1）一家公司正在将旧应用程序迁移到 Amazon EC2 实例。该应用程序使用源代码中存储的用户名和密码以连接到一个 MySQL 数据库。公司将该数据库迁移到适用于 MySQL 的 Amazon RDS 数据库实例。作为迁移的一部分，公司需要实施一种安全的方法以存储并自动轮换数据库凭证。

哪种解决方案将满足这些要求?

A) 将数据库凭证存储在 Amazon Machine Image (AMI) 上的环境变量中。替换 AMI 以轮换这些凭证。
B) 将数据库凭证存储在 AWS Systems Manager Parameter Store 中。将 Parameter Store 配置为自动轮换这些凭证。
C) 将数据库凭证存储在 EC2 实例上的环境变量中。重新启动 EC2 实例以轮换这些凭证。
D) 将数据库凭证存储在 AWS Secrets Manager 中。将 Secrets Manager 配置为自动轮换这些凭证。

2）开发人员正在创建一个 Web 应用程序，该应用程序必须使用户能够近乎实时地发表评论和接收反馈。

哪些解决方案将满足这些要求? （请选择两项）

A) 创建 AWS AppSync 架构和相应的 API。将 Amazon DynamoDB 表作为数据存储。
B) 在 Amazon API Gateway 中创建 WebSocket API。将 AWS Lambda 函数作为后端。将 Amazon DynamoDB 表作为数据存储。
C) 创建一个由 Amazon RDS 数据库支持的 AWS Elastic Beanstalk 应用程序。将该应用程序配置为允许长时间运行的 TCP/IP 套接字。
D) 在 Amazon API Gateway 中创建一个 GraphQL 终端节点。将 Amazon DynamoDB 表作为数据存储。
E) 建立到 Amazon CloudFront 的 WebSocket 连接。将 AWS Lambda 函数作为 CloudFront 分配的来源。将 Amazon Aurora 数据库集群作为数据存储。

3）开发人员正在将注册和登录功能添加到一个应用程序中。该应用程序必须对一个自定义分析解决方案进行 API 调用以记录用户登录事件。

开发人员应采取哪些组合措施以满足这些要求? （请选择两项）

A) 使用 Amazon Cognito 提供注册和登录功能。

B) 使用 AWS Identity and Access Management (IAM) 提供注册和登录功能。

C) 配置 AWS Config 规则以在验证用户身份时进行 API 调用。

D) 调用 Amazon API Gateway 方法以在验证用户身份时进行 API 调用。

E) 调用 AWS Lambda 函数以在验证用户身份时进行 API 调用。

4) 一家公司正在一个 AWS 账户中将 Amazon API Gateway 用于其 REST API。开发人员希望仅允许另一个 AWS 账户中的 IAM 用户访问这些 API。

开发人员应采取哪些组合措施以满足这些要求? (请选择两项)

A) 创建一个 IAM 权限策略。将该策略附加到每个 IAM 用户。将这些 API 的方法授权类型设置为 AWS_IAM。使用签名版本 4 对 API 请求进行签名。

B) 创建一个 Amazon Cognito 用户池。将每个 IAM 用户添加到该用户池中。将这些 API 的方法授权类型设置为 COGNITO_USER_POOLS。使用 Amazon Cognito 中的 IAM 凭证进行身份验证。将 ID 令牌添加到请求标头中。

C) 创建一个 Amazon Cognito 身份池。将每个 IAM 用户添加到该身份池中。将这些 API 的方法授权类型设置为 COGNITO_USER_POOLS。使用 Amazon Cognito 中的 IAM 凭证进行身份验证。将访问令牌添加到请求标头中。

D) 为这些 API 创建资源策略以仅允许每个 IAM 用户进行访问。

E) 为这些 API 创建 Amazon Cognito 授权方以仅允许每个 IAM 用户进行访问。将这些 API 的方法授权类型设置为 COGNITO_USER_POOLS。

5) 开发人员正在构建新的应用程序以将文本文件转换为 .pdf 文件。一个单独的应用程序将文本文件写入到源 Amazon S3 存储桶中。新应用程序必须在文件到达 Amazon S3 时读取这些文件，并且必须使用 AWS Lambda 函数将这些文件转换为 .pdf 文件。开发人员编写了一个 IAM 策略以允许访问 Amazon S3 和 Amazon CloudWatch Logs。

开发人员应采取哪种措施以确保 Lambda 函数具有正确的权限?

A) 使用 AWS Identity and Access Management (IAM) 创建一个 Lambda 执行角色。将 IAM 策略附加到该角色。将 Lambda 执行角色分配给 Lambda 函数。

B) 使用 AWS Identity and Access Management (IAM) 创建一个 Lambda 执行用户。将 IAM 策略附加到该用户。将 Lambda 执行用户分配给 Lambda 函数。

C) 使用 AWS Identity and Access Management (IAM) 创建一个 Lambda 执行角色。将 IAM 策略附加到该角色。将 IAM 角色作为环境变量存储在 Lambda 函数中。

D) 使用 AWS Identity and Access Management (IAM) 创建一个 Lambda 执行用户。将 IAM 策略附加到该用户。将 IAM 用户凭证作为环境变量存储在 Lambda 函数中。

6) 开发人员正在开发一个应用程序以将高度机密的数据存储在数据库中。开发人员必须使用具有信封加密的 AWS Key Management Service (AWS KMS) 以保护数据。

开发人员应如何配置数据加密以满足这些要求?

A) 使用 KMS 密钥加密数据。将加密的数据存储在数据库中。

B) 使用生成的数据密钥加密数据。将加密的数据存储在数据库中。

C) 使用生成的数据密钥加密数据。将加密的数据和数据密钥 ID 存储在数据库中。

D) 使用生成的数据密钥加密数据。将加密的数据和加密的数据密钥存储在数据库中。

7) 开发人员正在将 Amazon ElastiCache for Memcached 添加到公司的现有记录存储应用程序中。根据常见的记录处理模式分析，开发人员决定使用延迟加载。

哪个伪代码示例将正确实施延迟加载?

A)
```
record_value = db.query("UPDATE Records SET Details = {1} WHERE ID == {0}",
                        record_key, record_value)
cache.set (record_key, record_value)
```

B)
```
record_value = cache.get(record_key)
if (record_value == NULL)
    record_value = db.query("SELECT Details FROM Records WHERE ID == {0}",
                            record_key)
    cache.set (record_key, record_value)
```

C)
```
record_value = cache.get (record_key)
db.query("UPDATE Records SET Details = {1} WHERE ID == {0}", record_key,
         record_value)
```

D) record_value = db.query("SELECT Details FROM Records WHERE ID == {0}",
record_key)
if (record_value != NULL)
cache.set (record_key, record_value)

8) 开发人员正在构建一个使用 Amazon API Gateway 的 Web 应用程序。开发人员希望为开发（dev）和生产（prod）工作负载维护不同的环境。该 API 将由具有两个别名的 AWS Lambda 函数提供支持：一个别名用于 dev，另一个别名用于 prod。

开发人员如何以最少的配置维护这些环境?

A) 为每个环境创建一个 REST API。将这些 API 与 Lambda 函数的相应 dev 和 prod 别名集成在一起。将这些 API 部署到相应的阶段中。使用阶段 URL 访问这些 API。
B) 创建一个 REST API。使用阶段变量代替别名以将该 API 与 Lambda 函数集成在一起。将该 API 部署到两个不同的阶段中：dev 和 prod。在每个阶段中创建一个阶段变量，并将不同的别名作为值。使用不同的阶段 URL 访问该 API。
C) 创建一个 REST API。将该 API 与 Lambda 函数的 dev 别名集成在一起。将该 API 部署到 dev 环境中。为 prod 环境配置 Canary 版本部署，在该环境中，Canary 将与 Lambda prod 别名集成在一起。
D) 创建一个 REST API。将该 API 与 Lambda 函数的 prod 别名集成在一起。将该 API 部署到 prod 环境中。为 dev 环境配置 Canary 版本部署，在该环境中，Canary 将与 Lambda dev 别名集成在一起。

9) 开发人员希望跟踪在一组 Amazon EC2 实例上运行的应用程序的性能。开发人员希望查看和跟踪整个实例集的统计数据，例如平均请求延迟和最大请求延迟。开发人员希望在平均响应时间超过阈值时立即收到通知。

哪种解决方案将满足这些要求?

A) 在每个 EC2 实例上配置 cron 任务以测量响应时间，并且每分钟更新一次 Amazon S3 存储桶中存储的日志文件。使用 Amazon S3 事件通知调用 AWS Lambda 函数以读取日志文件，并将新条目写入到 Amazon OpenSearch Service 集群中。在 OpenSearch 控制面板中可视化这些结果。配置 OpenSearch Service，以便在响应时间超过阈值时向 Amazon Simple Notification Service (Amazon SNS) 主题发送警报。

B) 配置应用程序以将响应时间写入到系统日志中。在 EC2 实例上安装并配置 Amazon Inspector 代理以持续读取日志，并将响应时间发送到 Amazon EventBridge (Amazon CloudWatch Events)。在 EventBridge (CloudWatch Events) 控制台中查看指标图表。配置 EventBridge (CloudWatch Events) 自定义规则，以便在响应时间指标平均值超过阈值时发送 Amazon Simple Notification Service (Amazon SNS) 通知。

C) 配置应用程序以将响应时间写入到日志文件中。在 EC2 实例上安装并配置 Amazon CloudWatch 代理，以将应用程序日志流式传输到 CloudWatch Logs。从日志中创建响应时间指标筛选条件。在 CloudWatch 控制台中查看指标图表。创建 CloudWatch 警报，以便在响应时间指标平均值超过阈值时发送 Amazon Simple Notification Service (Amazon SNS) 通知。

D) 在 EC2 实例上安装并配置 AWS Systems Manager 代理（SSM 代理）以监控响应时间，并将响应时间作为自定义指标发送到 Amazon CloudWatch。 在 Amazon QuickSight 中查看指标图表。 创建 CloudWatch 警报，以便在响应时间指标平均值超过阈值时发送 Amazon Simple Notification Service (Amazon SNS) 通知。

10) 开发人员正在本地测试一个应用程序，并且已将该应用程序部署到 AWS Lambda 函数中。为了避免超过部署程序包大小配额，开发人员未在部署文件中包含依赖项。在开发人员远程测试该应用程序时，由于缺少依赖项，Lambda 函数无法运行。

哪种解决方案将解决该问题?

A) 使用 Lambda 控制台编辑器更新代码并包含缺少的依赖项。

B) 创建一个包含缺少的依赖项的额外 .zip 文件。在原始 Lambda 部署程序包中包含该 .zip 文件。

C) 在 Lambda 函数的环境变量中添加对缺少的依赖项的引用。

D) 创建一个包含缺少的依赖项的层。将该层附加到 Lambda 函数中。

答案

1) D - AWS Secrets Manager 帮助保护访问数据库、应用程序、服务和其他 IT 资源所需的凭证。借助 Secrets Manager，您可以在整个生命周期中轮换、管理和检索数据库凭证、API 密钥和其他密钥。用户和应用程序进行 Secrets Manager API 调用以检索密钥，而无需以明文形式硬编码敏感信息。Secrets Manager 提供密钥轮换，并具有内置的 Amazon RDS、Amazon Redshift 和 Amazon DocumentDB（兼容 MongoDB）集成。

2) A、B - AWS AppSync 使您能够创建灵活的 API 以安全地访问、处理和合并来自一个或多个数据源的数据，从而简化了应用程序开发。AWS AppSync 是一种托管服务，它使用 GraphQL 帮助应用程序获取所需的确切数据。您可以使用 AWS AppSync 构建需要实时更新一系列数据源（包括 Amazon DynamoDB）的可扩展应用程序。

在 Amazon API Gateway 中，您可以创建 WebSocket API 以作为 AWS 服务（例如 AWS Lambda 或 DynamoDB）或 HTTP 终端节点的有状态前端。WebSocket API 根据它从客户端应用程序中收到的消息内容调用后端。与接收和响应请求的 REST API 不同，WebSocket API 支持客户端应用程序和后端之间的双向通信。

3) A、E - Amazon Cognito 将用户注册、登录和访问控制添加到 Web 和移动应用程序中。您也可以创建一个 AWS Lambda 函数以对自定义分析解决方案进行 API 调用，然后使用 Amazon Cognito 身份验证后触发器调用该函数。

4) A、D - 资源策略可以使用签名版本 4 (SigV4) 协议为一个 AWS 账户中的用户授予另一个 AWS 账户中的 API 访问权限。

5) A - AWS Lambda 函数的执行角色为 Lambda 函数授予权限以访问 AWS 服务和资源。您在创建函数时提供该角色，而 Lambda 在调用函数时担任该角色。

6) D - 信封加密是使用数据密钥加密明文数据并使用另一个密钥加密数据密钥的做法。您必须存储加密形式的数据密钥，以便您可以使用数据密钥解密数据库中的加密数据。

7) B - 延迟加载是一种缓存策略，直到需要使用记录时才会加载该记录。在您实施延迟加载时，应用程序先在缓存中查找记录。如果记录不存在，则应用程序从数据库中检索记录，并将记录存储在缓存中。

8) B - 通过使用 Amazon API Gateway 中的部署阶段，您可以管理每个 API 的多个版本阶段。您可以配置阶段变量，以便 API 部署阶段可以与不同的后端终端节点进行交互。您可以使用 API Gateway 阶段变量以引用一个具有多个版本和别名的 AWS Lambda 函数。

9) C - 您可以配置 Amazon CloudWatch 代理以将日志和指标流式传输到 CloudWatch。您也可以从 CloudWatch Logs 上存储的日志中创建指标筛选条件。

10) D - 您可以将 AWS Lambda 函数配置为以层的形式提取额外的代码和内容。层是一个 .zip 文件归档，其中包含库、自定义运行时或其他依赖项。借助于层，您可以在 Lambda 函数中使用库，而无需将库包含在部署程序包中。