

通过迁移至 AWS Graviton2 实现显著的性能提升

EC2 存储优化型实例旨在提供高磁盘输入/输出性能和充足的存储空间。众多 AWS 的客户使用这些实例来托管高性能的实时数据库、分布式文件系统、数据仓库、键值存储等。

Im4gn 和 Is4gen

在最近 AWS 推出了两个最新存储优化型实例系列，即 **Im4gn** 和 **Is4gen**，它们由 Graviton2 处理器提供支持。这两个实例都使用 AWS 定制的 AWS Nitro SSD 设备，提供高达 30 TB 的 NVMe 存储空间。作为代表客户推动创新的一部分，AWS 将关注点转向了存储，并设计了多个设备，它们经过优化以支持高速访问大量数据。与第三代存储优化型实例相比，AWS Nitro SSD 可将输入/输出延迟降低多达 60%，且可将延迟可变性最多减少 75%。因此，客户可以为输入/输出密集型 EC2 工作负载获取更快、更可预测的性能。

Im4gn 实例非常适合需要大量 SSD 密集存储和高计算性能但内存密集性不是特别高的应用，例如社交游戏、会话存储、聊天机器人和搜索引擎。规格如下：

实例名称	vCPU 数量	内存	本地 NVMe 存储 (AWS Nitro SSD)	读取吞吐量 (128 KB 数据块)	EBS 优化带宽	网络带宽
im4gn.large	2	8 GiB	937 GB	250 MB/秒	最高 9.5 Gbps	最高 25 Gbps
im4gn.xlarge	4	16 GiB	1.875 TB	500 MB/秒	最高 9.5 Gbps	最高 25 Gbps
im4gn.2xlarge	8	32 GiB	3.75 TB	1 GB/秒	最高 9.5 Gbps	最高 25 Gbps
im4gn.4xlarge	16	64 GiB	7.5 TB	2 GB/秒	9.5 Gbps	25 Gbps
im4gn.8xlarge	32	128	15 TB	4 GB/秒	19 Gbps	50

		GiB	(2 个 7.5 TB)			Gbps
im4gn.16xlarge	64	256 GiB	30 TB (4 个 7.5 TB)	8 GB/秒	38 Gbps	100 Gbps

与 I3 实例相比，Im4gn 实例的性价比提高了 40%，每 TB 存储成本降低了 44%。新实例在 AWS 美国西部（俄勒冈）、美国东部（俄亥俄）、美国东部（弗吉尼亚北部）和欧洲（爱尔兰）区域推出，以按需型、竞价型、Savings Plan 和预留实例的形式提供。

Is4gen 实例非常适合对大量 SSD 存储进行大量随机输入/输出操作的应用。这包括共享文件系统、流处理、社交媒体监控和流媒体平台，所有这些都可以利用增加的存储密度，在本地保留更多数据。规格如下：

实例名称	vCPU 数量	内存	本地 NVMe 存储 (AWS Nitro SSD)	读取吞吐量 (128 KB 数据块)	EBS 优化带宽	网络带宽
is4gen.medium	1	6 GiB	937 GB	250 MB/秒	最高 9.5 Gbps	最高 25 Gbps
is4gen.large	2	12 GiB	1.875 TB	500 MB/秒	最高 9.5 Gbps	最高 25 Gbps
is4gen.xlarge	4	24 GiB	3.75 TB	1 GB/秒	最高 9.5 Gbps	最高 25 Gbps
is4gen.2xlarge	8	48 GiB	7.5 TB	2 GB/秒	最高 9.5	最高

					Gbps	25 Gbps
is4gen.4xlarge	16	96 GiB	15 TB (2 个 7.5 TB)	4 GB/秒	9.5 Gbps	25 Gbps
is4gen.8xlarge	32	192 GiB	30 TB (4 个 7.5 TB)	8 GB/秒	19 Gbps	50 Gbps

与 I3en 实例相比，Is4gen 实例的每 TB 存储成本降低 15%，计算性能提高多达 48%。新实例在 AWS 美国西部（俄勒冈）、美国东部（俄亥俄）、美国东部（弗吉尼亚北部）和欧洲（爱尔兰）区域推出，以按需型、竞价型、Savings Plan 和预留实例的形式提供。

除此之外，AWS 推出 AWS Graviton 快速启动，这是一项全新的计划，通过提供支持 Graviton 平台的 EC2 和其他托管服务的分步指导，可以更轻松地将工作负载迁移到 AWS Graviton：

- Amazon Elastic Compute Cloud (Amazon EC2) – EC2 为迁移提供了最灵活的环境，可以支持多种工作负载，例如 Web 应用程序、自定义数据库或分析。客户可以完全控制在 EC2 实例中运行的解释代码或编译代码。客户还可以使用许多支持 Arm64 架构的开源和商业软件产品。
- AWS Lambda — 迁移无服务器函数非常容易，特别是如果客户使用解释型运行时（如 Node.js 或 Python）。大多数情况下，客户只需要检查软件依赖项的兼容性即可。我在此博客文章中展示了几个示例。
- AWS Fargate – 如果客户的应用程序已经在容器中运行，或者客户计划对它们进行容器化，则 Fargate 效果最佳。通过使用多架构容器映像或映像清单中包含 Arm64 的映像，客户可以同时获得 Fargate 的无服务器优势和 Graviton 的性价比优势。
- Amazon Aurora – 关系数据库是许多应用程序的核心。如果客户需要与 PostgreSQL 或 MySQL 兼容的数据库，则可以使用 Amazon Aurora 来拥有由 Graviton 提供支持的高性能且全球可用的数据库。
- 亚马逊关系数据库服务 (RDS) – 与 Aurora 类似，PostgreSQL、MySQL 和 MariaDB 等 Amazon RDS 引擎可以使用基于 Graviton 的实例提供完全托管的关系数据库服务。

- Amazon ElastiCache – 当客户的工作负载需要超低延迟和高吞吐量时，客户可以使用 ElastiCache 加快应用程序的速度，在 Graviton 上运行完全托管的内存缓存，并与 Redis 或 Memcached 兼容。
- Amazon EMR – 借助 Amazon EMR，客户可以使用 Apache Spark、Apache Hive 和 Presto 等开源分析框架，在 Graviton 上运行大规模的分布式数据处理作业、交互式 SQL 查询和机器学习应用程序。

结论

通过将迁移托管服务及特定的工作负载迁移至 AWS Graviton2, 客户能够获得显著的性价比提升. 此外, 借助 ARM 架构的优势, AWS Lambda 和 Amazon EMR 也能够获得可观的性能提升。