实用拜占庭容错

中文版

第三届操作系统设计与实现研讨会上首次呈现，美国，新奥尔良，1999年2月

Practical Byzantine Fault Tolerance

Miguel Castro & Barbara Liskov

麻省理工学院

计算机科技实验室

马萨诸塞州剑桥科技广场545号

{castro,liskov}@lcs.mit.edu

摘要

本文提出了一种新的能够容忍拜占庭错误的复制算法。我们相信拜占庭容错算法在未来将变得越来越重要，因为恶意攻击和软件错误越来越普遍，并可能导致故障节点表现出任意行为。以前的算法假设是异步系统，又或者其速度太慢，导致算法不实用，但本文所描述的算法是实用的：它工作在像互联网这样的同步环境中，并且包含了一些重要的优化，这些优化可以将以前算法的响应时间提高一个数量级以上。使用我们的算法实现了一个拜占庭容错的NFS服务，并对其性能进行了测试。结果表明，我们的服务仅比标准的无副本的NFS慢3%。

介绍

恶意攻击和软件错误越来越多。工业和政府越来越依赖在线信息服务，这使得恶意攻击更具诱惑力，并使得成功攻击的后果更加严重。此外，由于软件规模和复杂性的增长，软件错误也在增加。由于恶意攻击和软件错误会导致故障节点表现出拜占庭（即任意）行为，拜占庭容错算法变得越来越重要。

本文提出了一种新的、实用的、拜占庭容错的状态机复制算法[17，34]。该算法保证了liveness和safety，可以容忍副本总数中最多1/3的节点同时出现故障。这意味着客户端最终会收到对其请求的回复，并且根据线性化能力[14，4]，这些回复是正确的。该算法适用于像Internet这样的异步系统，结合一些重要的优化，其能够很高效地运行。

对于拜占庭容错协议和复制技术，已经有了许多研究（从[19]开始）。然而，大多数早期的研究（例如，[3，24，10]）要么关注在证明理论上的可行性，这些在实践中使用起来效率太低，要么假设同步系统，即依赖于消息延迟和处理速度的已知界限。最接近我们的系统是Rampart[30]和Securering[16]，其设计是实用的，但它们都是基于同步假设来保证其正确性，这在恶意攻击面前是危险的。Anatacker可能通过延迟非故障节点或它们之间的通信直到它们被标记为故障并排除在副本组之外而危及服务的安全。这种拒绝服务攻击通常比控制无故障的节点更简单。

我们的算法不易受到这种攻击，因为它不依赖同步性来保证安全性。此外，如第7节所述，它的性能比Rapart和SecureRing高出一个数量级。它只使用一个消息往返来执行只读操作，使用两个消息往返来执行读写操作。此外，它在正常操作期间使用基于消息身份验证码的高效身份验证方案；公共密钥加密被称为Rampart中的主要延迟[29]和吞吐量[22]瓶颈，仅在出现故障时使用。

为评估我们的算法，我们实现了一个复制库并用它实现了一个真正的服务：一个拜占庭容错的支持NFS协议的分布式系文件系统。我们使用Andrew基准[15]来评估系统的性能。结果表明，在正常情况下，我们的系统仅比Unix内核中的标准NFS服务慢3%。

因此, 本文作了如下的贡献：

它描述了第一个可以在异步系统中容忍拜占庭错误的状态机复制协议。

它描述了一些重要的优化，这些优化使算法能够更好的执行，从而可以在实际系统中被应用。

它描述了拜占庭容错的分布式文件系统的实现。

它提供了量化复制技术成本的研究实验。

论文的其余部分安排如下：

我们首先描述我们的系统模型，包括我们的失败假设。第3节描述了该算法解决的问题并说明了正确性条件。第4节描述了算法，第5节描述了一些重要的优化。第6节描述了我们的复制库，以及我们如何使用它来实现拜占庭容错的NFS。第7节介绍了我们研究实验。第8节讨论相关工作。最后，我们对所取得的成就进行了总结，并对未来的研究方向进行了讨论。

2 系统模型

我们假设一个异步分布式系统，网络可能消息丢失、消息延迟、消息重复或者消息无序。其中节点通过网络连接。

我们使用拜占庭故障模型，即故障节点可能会出现任意行为，仅受下面提到的限制。我们假设独立节点失败。为了这个假设在恶意攻击存在的情况下是正确的，则需要采取一些步骤，例如，每个节点应该运行服务代码和操作系统的不同实现，并且应该具有不同的root密码和不同的管理员。从同一个代码库[28]获得不同的实现是可能的，对于低复制级别，可以从不同的供应商购买操作系统。N-version programming，即不同的编程团队产生不同的实现，是某些服务的另一种选择。

我们使用加密技术防止欺骗和重放，并检测损坏的消息。我们的消息包含公钥签名[33]、消息身份验证码[36]和由抗冲突哈希函数生成的消息摘要[32]。我们将节点i签名的消息m表示为<m>(i)，将消息m的摘要表示为D(m)。我们遵循通常的做法，对消息摘要进行签名，并将其附加到消息的明文中，而不是对完整消息进行签名（应该这样解释）。所有副本都知道其他节点的公钥从而可以验证签名。

我们允许一个非常强大的对手来协调故障节点、延迟通信或延迟正确的节点，以便对复制服务造成最大的损害。我们假设对手不能无限期地延迟正确的节点。我们还假设对手（以及它控制的错误节点）在计算上是受限制的，因此（以非常高的概率）它无法颠覆上述加密技术。例如，对手无法生成非故障节点的有效签名，无法从摘要计算摘要汇总的信息，也无法找到具有相同摘要的两条消息。我们使用的密码技术被认为具有这些特性[33，36，32]。

3服务属性

我们的算法可以用来实现任何具有状态和某些操作的确定性复制服务。这些操作不限于对服务状态部分的简单读写；它们可以使用状态和操作参数执行任意确定的计算。客户端向复制的服务发出请求以调用操作并阻止等待答复。复制服务由n个副本实现。如果客户端和副本遵循第4节中的算法，并且没有攻击者可以伪造它们的签名，则它们是无故障的。

该算法在假设不超过[（n-1）/3]个副本出现错误的情况下，提供了安全性和活性。安全性意味着复制的服务满足线性化能力[14]（修改为解释拜占庭错误客户机[4]）：它的行为类似于一个集中的实现，一次原子地执行一个操作。

安全性要求错误副本数量的限制，因为错误副本可以任意行为，例如，它可以破坏其国家安全无论有多少faultyclient在使用服务（即使它们与错误的副本串通），都会提供：由错误的客户端执行的所有操作都由非错误的客户端以一致的方式观察客户。特别是，如果服务操作是为了在服务状态上保留一些不变量而设计的，那么故障客户机就不能破坏这些不变量。

安全属性不足以防范故障客户端，例如，在文件系统中，故障客户端可以将垃圾数据写入某个共享文件。但是，我们通过提供访问控制来限制有故障的客户机可能造成的损害：我们对客户机进行身份验证，如果发出请求的客户机无权调用操作，则拒绝访问。此外，服务可以提供更改客户端访问权限的操作。由于该算法确保所有客户端一致地观察到访问撤销操作的效果，因此它提供了一种强大的机制来从故障客户端的攻击中恢复。