**本科毕业论文(设计)文献综述**

|  |  |  |  |
| --- | --- | --- | --- |
| **学 院** | 医学技术学院 | **专 业** | 计算机科学与技术专升本 |
| **姓 名** | 胡立哲 | **学 号** | 201812243102012 |
| **指导老师** | 李懿 | **职 称** | 讲师 |
| **合作老师** |  | **职 称** |  |
| **题 目** | **相册簿管理系统** | | |

**文献综述**（主要包括国内外现状、研究方向、进展情况、存在问题、参考文献等）

**网络并发与安全技术的现状与发展**

当我分析了这个课题后发现主要的难点有两个。一个是如何解决服务端的并发问题，当多个用户向服务端发起请求时，比如同步自己的相册簿，服务器该如何处理这些请求？；当多个用户同时在线浏览好友的相册簿时，服务器要快速地向不同的用户发送不同的大量的数据信息，才能让用户感觉不到高延迟，那么如何做到快速呢？还有一个是网络安全问题，一旦服务器被黑客攻击，内部的所有用户个人信息将泄露到不法分子手里，如何保护这些个人信息是一个关键问题。然而，对于我这样还没接触过网络编程的小白来说，还得一步一步从网络基础开始。

首先，我们的程序要能实现一台主机与另一台主机的通信。在进行软件编码之前，应当对应用程序有一个宽泛的体系结构计划。目前，现代网络应用程序中有两种主流体系结构：客户机-服务器体系结构（C/S）和对等体系结构（P2P）[[[1]](#footnote-2)]。C/S结构是最常用的，也是传统的结构。所以我们采用C/S体系结构来进行主机之间的通信。

在实现基本的通信基础上，我们就可以开始了解并发相关的问题。首先，我们要知道高并发的概念，高并发是指在同一个时间点，有很多用户同时的访问同一API接口或者url地址。它经常会发生在有大活跃用户量 ，用户高聚集的业务场景中。在互联网日益发展的今日，人们每天浏览着大大小小的网站，使用着各式各样的APP，无数的流量在无形中穿梭。那我们是如何判断一个网站或者APP的后台服务器端的技术强弱呢？每年大概有两个日子，是最能体现一个网站技术架构好坏的。一个是11月11日，传统的光棍节，同时也是我们马云爸爸为世界创造的一个购物节。这个节日是我们这一辈看着它发展的，从一开始的网站崩溃，到后来支持每秒上千万条订单的完成，这就是一个技术架构的成长。还有一个日子——每年春运开始抢票的日子。日复一日，年复一年，12306非常奇怪的是，无论花再多的钱，有再多的经验，他们的架构师似乎对这种可能发生大规模并发访问的问题完全没有一点概念，系统好像从来也没有经过任何高并发场景的性能评估和测试，就那么干脆利落地每年都发生着同样的事情——趴在那里长时间起不来。其实，如何打造一个高可用、高性能、易扩展、可伸缩且安全的网站？无论是非技术人员的互联网创业者，还是单纯开发的技术人员晋升之路，都是起着非常大的一个决定性作用。

在发展初期，一般都是小型网站，只需要一天服务器就绰绰有余，这时候的网站架构如下：应用程序，数据库，文件等所有的资源都在一台服务器上。。之后随着网站业务的发展，一台服务器逐渐不能满足需求；越来越多的用户访问导致性能越来越差，越来越多的数据导致存储空间不足。这时就需要把应用和数据分离，应用和数据分离后使整个网站使用三台服务器：应用服务器，文件服务器和数据库服务器，这三台服务器对硬件资源的要求各不相同，应用服务器需要处理大量的业务逻辑，因此需要更快更强大的CPU；数据库服务器需要快速磁盘检索和数据缓存，因此需要更快的硬盘和更大的内存；文件服务器需要储存大量用户上传的文件，因此需要更大的硬盘。应用和数据分离后，网站的并发处理能力和数据储存空间得到了很大的改善，支持网站业务进一步的发展。但是随着用户逐渐增多，网站又一次面临挑战：数据库压力太大导致访问延迟，进而影响整个网站的性能，用户体验受到影响。这时需要对网站架构进一步优化。鉴于网站访问也遵循二八定律，即80%的业务访问集中在20%的数据上，例如：新浪微博，只有经常登录的用户才会发微博，看微博，而这些用户对于总用户数只是冰山一角。既然出现这一现象，那么缓存这部分的数据是不是可以解决这现象呢？网站缓存可以分为本地缓存和分布式缓存这两种，二者的区别是本地缓存速度快但是受服务器内存限制缓存的数量有限，而分布式缓存采用的是集群处理，理论上是可以避免内存瓶颈。使用缓存后，数据访问压力得到有效缓解，但是单一应用服务器能够处理的请求连接有限，在网站访问高峰期，应用服务器成为整个网站的瓶颈。使用集群是网站解决高并发、海联数据问题的常用手段。当一台服务器的处理能力、储存空间不足时，不要企图去更换更强大的服务器，对大型网站而言，不管多么强大的服务器，都满足不了网站持续增长的业务需求。这种情况下，更恰当的做法是增加一台服务器分担缘由的服务器访问及储存压力。[[[2]](#footnote-3)]

服务器应用程序在接受来自多个远程客户端的套接字连接请求时，如果为每个连接都分配其各自的线程并且使用同步I/O，那么就会降低这类程序的开发难度。如果某个应用程序对套接字执行读操作而此时还没有数据到来，那么这个读操作将会一直阻塞直到数据到来。在单线程应用程序中，这不仅意味着在处理请求的过程中将停顿，而且还意味着在这个线程阻塞期间，对所有请求的处理都将停顿。为了避免这个问题，单线程服务器应用程序必须使用非阻塞I/O，这种I/O的复杂度要远远高于同步I/O，并且容易出错。然而，如果每个请求都拥有自己的处理线程，那么在处理某个请求时发生的阻塞将不会影响其他请求的处理。然而，操作线程也是有风险的，一是对开发人员的技术要求更高了，现在主流开发人员都必须了解线程方面的内容。二是安全性问题，由于多个线程要共享相同的内存地址空间，并且是并发运行，因此它们可能会访问和修改其他线程正在使用的变量。当然，这是一种极大的便利，因为这种方式比其他线程间通信机制更容易实现数据共享。但它同样也带来了巨大风险：线程会由于无法预料的数据变化而发生错误。[[[3]](#footnote-4)]

然后就是要了解关于网络安全这块的知识。起初，研究计算机系统和网络的人，被称为“hacker”也就是黑客，他们对计算机有着深刻的理解。在计算机安全领域，黑客是一群破坏规则，不喜欢被拘束的人，因此总想着能找到系统漏洞以获得一些规则之外的权力。在中国，黑客的发展阶段可以分为三个阶段：启蒙时代，黄金时代，黑暗时代。先来讲讲启蒙时代，大概在20世纪90年代，此时中国的互联网正处于起步阶段，一些热爱新兴技术的青年收到国外黑客技术的影响，开始研究安全漏洞，好奇心与求知欲是驱使他们前进的动力，没有任何利益的瓜葛。之后是黄金时代，这个时期以中美黑客大战为标志。自此事件后，各种黑客组织如雨后春笋般冒出。此时期的黑客圈子里贩卖漏洞，恶意软件的现象开始升温，同时因为黑客群体的良莠不齐，也开始出现以盈利为目的的攻击行为，黑色产业链逐渐成型。最后是黑暗时代，这个阶段一直延续到今天，也许还会继续下去。在这个时期黑客组织也遵循着社会发展规律，优胜劣汰，大多数的黑客没有坚持下来。在上一个时期非常流行的黑客技术论坛越来越缺乏人气，最终走向没落。所有门户型的漏洞披露站点，也不再公布任何漏洞相关的技术细节。[[[4]](#footnote-5)]

在计算机网络安全部分，计算机病毒的入侵是影响安全最为常见的一种问题，也是一种难以有效完全消除的问题。随着整个互联网的快速发展，目前计算机病毒的种类也越来越多，计算机病毒的种类甚至会随着计算机网络系统的不断维护而增多[[[5]](#footnote-6)]。在网络软件和操作系统层面上也存在着安全问题，当今整个计算机网络技术处于不断上升的趋势，其也对国家的经济发展做出了巨大贡献，同时在计算机网络以及软件的体量上不断增多，需要编写的程序也越来越多，难以做到完全无bug的情况[[[6]](#footnote-7)]。因此，针对这些问题人们做出了一些对策，比如防火墙技术是目前在计算机网络病毒控制上应用最为广泛的一项技术，通过合理地使用防火墙技术可以将非法用户进行隔离，将非法数据进行隔离，继而可以极大程度地减少因外部用户访问而导致的黑客入侵的情况[[[7]](#footnote-8)]。另外针对一些计算机网络安全问题中的计算机网络系统和软件一类的硬件部分安全问题，在安全防范措施上应当从提升这些硬件的安全防护管理水平的角度出发，不断提高网络工程的安全管理力度。首先，必须强化对计算机软件设备的防护管理工作，注重对计算机网络运行系统以及应用软件的安全审核工作，尽可能避免任何安全隐患因素的存在切实提高整个计算机网络操作系统和软件的安全性[[[8]](#footnote-9)]。其次，还必须加强对计算机硬件设备的日常维护检修的工作，能够在日常工作中对可能存在的安全漏洞进行检测和排查，尽一切可能将存在的安全隐患一一解决。针对于目前网络共享资源潜在危害而导致的计算机网络安全问题，应当由政府部门和行业协会直接通过彼此间的共同合作持续性地推动净网活动，实现对网络共享资源的优化，将藏匿在计算机网络共享资源中的安全隐患一一消灭[[[9]](#footnote-10)]。

为了防止用户信息被黑客盗取，我们需要对用户信息进行加密处理，为此我们务必要了解一些加密技术。传统的技术主要是对称加密或叫单钥加密，是20世纪70年代公钥密码产生之前唯一的加密类型。迄今为止，它仍是两种加密类型中使用最为广泛的加密类型。密码学有一些术语，如原始的消息叫明文，加密后的消息叫密文，从明文到密文的变换过程称为加密，从密文到明文的变换过程称为解密。对称密码有一个对称密码模型，它有5个基本组成部分：明文，加密算法，密钥，密文，解密算法。[[[10]](#footnote-11)]具体内容我将会在之后的章节中进行介绍。至于具体选用哪一种加密技术，目前还需要学习和了解才能做出合适的决定。

1. []James F.Kurose,keith W.Ross. 计算机网络:自顶向下方法[M]. 北京：机械工业出版社，2008.12 53-54 [↑](#footnote-ref-2)
2. []李智慧. 大型网站技术架构——核心原理与案例分析[M]. 北京：电子工业出版社，2013.9 [↑](#footnote-ref-3)
3. []Brian Goetz,Tim Peierls,Joshua Bloch等. java并发编程实战[M]. 北京：机械工业出版社，2012.2 [↑](#footnote-ref-4)
4. []吴翰清. 白帽子讲web安全[M]. 北京：电子工业出版社，2012.3 [↑](#footnote-ref-5)
5. []李慎之. 关于计算机网络安全防御技术分析[J].赤子，2019，10（7）：132 [↑](#footnote-ref-6)
6. []邹洋. 计算机网络安全技术在网络安全维护中的应用[J].山东工业技术，2019，2（4）：143 [↑](#footnote-ref-7)
7. []张葛. 计算机网络安全现状和防御技术分析[J].山东工业技术，2019，21（4）：141 [↑](#footnote-ref-8)
8. []陈谷涵. 计算机系统安全与计算机网络安全[J].中文信息，2018，2（12），2 [↑](#footnote-ref-9)
9. []刘燊. 计算机网络安全与对策[J].通讯世界，2019，26（1）：29-30 [↑](#footnote-ref-10)
10. []William Stallings. Cryptography and Network Security Principles and Pratice,Sixth Edition.[M] 北京：电子工业出版社，2015.3 [↑](#footnote-ref-11)