

数据湖：设计更好的架构、存储、安全和数据治理

前言

对任何业务来说，数据驱动的结果、预告和对趋势的预测都是必不可少的。今天，在我们所做的每件事中，都能看到某种分析的逻辑在背后。从点击网站（点击流分析）、在线购买（客户行为）、遗传学、CRM、公用事业、医疗保健，甚至选举，我们都可以看到分析的存在。分析的能力不再让你获得优势，它已经变成了你保持业务不被淘汰的必要条件。它倒逼组织建立数据湖或升级现有的数据仓库。

这就引出了一个非常有趣但也令人困惑的问题：我应该用数据仓库还是数据湖？答案其实很简单。一般情况下，你应该同时拥有数据仓库和数据湖，更准确地说，数据仓库位于数据湖中。

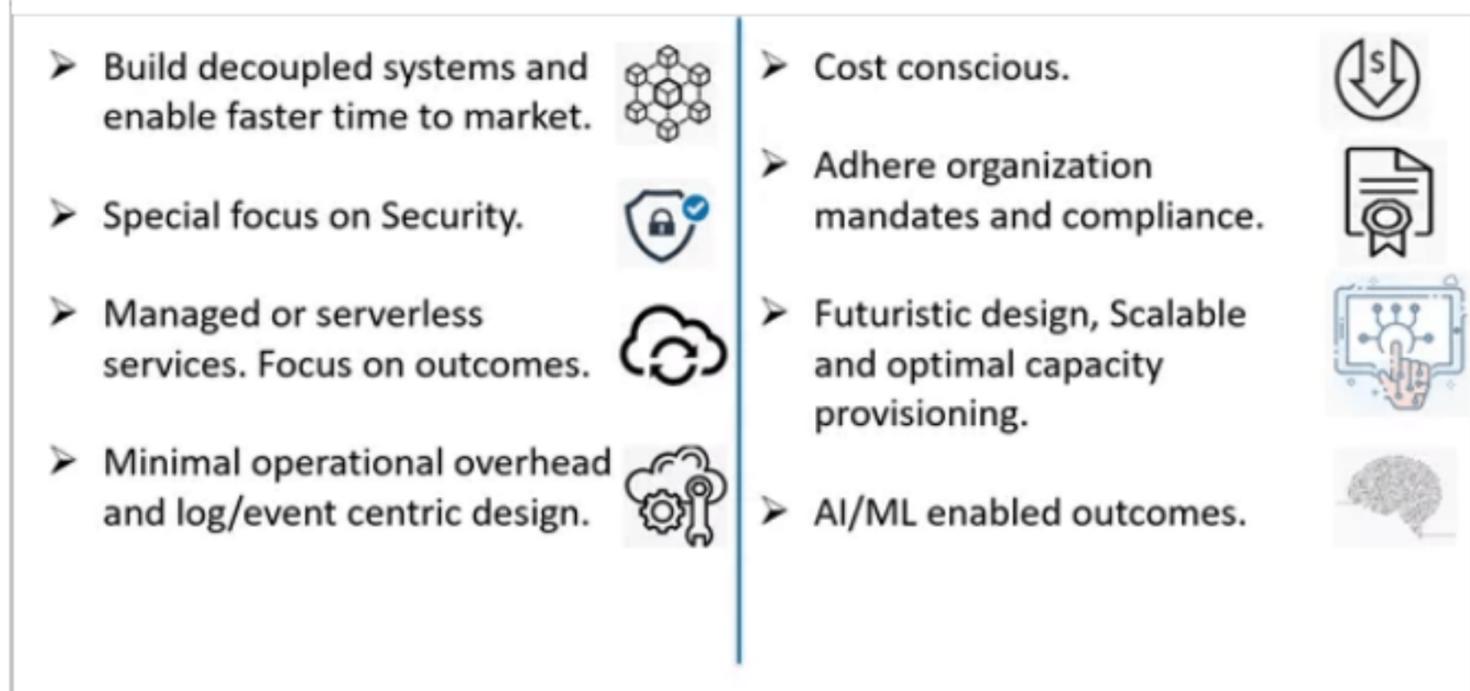
数据仓库 vs 数据湖

数据仓库是为分析来自不同系统或业务线的结构化数据而优化的数据库。为了支持更快的 SQL 驱动操作报告和分析，模式和数据结构都已经预先定义了。数据仓库中的数据已经被清理、丰富和转换为“单一的真理来源”。

然而，数据湖存储来自商业应用程序、移动应用程序、物联网设备和社交媒体的结构化和非结构化数据。模式在数据刚捕获阶段是

不需要提前定义的。这意味着你可以存储数据，而不需要仔细设计，也不需要知道要获得什么样的见解。它支持大数据分析、搜索分析、机器学习、实时分析、日志分析和点击流分析等。

理论上，数据湖听起来像是所有问题的一站式解决方案，但并不令人惊讶的是，很多数据湖都失败了。数据湖解决了两个主要问题：“消除数据竖井”和“存储异类源”。然而，这也带来了许多挑战，需要正确的体系结构、存储、数据治理和安全模型来驱动业务结果。



数据湖的特性

对数据湖的预期

数据湖应该能够交付：

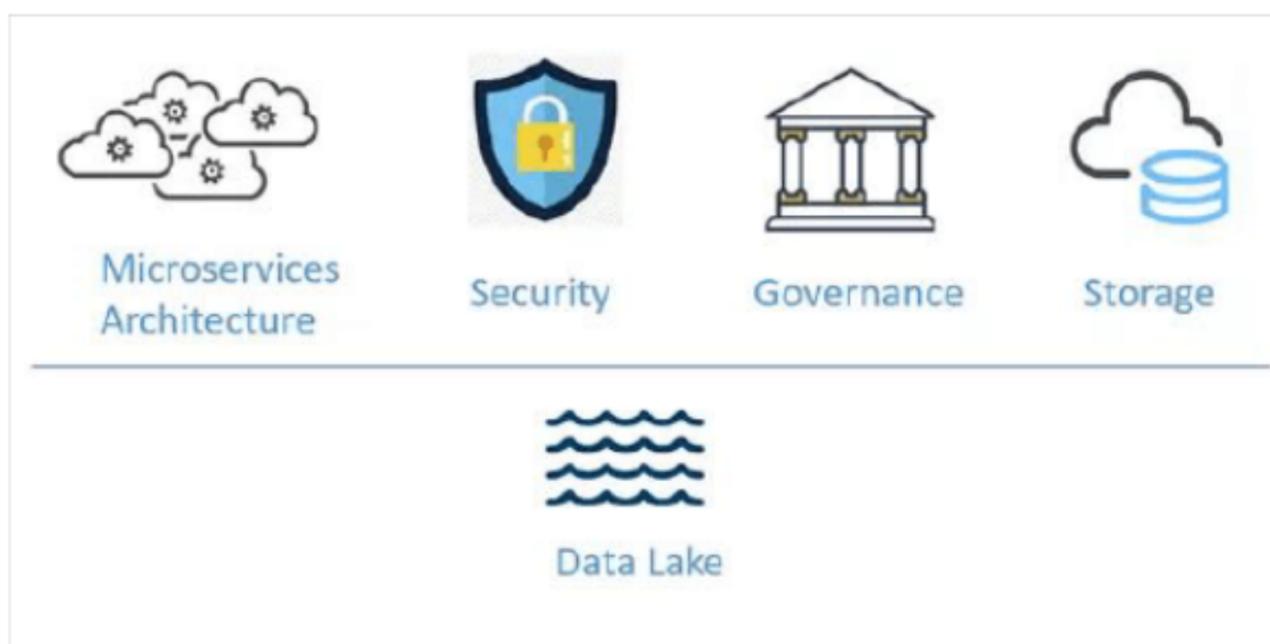
1. 不同的数据源：数据湖应该支持从任何数据源高效高速收集数据，来自不同来源的数据有助于执行完整和深入的分析；

2. 数据可访问性： 它应该允许组织 / 部门范围内的数据以一种安全的授权方式从多个来源访问数据，数据专业人员和企业不受 IT 部门的官僚主义影响；

3. 及时性： 数据很重要，但只有在及时收到数据的情况下才重要。所有用户都有一个有效的时间窗口，在此期间，正确的信息可以影响他们的决策；

4. 自助式服务： 对于组织范围的数据，数据湖应该允许用户使用所需的工具集构建他们的报告和模型。

我们接下来要讨论怎样设计更好的数据湖， 下图提到的微服务架构、安全、治理、和存储是构建有效的、数据驱动的、未来感十足的数据湖的四大支柱。



数据湖的四大支柱

架构



大多数现代数据湖都是使用微服务体系结构构建的，其核心是构建一套专注于业务功能并可独立部署的小型服务。微服务体系结构是构建解耦的、敏捷的和自动化的数据湖应用程序的理想选择。一个理想的架构应该有：

解耦应用程序： 所有进程都应该解耦，以避免出现故障时无法应对。例如，处理一组数据管道的失败不应阻止处理其余的数据管道。

消除单点故障： 单点故障可能导致整个系统崩溃。而多点故障可以确保在工程师解决故障时，其他的数据管道可以不受影响。这也有助于防止类似 DDoS 的攻击。这种方法应该同时用于硬件和应用程序。

敏捷： 在小的 sprint 中与业务合作交付最可行的产品 (MVP)。业务和 IT 作为合作伙伴在 sprint 中添加特性。这确保了没有意外，而且有效的微服务模型允许并发应用程序部署。

计算与存储解耦合： 允许存储和计算资源的独立扩展，可以垂直地(向同一台机器添加容量)和水平地(添加更多的机器)，以接近最佳配置。

审核和日志记录： 由于有如此多的应用程序和进程以一种解耦的模式运行，因此记录事件以分类问题和流变得非常重要。从数据治理的角度来看，记录各种 API 和事件的审计记录变得非常重要。如果存在任何破坏或未经授权的访问，这对于理解服务是如何被使用非常方

便。几乎所有的云平台都提供审计服务，需要启用这些服务来存储日志。始终采取预防措施，使审计日志不可变，不受篡改。

存储

存储是现代数据湖的核心。数据湖服务于具有不同背景和工具偏好的不同客户，比如数据科学家、分析师和业务用户，他们都需要一组不同的工具和对数据的访问。集中存储有助于更好的治理、维护和使用多种工具的能力。通过集中呈现数据，所有应用程序和工具都可以轻松读写数据。这种方法提供了替换工具的灵活性，因为存储已经集中并解耦了。然而，拥有以下特征是很重要的：



存储特性

可伸缩性：企业数据湖充当整个组织或部门数据的集中式数据存储。它必须具备扩展性，没有容量的限制。像 AWS S3 或 Azure 存储这样的服务有助于实现这一点。

高可用性：数据读取的及时性和不间断可用性是决策的关键。跨多可用性区域的复制有助于实现高数据可用性。多区域数据复制确保有效的灾难恢复。对于用户跨多个区域工作的业务，跨不同区域复制数据有助于更快地为用户提供服务，因为数据距离用户或应用程序更近。

数据持久性：数据一旦存储，就不会因为磁盘、设备、灾难或任何其他原因而丢失。核心存储层具有非常高的持久性，可以实现出色的数据健壮性

安全性：无论是云计算还是本地计算，数据安全性都是最重要的考虑因素。数据必须是加密的、耐篡改的、不可变的（在需要的地方），并且符合要求的规则。数据丢失就是业务丢失。

治理和审计：应用治理规则、数据不变性、识别 PII 数据以及提供数据使用的完整审计日志的能力对于满足法规和法定要求至关重要。

存储任何内容：数据湖的主要设计考虑事项之一应该是存储任何格式的数据（结构化和非结构化），并提供快速的检索时间。对于此类使用场景，强烈建议使用对象存储。

存储文件的大小和格式：一个小文件有大小小于 Hadoop 文件系统(HDFS)默认块大小为 128 MB 的。在 Hadoop 框架,集群中的每个文件被表示为一个对象的名字节点的内存,每个占 150 个字节。这意味着大量文件将大量消耗内存。大多数基于 Hadoop 的框架在使

用小文件时效率不高。另一个重要的方面是文件的格式（行存储 vs 列存储）。通过柱状文件格式，可以只读取、解压和处理当前查询所需的值。流行的文件格式是 ORC 和 Parquet，它们都有自己的用例和优点。

汇聚

数据汇聚是将数据从不同来源（如点击流、数据中心日志、传感器、物联网设备、API 和数据库）获取的过程。根据源的类型以及是否需要实时处理数据，可以实时或批量地获取数据。在数据湖中，数据以原始格式（结构化或非结构化）引入。可以使用流行的数据复制工具、流工具或 ETL 工具摄取数据。数据摄入的主要目的是快速有效地以原始格式获取数据。在这个阶段没有应用转换，如果有了可用的原始数据，我们可以回到需要的时间点。建议有效地组织数据存储，以实现更快的数据访问。组织结构的例子——主题区域 / 数据源 / 对象 / 年 / 月 / 日被广泛使用。

数据处理

这涉及到构建数据管道并处理数据。理想情况下，第一步应该是创建一个数据目录（我们将在数据治理部分详细讨论它）。通常，在数据处理过程中应该生成多层独立的数据，如标准化、清理和特定于应

用程序的转换数据，用于不同的目的，如机器学习、数据仓库或分析。这个阶段还包括为处理数据选择正确的框架。

大数据框架：要在高速下处理大量数据，分布式框架是首选。分布式框架意味着数据集被划分为多个文件（默认为 128 MB），然后在多台机器上并行处理，然后合并数据。分布式使更短的时间内处理大型数据集成为可能。有各种各样的框架，比如 Apache Hadoop、Apache Spark，还有一些商业可用的云框架。最流行的框架之一是 Spark 2.0，它是高度内存密集型的，并提供了各种选项，如处理时间序列数据、图形数据和 Spark SQL 来简化编码。AWS 提供了 AWS EMR，这是一个托管服务，并提供了许多预先安装的工具，可以选择您所选择的框架。

ETL 工具：像 Informatica PowerCenter、Talend、Microsoft SQL server SSIS 和 Matillion 这样的 ETL 工具非常适合运行 ETL 数据管道。它还提供了数据编目选项。

Features

可伸缩性：理想的数据处理框架应该允许在任何时间点进行垂直伸缩（在同一台机器上增加计算能力）和水平伸缩（并行地增加更多机器），并根据数据负载需求实现从零到最小的停机时间。自动分组是一种基于 CPU 或 IOPS 等重要参数自动增加计算能力的好方法。

永久集群与临时集群：一些业务需要 24*7 运行的集群，这意味着资源一直在被积极使用。然而，有些业务需要每天或每周花几个小

时处理数据。在这种情况下，不间断运行集群并产生成本是没有意义的。在 Hadoop 集群中，数据存储存储在节点上，这使得在不丢失数据的情况下终止集群非常困难。然而，像 AWS EMR 这样的服务允许将数据存储到 AWS S3。这允许轻松地终止 EMR 集群，并在需要时重新启动集群。这是非常划算的。

托管集群：管理 Hadoop 集群相当麻烦。它需要大量的投资和维护，而且相当昂贵。现在，AWS、Azure 和谷歌等供应商都在提供托管集群，能够快速终止和创建集群。这允许企业将精力集中在数据结果上，而不是支持服务器。

消费

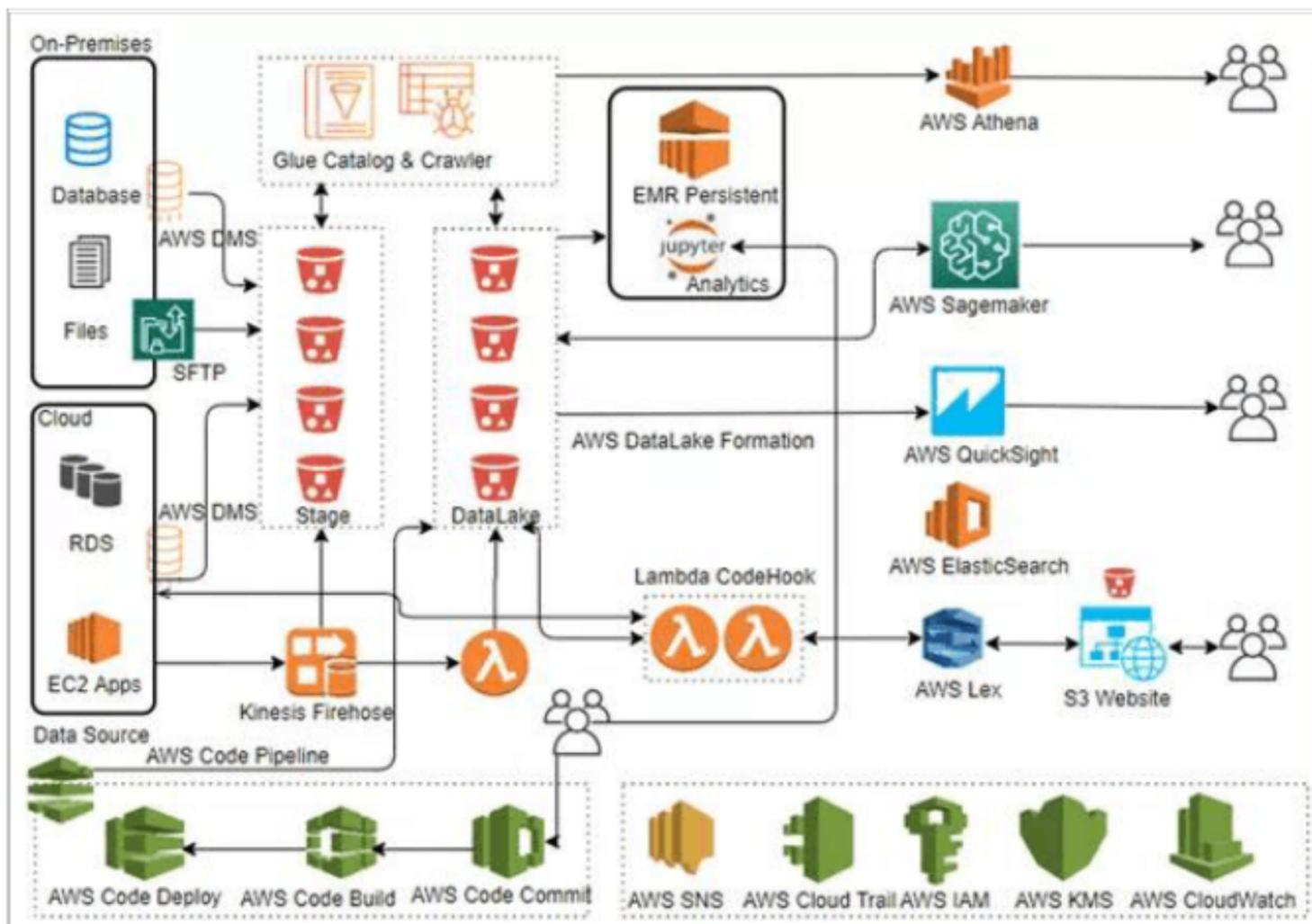
数据湖的主要推动力之一是允许拥有不同技能集的客户使用不同首选工具的数据。数据编目、不可变的原始数据、集中存储、多层数据处理和只读模式的多种工具使用。我们可以把消费者分成以下几类：

数据仓库：业务用户需要高性能的数据仓库来运行 pb 级数据上的复杂 SQL 查询，以返回复杂的分析输出。一些设计因素使得工具能够提供快速的结果。AWS Redshift spectrum、谷歌 BigQuery 和 Azure SQL Data warehouse 等工具提供了巨大的压缩、区域映射、柱状存储，以及在存储文件上高性能运行复杂查询的能力。此外，云平台还提供了可用性、持久性、安全性和成本效益。

交互式查询： 对于一些用例，数据分析师需要运行 SQL 查询来分析大量的数据湖数据。 Apache hive 、 Apache Presto 、 Amazon Athena 和 Impala 等工具使用数据目录构建 SQL 友好的逻辑模式，以查询存储在选定格式文件中的底层数据。 这允许在数据文件上直接查询结构化和非结构化数据。

机器学习： 数据科学家经常需要针对一个巨大的数据集运行机器学习算法来进行预测和预测。 数据湖提供了对企业范围数据的访问，以探索和挖掘数据以获得业务洞察力。 数据科学家可以使用 Dataiku 、 Tensorflow 和 Sagemaker 等工具，或者在 AWS 等平台上运行用 R 或 Python 编写的算法， AWS 提供了在集中式数据存储上使用经济实惠的 spot EC2 实例按需旋转 EMR spark 集群的能力。

人工智能和自动化： 像在数据湖之上使用 Alexa 的聊天机器人和语音分析这样的解决方案， 或者数据湖托管解决方案显著改善了客户体验， 减少了运营开销。 这些解决方案可以与安全网站或移动应用程序集成。



构建在 AWS S3 上的数据湖架构

数据治理

在数据湖中，从多个来源收集组织范围的数据，包括消费者个人识别信息 (PII) 数据。该数据包含分析员可以用来识别和改进业务产品的重要信息。然而，这些敏感数据必须受到保护，符合隐私法律法规。这使得数据治理成为设计数据湖的关键支柱。数据治理是指对企业中数据的可用性、可用性、完整性和安全性的全面管理。它主要取决于业务策略和技术实践。治理应该从一开始就作为设计的一部分合并，或者至少从一开始就应该合并最低标准。数据治理主要包括以下方面：

元数据管理： 由于数据湖中存储了大量数据，因此很难跟踪哪些数据已经可用，并可能导致数据溢出。 对此的一个解决方案是数据目录。数据目录是与数据管理和搜索工具相结合的元数据的集合， 这些工具可以帮助分析人员和其他用户找到他们需要的数据。 数据目录作为可用数据的目录，并提供用于评估健身数据的预期用途的信息。最有效的方法是维护一个中央数据目录，并跨各种处理框架（如 Apache Hadoop 、 Apache Spark 、 AWS Athena 和各种其他可用工具)使用它。这确保了元数据的完整性，并应用了简单的数据治理规则。

数据质量： 数据质量与数据完整性、准确性、一致性、数据屏蔽和标准化有关。在使数据可用之前，它确保应用了所有这些属性并正确地分类了数据。

遵从性和规则： 必须根据所操作的业务领域实现几个遵从性要求。例如 GDPR, HIPAA 和 ISO 标准。如果不遵守，可能会被处以巨额罚款，甚至采取更严厉的行动。它也削弱了信任和商业信誉。有一些产品和服务可以帮助实现这一点，比如 AWS Macie 帮助识别 PII 信息，AWS HSM 提供完全控制和安全的密钥管理服务（KMS）。

安全

对于本地和基于云的企业数据湖，安全性都是至关重要的，应该是最优先考虑的问题。安全性应该从一开始就进行设计，并且需要在

非常基本的架构和设计中进行整合。此外，只有在企业的整体安全基础设施和控制框架中部署和管理数据湖的安全性，才能成功。安全可分为以下几类：

数据安全： 静态和传输中的加密：组织的数据是一种需要保护的资产，不被窥探。几乎所有数据都必须在静止状态下（存储在文件和数据库中）得到保护。默认情况下，所有云提供商都为其存储层提供加密机制。此外，可以通过选择加密算法以及由谁（云提供商或客户）管理和旋转密钥，使用密钥管理服务来实现加密。对于非常安全的系统或由于监管需要，组织希望管理其机器上的密钥，可以使用硬件安全模块(HSM)。传输中的数据意味着数据在网络上在设备和服务（如 API）之间移动。这可以通过使用带有证书的 TLS/SSL 传输来实现。

网络安全： 下一个重要方面是网络安全。对于云解决方案，虚拟私有云 (VPC) 提供了云中的网络隔离。它提供了使用安全组和使用传统方法 (如网络 ACL 和 CIDR 块限制) 限制连接的灵活性。VPC 端点的使用允许流量通过私有网络而不是公共网络传输。所有这些策略都创建了一个网络非军事区 (DMZ)。另一个方面是网络防火墙，它控制访问并监视网络上的 web 流量。它还授权出站会话。它与 OSI 层中的网络层属性位于一起，所以它只在网络层上提供访问控制。

访问控制： 企业数据湖包含组织范围的数据，因此确保正确的身份验证策略变得非常重要。每个组织都使用常用的技术来维护标准认证，比如 active directory，它可以用于为数据湖生态系统产品提供

认证。本地和云平台都支持将企业身份识别基础设施映射到云提供商的许可基础设施上的方法。此外，可以使用身份访问管理 (IAM) 控制细粒度访问。像 AWS 这样的云平台使用 AWS IAM 和 bucket 策略来访问数据文件，提供精细的访问管理。这确保只有正确的用户集可以访问所需的资源。

应用程序安全性：保护应用程序免受外部攻击是至关重要的。一般来说，网络防火墙没有检测 / 防止威胁的机制。为此，我们应该使用 web 应用程序防火墙来帮助保护您的 web 应用程序或 api，防止常见的 web 攻击影响可用性、危害安全性或消耗过多的资源。应用程序防火墙允许您创建阻止常见攻击模式 (如 SQL 注入或跨站点脚本编写) 的安全规则，以及过滤您定义的特定流量模式的规则，从而控制流量如何到达应用程序。另一种方法是实现微服务体系结构，比如将应用程序与存储或其他应用程序解耦，以减少表面的攻击。如果使用云，设计应该包含自动提供组，它可以自动添加资源来吸收高流量攻击，比如分布式拒绝服务 (DDoS)。