instance-available --db-instance-identifier $read\_replica\_db\_instance\_name

(Không bắt buộc) Khi read replica được khởi tạo và Status hiển thị Available, chuyển đổi read replica sang Multi-AZ deployment và bật tính năng backups.

Khi read replica Status hiển thị Available, Nâng cấp read replica sang MySQL 8.0.

[](https://book.saobang.vn/uploads/images/gallery/2024-02/UOCZqDQeujZPRagJ-image-1709195431191.png)

# Upgrade

aws rds modify-db-instance \

--db-instance-identifier $read\_replica\_db\_instance\_name \

--engine-version $new\_mysql\_version \

--allow-major-version-upgrade \

--apply-immediately

​

# Wait util read replica db instance becomes available (~13 minutes)

aws rds wait db-instance-available --db-instance-identifier $read\_replica\_db\_instance\_name

​

# Ensure that upgraded read replica has latest MySQL engine version

aws rds describe-db-instances \

--db-instance-identifier $read\_replica\_db\_instance\_name \

--query 'DBInstances[\*].{DBInstanceIdentifier:DBInstanceIdentifier,DBInstanceStatus:DBInstanceStatus,EngineVersion:EngineVersion}'

Khi quá trình upgrade hoàn thành Status hiển thị Available hãy xác nhận lại xem read replica có được cập nhật với phiên bản MySQL 5.7 DB chưa.

(Không bắt buộc) Bạn có thể tạo thêm read replica để tăng HA.

(Không bắt buộc) Cấu hình các custom DB parameter group cho read replica.

Promote bản sao read replica lên thành 1 phiên bản DB độc lập

# Promote read replica

aws rds promote-read-replica \

--db-instance-identifier $read\_replica\_db\_instance\_name

​

# Wait util read replica db instance become available (~5 minutes)

aws rds wait db-instance-available --db-instance-identifier $read\_replica\_db\_instance\_name

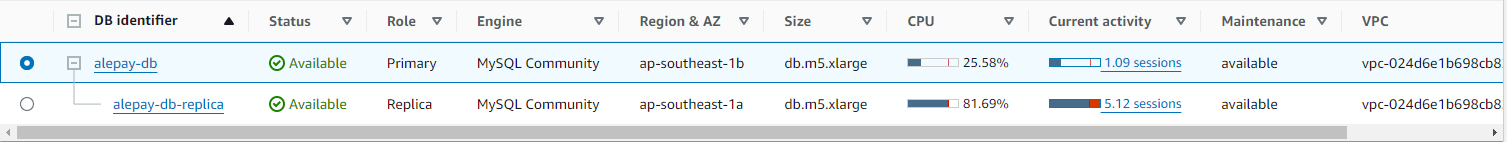
* **Thời gian Downtime**
  + Short downtime: Trong khi read replica instance tiến hành upgraded, ứng dụng của chúng ta vẫn có thể truy cập vào primary db instance. Sau quá trình upgrade tất cả các dữ liệu sẽ được đồng bộ từ primary sang read replica instance.

Downtime = Time to promote read replica to standalone instance + Time to change DNS endpoint/DB setting app.

Easy to rollback: Nếu chẳng may có bất kỳ vấn đề nào xảy ra trong quá trình nâng cấp, chúng ta có thể dễ dàng rollback lại bằng cách thay đổi DNS endpoint/DB endpoint sang v5.7 DB instance.

**4. Kiểm tra sau upgrade**

Chúng ta có thể sử dụng câu lệnh sau để kiểm tra xem trạng thái của DB instance đã sẵn sàng chưa

[](https://book.saobang.vn/uploads/images/gallery/2024-02/bqiSkHuGr6agEw5W-image-1709195781101.png)

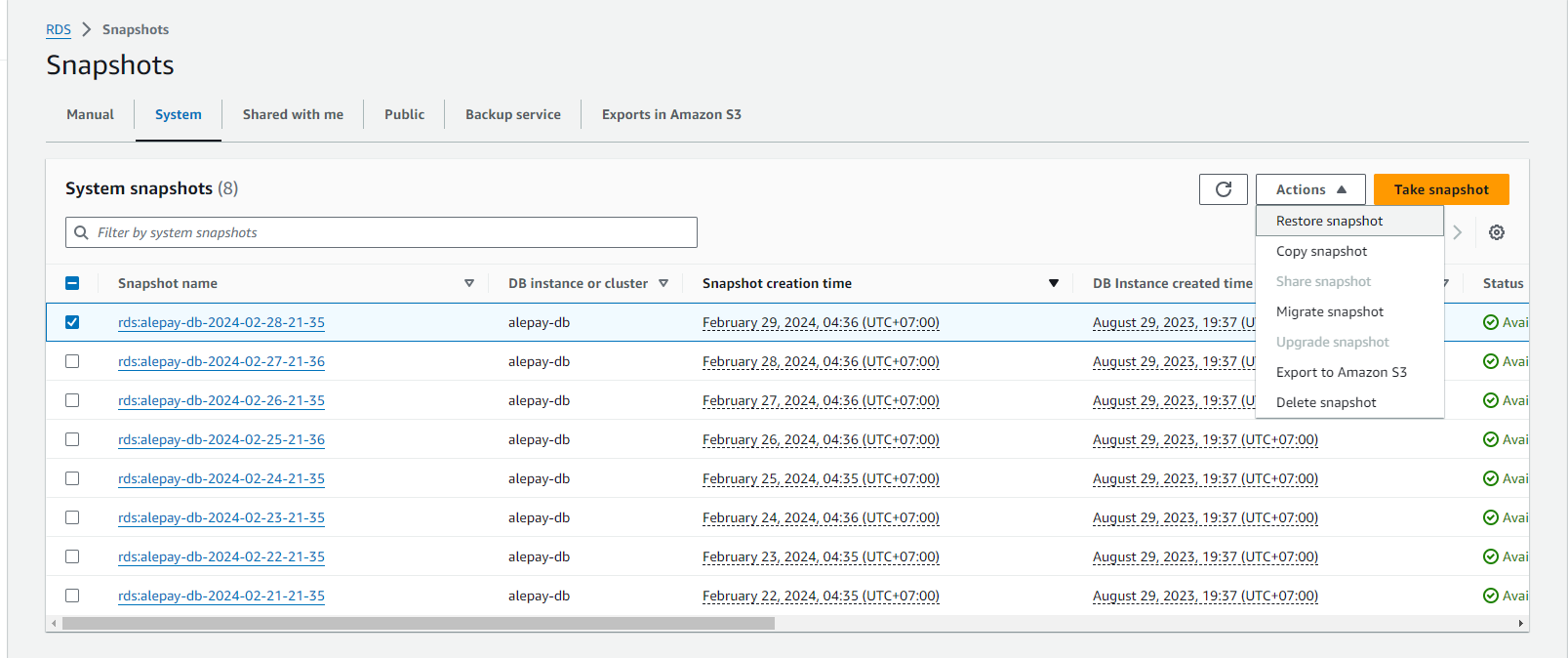
1

aws rds wait db-instance-available --db-instance-identifier $db\_instance\_name

Kết nối ứng dụng vào new DB instance, và thực hiện retest hành vi hệ thống.

**5. Rollback trong trường hợp upgrade failed**

Trường hợp Upgrade directly

* Chúng ta sẽ không thể rollback engine hiện tại (v8.0) về engine phiên bản trước (v5.7). Mà sẽ buộc phải upgrade latest snapshot được tạo trước khi tiến hành nâng cấp:
* [](https://book.saobang.vn/uploads/images/gallery/2024-02/6CC3PgFa50PrMVAN-image-1709195322512.png)

new\_db\_instance\_name="new-rds-upgrade-testing"

​

aws rds restore-db-instance-from-db-snapshot \

--db-instance-identifier $new\_db\_instance\_name \

--db-snapshot-identifier $db\_snapshot\_name

* Cập nhật lại endpoints / Route 53 trỏ về new RDS, và connection endpoint trong ứng dụng sang new MySQL RDS instance.

Trường hợp Upgrade using read replica

* Chúng ta chỉ cần upgrade DNS endpoint sang DB endpoint (v5.7) cũ vì DB instance này chạy hoàn toàn độc lập với RDS master instance (v8.0).

**6. Kết luận**

* Phương án 2 được khuyến khích sử dụng vì những lợi ích sau
  + Thời gian ngừng hoạt động ngắn
  + Dễ dàng quay trở lại.
* Dù dùng cách gì thì chúng ta cũng cần cẩn thận kiểm tra, diễn tập trước khi tiến hành, và đây là các đầu mục lớn cần làm:
  + Kiểm tra trước để nâng cấp.
  + Sao lưu dữ liệu vào ảnh chụp nhanh.
  + Testing an upgrade with test DB instance. ( Tạo một snapshot từ staging db instance => Tạo một DB instance mới từ snapshot này. Thử tiến hành upgrade trên DB instance này).
  + Để hạn chế lỗi, hãy biến toàn bộ các thao tác về scripts: backup, upgrade and rollback vì làm như vậy sẽ giảm thiểu rủi ro, nhớ nhớ quên quên.
* Tính toán sơ bộ thời gian dùng để upgrade: việc này có thể phụ thuộc vào số lượng instance, kích thước database, phương án upgrade, database schema, engine, …

<https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/USER_UpgradeDBInstance.MySQL.html#USER_UpgradeDBInstance.MySQL.ReducedDowntime>

**7. Một vài lưu ý cần chỉnh sửa sau khi test với app**

* Check lại 1 lượt plugin authentication ứng với các user phải được đặt là *mysql\_native\_password*chứ không phải là *caching\_sha2\_password,*nếu chưa đúng chuẩn hãy chuyển plugin bằng lênh sau :

ALTER USER 'bireport'@'%' IDENTIFIED WITH mysql\_native\_password BY 'password';

* Kiểm tra lại với dev về kiểu dữ liệu bảng HistoryMerchantConfig và MerchantConfig trong DB Alepay-live đã thay kiểu dữ liệu medium text sang varchar hay chưa
* Thêm trường status vào bảng EmailOutbox bằng lệnh sau và kiểm tra lại :

ALTER TABLE EmailOutbox ADD COLUMN status VARCHAR(100) DEFAULT NULL;

DESCRIBE EmailOutbox;