ch12 分布式系统

概述

分布式系统的组成 P6

如今所说的分布式系统通常是由分布在不同地理位置的系统组件所构成，这些系统组件通过网络传递消息完成动作协调，从而相互协作完成共同的任务

- 交互网络 — 实现协作的前提，系统组件基于交互网络实现消息交互
- 分布式算法 — 协调分布式组建之间的交互，确保系统稳定运行
- 信任机制 — 解决不同系统组件之间的信任问题，实现可信协作

三个属性 P7~8

- 一致性 : 分布式系统中各正确节点的所保存的同一变量状态信息都是一致的
- 可用性 : 分布式系统在收到客户端的请 求后，必须给出回应，不能让客户端陷入无限等待过程中
- 分区容错性 : 分布式系统容忍其中节点 出现分区，当分区出现时，一个区域中节点发往另一个区域中节点的数据包全 部丢失，即区域间无法进行通信

在一个分布式系统中，一致性、 可用性、分区容错性，三者不可得兼

P（分区容错率）总需要满足，而C（一致性）和A（可用性）矛盾。

安全问题 P9

安全问题可能会造成系统分区，或是危害系统的一致性、有效性和隐私性

从分布式系统的组成来看，其安全问题的根源体现在交互网络、分布式算法和信任三个方面 :

- 攻击者可以攻击交互网络，监听交互过程中的隐私信息或造成网络分区
- 协作过程中可能会出现冲突，故障等问题，从而影响一致性和可用性
- 系统内部可能存在恶意组件，系统和客户端交互时也存在信任问题

交互网络 P12

交互网络的组成 P12

在分布式系统中，节点之间可以通过交互信道实现消息传递；分布式节点选择邻居节点并建立交互信道，从而构成一个交互网络

交互信道：两个主机通过交互信道实现host-to-host的通信，常见的交互信道有UDP、TCP和TLS等

邻居选择机制：每个节点依靠邻居选择机制选取系统中的部分或全部节点建立交互信道，从而建立邻居关系，节点可通过邻居节点与其它非邻居节点交互

交互网络存在的安全隐患 P13~16

根据网络体系结构的不同层面进行分类：网络层面，应用层面。

根据攻击者的位置分类：中间转发节点，其它节点，数据包丢失和重复，信息泄漏，数据包篡改，数据包伪造，重放攻击

TCP/TLS 交互信道 P17~18

TCP协议采用滑动窗口协议确保数据包有序提交，并解决重复问题；采用超时重传机制解决链路拥塞导致的数据包丢失问题。当未按序接收时，等待前面的数据包完成接收后再将数据包序列提交到上层应用。当长时间未收到数据包时，触发超时空重传机制，请求源端超时重传数据包。

TLS 协议建立在TCP协议之上运行，提供以下三个安全功能：

1. 数据完整性：验证数据是否伪造而来或未遭篡改过
2. 身份认证：获取通信方的身份，为访问控制机制作支撑
3. 数据加密：协商会话密钥用于数据加密，防止第三方窃听传输数据

防止重放攻击 P19

1. TLS 防止重放攻击 P19：TLS交互信道带来的优势——基于TCP协议确保传输层面数据包不会出现重复；通过数据加密确保恶意节点无法通过监听获取应用层消息。以下消息依然可能遭受重放攻击：客户端发起的TLS请求建立信息ClientHello；TLS1.3为了加速握手过程而引入的0-RTT信息。
2. **基于时间戳防御重放攻击** P20：优点：容易实现；缺点：要求交互双方进行精确的时钟同步；存在一个可以被攻击的时间窗口。
3. **基于随机数防御重放攻击** P21：优点：无需时钟同步；不存在可被攻击的时间窗口；缺点：复用之前的数时，可能遭受重放攻击；需要记录大量已用过的随机数。
4. 时间戳和随机数相结合（书 P409），只需要在 δ 时间内随机数不重复即可 书page409

链路故障 P23~26

链路崩溃，性能问题，链路故障特征 P23

造成网络故障的主要原因：Internet的不稳定性，分布式系统所依赖的底层网络随时都可能出现物理链路故障、路由故障等因素。

P26 覆盖网络

弹性覆盖网络RON P27

弹性覆盖网络是一种分布式覆盖网络体系结构，它可以使分布式应用检测到路径的失效和周期性的性能降低现象并能够迅速恢复；恢复时间少于20秒；对比：在目前的Internet中，BGP恢复时间为几分钟。

RON节点监控它们之间的路径的质量并根据这些信息决定：直接利用Internet转发分组或者通过其他RON节点转发分组。

RON 的目标 P28，设计 P29~30

P2P vs C/S p33~36

二者在结构和构成上有很大区别：管理能力、构态能力、功能（查找或发现）、组织、元素（DNS）和协议。但又无明显边界，都能运行在不同的（Internet / Intranet）平台上。都能服务传统或新的应用：eBusiness eServices ...

P2P (Peer-to-Peer)又称对等网络技术，也是建立在Internet之上的分布式Overlay网络，能够不依赖中心节点而只依靠对等协作实现资源发现与共享

P2P 网络建立流程 P34~36 1 网络节点发现 2 邻居节点维护

日蚀攻击 P37~40

攻击目标：针对网络节点发现以及邻居节点维护两个过程进行攻击，使得受害节点的所有邻居节点都是由攻击者所控制的恶意节点。危害：攻击者可以选择性的转发对攻击者有利的消息给受害者，从而配合其它网络攻击；攻击者可以完全隔离受害者与网络中其它节点的信息交互。

挤占入向连接 P38，**挤占出向连接** P39 **配合路由劫持攻击** P40——主要思路 解决思路

分布式算法 P42

协作过程中的安全隐患 P42

分布式算法的目的是协调分布式组件之间的交互和动作，从而确保它们能够稳定协作实现一个共同的任务。缺乏全局时钟；并发；故障。

时钟同步算法 P43~46

时钟不同步引发的问题 P43。时钟同步算法 P44~46。

基本思想：若节点A想与节点B完成时钟同步，则请求节点B发送本地时间T；然后预测A与B的网络时延RTT；之后将时间设置为T+RTT/2；RTT预测的越精确，时钟同步的误差越小。

P47 网络时间协议的主要目标和整体架构：NTP协议定义了时间服务的体系结构，以及如何在互联网上发布时间信息。

并发控制 P48~71

P48~49：并发问题破坏一致性或者导致指令之间相互影响。

P50 事务与分布式事务：事务是原子的一组请求，这组请求不受其它客户端干扰，并且要么全部完成，要么不对服务器造成任何影响。四个特性：1 原子性 2 一致性 3 隔离性 4 持久性

P51 **两阶段提交确保一致性** 在提交之前先询问所有存储该值的数据库是否可以更新，保证了**一致性**

P52 **基于锁机制确保隔离性** 在对 A 修改之前先上锁，在 Commit 或者 Decommit 之后解除 A 的锁

P55~57 对于两阶段提交协议，会出现三种故障： 1 数据库节点故障 2 协调者故障 3 链路故障

无法容错的根源：在简单的二阶段提交协议中，必须所有相关节点都进行回应后才成功提交，因此协议无法容忍任何故障。

Quoroum 机制 P58 书 P 419

- 满足 一致性准则 和 有效性准则。写 W 个节点自后，认为写操作成功，在读取数据的时候，要读 R 个节点，由于 W+R > N，所以一定能读到正确的数据

Paxos 算法 书 P421 P60

- 无leader ，每个节点提出自己的提案，模拟一群追求快速达成一致的立法者。缺陷在于，追求所有节点相对平等的关系，造成不必要的信息交互。甚至可能会形成**活锁**

Raft 算法 书 P423 P65

- 增加了领导者的角色，只有领导者能够提案。算法分为 **领导者选举** 和 **日志复制** 两个部分。只有领导者故障的时候进行领导者选举。当用户来了一个请求，给领导者，领导者试图和所有跟随者同步日志，如果同步成功，则提交事物，并给客户端返回，否则进行回滚。

信任问题 P73

零信任网络 P73~74：从来不信任，永远在校验。零信任对传统访问控制机制进行了范式上的颠覆其本质是以身份为基石的动态可信访问控制。

访问控制 P75

访问控制（Access Control）指系统按用户身份，以及该身份所在的策略组来限制用户对某些信息项的访问，或限制其对某些控制功能的使用。

基于角色的访问控制 P76

RBAC Role Based Access Control：不同角色有不同的操作集，RBAC设置不同的角色，并将各种身份信息与角色进行绑定，然后只对角色的访问权限进行控制，可极大地简化权限管理。

基于属性的访问控制 P77

ABAC Attribute：用户属性指的是可以对用户进行区分的特性。用户是访问控制的主体，以其属性作为访问网络数据的依据，可以实现更细粒度的访问控制。

包含了新的节点： 1 PEP 策略实施节点 2 PDP 策略决策节点 3 PIP 策略信息节点

基于信任度的访问控制 P78

分为基本信任，直接信任和推荐信任。 基本信任是由自身属性所决定，直接信任随着交互逐渐增加；推荐信任依赖信任值的传递。

访问控制的局限性 P79

访问控制只能解决被访问者对访问者的信任问题，但无法解决访问者对被访问者的信任问题，访问者可能访问到恶意、无效资源。

信任评价模型 P80~86

page81 包含三个角色，Trustee Trustor Recommender。系统举例 Eigentrust，节点根据历史交互评价计算信誉值。信任评价模型的五个要求 page83

拜占庭将军问题 P87

> 一组拜占庭将军分别各率领一支军队共同围困一座城市。为了简化问题，将各支军队的行动策略限定为进攻或撤离两种。因为部分军队进攻部分军队撤离可能会造成灾难性后果，因此各位将军必须通过投票来达成一致策略，即所有军队一起进攻或所有军队一起撤离。因为各位将军分处城市不同方向，他们只能通过信使互相联系。在投票过程中每位将军都将自己投票给进攻还是撤退的信息通过信使分别通知其他所有将军，这样一来每位将军根据自己的投票和其他所有将军送来的信息就可以知道共同的投票结果而决定行动策略。

系统的问题在于，可能将军中出现叛徒，他们不仅可能向较为糟糕的策略投票，还可能选择性地发送投票信息。假设有9位将军投票，其中1名叛徒。8名忠诚的将军中出现了4人投进攻，4人投撤离的情况。这时候叛徒可能故意给4名投进攻的将领送信表示投票进攻，而给4名投撤离的将领送信表示投撤离。这样一来在4名投进攻的将领看来，投票结果是5人投进攻，从而发起进攻；而在4名投撤离的将军看来则是5人投撤离。这样各支军队的一致协同就遭到了破坏。

由于将军之间需要通过信使通讯，叛变将军可能通过伪造信件来以其他将军的身份发送假投票。而即使在保证所有将军忠诚的情况下，也不能排除信使被敌人截杀，甚至被敌人间谍替换等情况。因此很难通过保证人员可靠性及通讯可靠性来解决问题。

假使那些忠诚（或是没有出错）的将军仍然能通过多数决定来决定他们的战略，便称达到了拜占庭容错。在此，票都会有一个默认值，若消息（票）没有被收到，则使用此默认值来投票。

一致性和正确性条件 P95

一致性条件不够充分，忠诚将军的信息有可能被修改，所以必须引入另一个正确性的条件。

一致性（等价IC1）：每个将军必须得到相同的(v1, v2, ..., vn)指令向量或指令集合；忠诚的将军不一定使用i将军发来的信息作为 vi，i将军有可能是叛徒。

正确性（等价IC2）：若i将军是忠诚的，其他忠诚的将军必须使用他发出的值作为vi。

简化问题的解决方案 P96

由于将军可能是叛徒，会对不同的副官发布不同的指令，因此副官不能完全信任将军的指令。解决思路：副官在收到将军的指令后，互相交换从将军处收到的指令值，再基于交互结果选择最终执行指令。

口头协议 P97

P199 **口头消息协议 OM 协议**：当存在 m 个叛徒 而且 $n \geq 3m+1$ 的时候，口头消息协议能够确保一致性，也即问题有解

签名消息协议 P111

签名消息协议 SM 协议：改进：在实际系统中，可以引入签名机制确保节点发送的消息无法被篡改，从而得到签名消息算法SM(Signed Messages)。

将军和副官发的消息都需要自己的签名，解决了口头消息协议交互复杂度过高的问题。结论，当 $n \geq 2m+1$ 的时候，只需要**两轮交互**，就能确保共识达成

实用拜占庭容错 P115

区块链：构造一个对客户端来说可信的分布式账本；容忍少数拜占庭节点作恶解决分布式系统不同节点间协作时的信任问题。

同步异步 BFT 共识的 Quorum 设计 P117~119

同步BFT共识依靠同步网络假设确保共识安全性和活性，若网络延迟超过预设最大值，则会造成节点间出现状态不一致。

异步BFT共识只能采用无Leader的共识模式，因为异步网络中无法分辨Leader错误和网络延迟过高两种情况；因此，异步BFT共识不仅效率低下，还可能导致理论上的活锁（类似于前面分析的Paxos共识）。

P120 引入部分同步网络模型

P122 PBFT 部分同步算法

PBFT 基于 leader 完成共识，基于 Quromou 机制，这里总结点 n = 3m+1 ； q=2m+1

PBFT 和 Paxous 和 Raft 区别是，增加了**共识之后的交易确认阶段**

Hotstuff 方案 page126-128