Fast, Reliable, Secure, Affordable MongoDB on AWS EC2

Paul Marcelin
Why are we here today?

• Among open-source databases, MongoDB is a **complete** production-ready solution

• Self-managing MongoDB is **worthwhile**, for the best AWS performance at the lowest cost

• A few simple but not widely-understood AWS tips ★ **prevent** most AWS performance, cost, and security problems
Compute for MongoDB

• Choose between 3 major instance families...
  • Memory-optimized
  • Compute-optimized
  • General-purpose
• And one with a twist:
  • Burstable performance
• Use the latest generation
  • Better performance, lower unit price
Disk for MongoDB

- Network storage as a service: Elastic Block Store
  - Affordable: from 10¢ per gigabyte per month
  - Reliable: multiple copies; decoupled from compute
  - Convenient: snapshots; online volume enlarge

- Don’t use Provisioned IOPS SSD (io1 volumes) or local SSD (i3, i3en instances) before you:
  - Optimize for EBS general-purpose SSD (gp2) volumes
  - Study CloudWatch metrics data

Check with AWS for official prices.
Optimize for EBS General-Purpose SSD

- Use latest-generation instances
  - More EBS bandwidth
  - Less EBS overhead
- Use larger instances
  - Even more bandwidth!
- Enlarge your general-purpose (gp2) volumes
  - More operations per second at a much lower cost than Provisioned IOPS (io1)

\[
gp2 \text{ IOPS: } \text{Base} = 3 \times \text{volume size in GB} \\
\text{Max} = 16,000 \text{ IOPS (5.3 to 16 TB volume size)}
\]
Back Up MongoDB with EBS Snapshots

- Create at least two separate EBS volumes
  1. Operating system + software
  2. Data + journal
- Enable journaling
- Keep journal on same volume as data
- Take frequent snapshots
  - You pay only for changed disk blocks
  - AWS Backups: every 12 or 24 hours
  - [github.com/sqlxpert/aws-tag-sched-ops](https://github.com/sqlxpert/aws-tag-sched-ops): up to every 10 minutes
Basic Fault Tolerance

- Each AWS region has multiple availability zones, in separate locations
- Distribute your MongoDB replicaSet across availability zones
- Plan for a 2-zone minimum in some regions
- Multi-region replicaSets are possible (but consider network latency)
More Fault Tolerance

- Distribute your application layer, not just your database!
- Always use replicaSet connection strings for application traffic
Yet More Fault Tolerance

• Detect and replace failed replicaSet members
• Make this automatic, or at least quick and convenient
• Potential components:
  • An EC2 Fleet automatically replaces failed instances
  • A configuration management system (AWS OpsWorks, Chef, Ansible, SaltCloud, etc.) configures each new instance upon first boot
  • A pipeline updates a base Amazon Machine Image (AMI)
Basic Security Elements

- AWS Key Management System (KMS) customer-managed key
  - Encrypts disks and snapshots
- TLS certificates (not from AWS)
  - Encrypt application and replication traffic
  - Validate server identity
- AWS security groups
  - Enforce network firewall rules
  - Also validate server identity (within the same AWS region)
Understand Disk Encryption (EBS + KMS)

- Use customer-managed keys, not your default EBS service key!
- Create a separate key for every MongoDB replicaSet
- Encrypt both your data volume and your OS volume
- Snapshots of encrypted volumes are necessarily encrypted
- Edit the key policy to limit the people who can:
  - Attach encrypted volumes to instances
  - Create volumes from snapshots
  - Copy snapshots
Take Full Advantage of TLS Certificates

- Every replicaSet member needs its own TLS certificate (and DNS record)
- Underlaying an AWS Route 53 DNS private hosted zone may help!
- The private key should live only as long as the replicaSet member
- Obtain certificates from a third party; don’t self-sign
- My personal favorites:
  - Let’s Encrypt (free)
  - DigiCert’s duplicate certificate feature + a wildcard certificate (worth the cost)
Understand AWS Security Groups

• A security group is a set of network firewall rules
• These rules can only allow, not block
• Traffic that’s not allowed is blocked, but...
• If you send out a request, the response is always allowed in ("stateful")
• Never use a default security group!
• An instance can be a member of multiple security groups
• Never reference same-region instances by IP address; instead, identify source and destination instances by their security groups
• If you only police inbound traffic, put all instances in an all-outbound group and delete the default all-outbound rule from all other groups
Define Security Groups by Membership

all-out
- All traffic *out* to 0.0.0.0/0

**mongodb-server**
TCP 27017 *in* from:
- mongodb-server (replication)
- mongodb-client (application data)

**mongodb-client**
No rules; just identifies clients

**https-server**
- TCP 443 *in* from 0.0.0.0/0
Advanced Security Elements

- AWS Identity and Access Management EC2 instance role
  - Authorizes AWS API calls from an instance (including calls made by the Systems Manager agent)
  - No AWS API keys to rotate, distribute, and hold on disk
- Task-specific IAM roles
  - Grant specific people shell access to specific instances
- AWS Systems Manager – Session Manager
  - Provides shell access, with no SSH key pairs to manage
Summary: MongoDB on AWS EC2

- Three t3 instances, in multiple availability zones
- Instance role with *AmazonEC2RoleforSSM* policy
- AWS Systems Manager – Session Manager for shell access
- Security groups: all-out, mongodb-server, mongodb-client
- KMS key to encrypt only this replicaSet’s disks
- For each instance:
  - Large additional EBS gp2 volume for data + journal
  - Public + private DNS records and a TLS certificate
We’re Almost Done

• Thanks for coming!

• Keep in touch at marcelin@alumni.cmu.edu

• Try my template, github.com/sqlxpert/mongodb-percona-live

• Or try Amazon’s, aws.amazon.com/quickstart/architecture/mongodb/

• Don’t forget to rate this session...
Rate This Session

Introducing gh-ost: triggerless, painless, trusted online schema migrations

Tap to rate & review

Feedback (optional)

SUBMIT
Compute for MongoDB

- Choose between 3 major instance families...
  - Memory-optimized \( \text{R5} \) from $92 per month \( 16 \text{ GB RAM} \)
  - Compute-optimized \( \text{C5} \) \$63 \( 4 \text{ GB} \)
  - General-purpose \( \text{M5} \) \$71 \( 8 \text{ GB} \)
- And one with a twist:
  - Burstable performance \( \text{T3} \) \$31 \( 4 \text{ GB} \)
- Use the latest generation
  - Better performance, lower unit price

Smallest \( \text{t3: smallest reasonable} \) instance size, with 2 vCPUs in each case. Reserve, or use spot, for even lower prices. Check with AWS for official prices.
MongoDB + AWS Loose Ends

- Swap
  - MongoDB documentation recommends it!
  - Put it on a third EBS volume
  - Encrypt that volume (of course)
  - Extra work is required to put swap on local (instance store) volumes, for instance types (e.g., m5d) that offer local storage

- Customary configuration changes for MongoDB
  - Transparent Huge Pages: disable
  - File descriptor and process limits (`ulimit`): increase
  - Data volume mount options: add `noatime`