基于J2EE的移动电商平台安全性研究

1. 绪论
2. 移动电商平台综述
3. 移动电商平台的概念

传统电子商务的主界面是 PC，称为“有线的电子商务”；移动电商则是“无线的电子商务”，它是指用户通过移动终端如手机、掌上电脑等设备访问网络，进行各种电子商务活动。近几年，移动电子商务平台数量在逐渐的增多，买家和卖家通过移动电子商务平台进行的交易额也在逐年呈现暴涨趋势。移动电商平台是建立在移动互联网基础上电子商务平台，像美团，饿了么、淘宝、拼多多都属于典型的移动电商平台。

1. 移动电商平台的基本安全需求

保证交易过程中数据信息的安全传输是电子商务系统的关键,在移动商务中,需要为以下因素提供安全：

1. 保护交易对交易方和交易数据提供一定安全保护信息
2. 保护敏感和有价值的用户信息基础设施
3. 保护承载网络不被攻击。

基于此,移动电子商务系统的主要安全需求如下：

* 有效性

有效的防止系统延迟和拒绝服务情况的发生，保证交易数据在确定的时刻、确定的地点是有效的

* 机密性

防止合法或隐私数据为非法用户所获得，通常使用加密的手段实现，从而确保在交易过程中只有交易的双方才能唯一知道交易的内容。

* 完整性

确保交易事务的完整性，非交易第三方或非法攻击者不能对交易的内容进行修改

* 可靠性

提供完备的身份认证,确保交易双方是可以信任的,即确保服务间的相互身份认证,防止欺诈行为的产生

* 防抵赖性

系统具有防抵赖功能,确保交易双方不能对发生的正确交易行为抵赖

1. 基于JAVA的移动电商平台系统架构
2. JAVA的概述

Java是一门面向对象编程语言，不仅吸收了C++语言的各种优点，还摒弃了C++里难以理解的多继承、指针等概念，因此Java语言具有功能强大和简单易用两个特征。Java语言作为静态面向对象编程语言的代表，极好地实现了面向对象理论，允许程序员以优雅的思维方式进行复杂的编程。并且具有以下特点：

* 面相对象

把相关的数据和方法组织为一个整体来看待，从更高的层次来进行系统建模，更贴近事物的自然运行模式。

* 健壮性

吸收了C/C++语言的优点，但去掉了其影响程序健壮性的部份（如指针、内存的申请和释放等），提供了一个相对安全的内存管理和访问机制

* 跨平台

通过java语言编写的应用程序在不同的系统平台上都可以运行。只要在需要运行java应用程序的操作系统上，先安装一个java虚拟机（JVM）即可。由JVM来负责java程序在该系统中的运行。

1. 基于JAVA的移动电商平台系统架构

随着移动电子商务平台的不断发展，通过移动电子商务平台进行的交易额也在逐年呈现暴涨趋势。为了应对愈发暴涨的流量，网站的系统架构，从最初的单独部署应用服务，到现在大规模集群方式，技术的发展经过了一代又一代的革新。本文以阿里为参考，得出一个基于JAVA的移动电商平台系统架构如下：

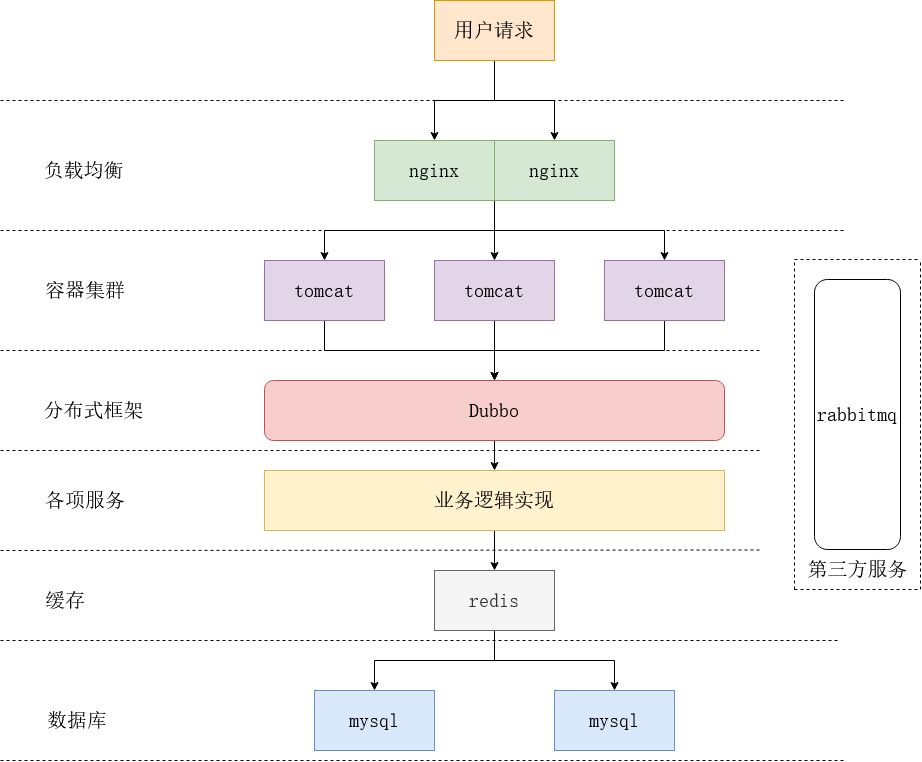


图1 基于JAVA的移动电商平台系统架构

从系统架构来看，一个基于JAVA的移动电商平台按照各组成部分可以分为以下几个部分:

* 负载均衡

负载均衡是指有效地将传入的网络流量分布到一组后端服务器上，也称为服务器池。移动电商平台必须为用户提供成百上千的并发请求，并以快速和可靠的方式返回正确的文本、图像、视频或应用程序数据。受限与现代科学技术，一台服务器并不能承载成百上千的并发请求，为了以成本有效的方式处理这些请求，现代计算最佳实践通常是添加更多的服务器来解决。

通过负载均衡，可以将用户请求分发到提供相同功能的不同服务器，降低服务器因并发量过大而导致容器奔溃的几率。也可以将CSS，JS，HTML，图片等静态资源从容器中分离出来，使得单个容器更易维护，运行效率更高。

现在使用最广泛的负载均衡工具为nginx。

* 容器集群

Servlet容器是指使用JAVA编写的，一个符合Servlet容器规范的程序，该程序封装了包含HTTP的网络协议规范，使得开发过程中，不用关注网络部分，从而能更好的专注于业务开发。Servlet容器的作用是负责处理客户请求，当客户请求来到时，容器获取请求，然后调用某个Servlet，并把服务的执行结果返回给客户。

容器集群，在一个移动电商平台中，单个容器并不能承载如此庞大的并发量，为了加强整体架构的刚性，承载更大的并发量，在一个移动电商平台中，通常采用多个容器，形成一种容器集群的模式。

* 分布式框架

在一个移动电商平台中，为了拆分业务，更好的使用容器，单个容器并不会拥有太多的功能，随着业务的发展，容器的增多，容器间相互通信与管理成为了一个很大的难题。分布式框架并不是一个程序，它是一段预先编写好的代码，封装了容器与容器之间相互通信的协议，提供了丰富的API，使得开发过程中，不用过分关注容器与容器之间的通信，每个开发人员可以只关注自己的部分业务。

一个分布式框架通常包括: 服务发现注册、配置中心、消息总线、负载均衡、断路器、数据监控等功能。

* 服务

这里的服务是值由开发人员编写的程序逻辑，一个移动电商平台常见的服务有：登录服务、购物车服务、交易服务、对账服务等。开发人员进行开发时，通常会分为三层：视图层、业务层与持久层。视图层负责页面展示与调整、业务层负责业务模块的逻辑应用设计、持久层负责数据持久化工作。

* 缓存

缓存就是将程序或系统经常要调用的对象存在内存中，一遍其使用时可以快速调用，不必再去查询数据库或者第三方服务。在一个移动电商平台中，往往存在着大量的热点数据，这些数据查询的频率很高，但是并不需要实时更新，如商品详情，商品浏览数，以及商品评论等。对这类数据采用缓存，能有效的降低数据库的访问频率，减少由数据库瓶颈对整个移动电商平台的影响。

* 数据库

数据库是一个长期存储在计算机内的、有组织的、有共享的、统一管理的数据集合。在一个移动电商平台中，单个数据库不会存储所有业务的数据，而是会根据业务进行划分，单个数据库只存储与单个容器相关的业务数据。如果需要表与表之间的关联查询，往往是通过业务逻辑来实现的。

* 第三方服务

第三方服务是指由第三方实现的程序，它封装了一系列的功能，只要按照程序的API给与入参，程序就能使用相关功能，节约了开发者开发相关功能的时间。目前在移动电商平台中用的比较多的第三方服务有：消息队列服务(rabbitmq),工作流服务(actoviti),搜索引擎(solr)。

1. 基于JAVA的移动电商平台的安全性机制
2. JAVA安全机制

java通过的沙箱安全模型保证了其安全性， java提供的沙箱组件有以下几种：

1. 类装载器结构；
2. 防止恶意代码去干涉善意的代码。

这是通过为不同类加载器提供不同的命名空间来实现的，在java虚拟机中，在同一个命名空间内的类可以直接进行交互，而不同的命名空间中类甚至不能觉察彼此的存在，除非显式地提供允许它们交互的机制。

1. 守护了被信任的类库的边界

虚拟机通过使用不同的类装载器装载可靠的包和不可靠的包，即所谓的双亲委派模式：在某个特定的类装载器试图以常用方式装载类型之前，它会先默认将这个任务“委派”给它的双亲，这个双亲再依次请求自己的双亲来装载这个类型。这个委派的过程一直向上继续，直到达到启动类加载器（boostrp classloader），如果一个类加载器的双亲类加载器有能力装载这个类型，则这个类加载器返回这个类型，否则这个类装载器试图自己来装载这个类型。

1. 将代码归入某类（称为保护域），该类确定了代码可以进行哪些操作。
2. class文件检验器；

和类加载器一起，class文件检验器保证了装载的class文件内容中有正确的内部结构，并且这些class文件相互间协调一致，如果class文件检验器在class文件中发生了问题，它将抛出异常。

java虚拟机的class文件检验器在字节码执行之前，必须完成大部分检验工作。class文件检验器需要经过四趟独立的扫描来完成其工作。

第一趟扫描：class文件结构检查。对每一段被当做类型导入的字节序列，class文件检验器都会确认其是否符合class文件格式。

第二趟扫描：类型数据的语义检查。检验一些java语言应该在编译时遵守的强制规则。

第三趟扫描：字节码验证。它确保局部变量在赋值之前不可访问，类的字段中总必须赋予正确类型的值等等。

第四趟扫描：符号引用验证。在这趟检查中，java虚拟机将追踪那些引用，从被验证的class文件到被引用的class文件，确保这个引用是正确的。

1. 内置于java虚拟机（及语言）的安全特性；

java虚拟机装载了一个类，并且对它进行了第一到第三趟的class文件检验，这些字节码就可以被运行了。除了对符号引用的检验（class文件检查的第四趟扫描），java虚拟机在执行字节码时，还进行了一些内置的安全机制的操作。这些机制大多数是java类型安全的基础：

1.类型安全的引用转换；

2.结构化的内存访问（无指针算法）；

3.自动垃圾收集；

4.数组边界检查；

5.空引用检查。

1. 安全管理器及java API。

安全管理器定义了沙箱的外部边界，保护虚拟机的外部资源不被虚拟机内运行的恶意或者有漏洞的代码侵犯。javaAPI在进行一个可能不安全的操作前，总是会检查安全管理器，所以javaAPI不会在安全管理器建立的安全策略下执行被禁止的操作。

1. 数据传输安全机制

数据传输过程中，安全性十分重要，尤其是对于一些敏感的用户信息，其安全性保证更应当被重视。保证传输数据的安全性，主要有以下方法：

1. 数据加密

数据加密一直是数据安全的重要部分，数据加密也就是利用技术手段对重要数据进行加密，然后把加密过后的数据进行传输，到达目的地后再用相同或不同的手段进行解密。对数据加密的技术分为两类,即对称加密技术和非对称加密技术。

* 对称加密

对称加密的特点是加密与解密时采用相同的密钥。对称加密算法简单快捷,密钥较短,且破译困难。对称加密以数据加密算法(DES, Data Encryption Standard)为典型代表。对称加密在使用中会存在以下几个问题：

1.通讯双方在首次通讯时需要一个安全的通道协商一个共同的密钥、

2.密钥的数目难于管理、

3.对称加密算法一般不能提供信息完整性的鉴别、

4.对称密钥的管理和分发工作是一件具有潜在危险的和烦琐的过程。

* 非对称加密

非对称加密算法需要有两个密钥，公开密钥和私有密钥。一般来说，公钥并不需要对任何人保密，是公开的。私钥并不公开，只有其所有者知道。如果用公开密钥对数据进行加密,只有用对应的私有密钥才能解密如果用私有密钥对数据进行加密,那么只有用对应的公开密钥才能解密。因为加密和解密使用的是两个不同的密钥,所以这种算法叫作非对称加密算法。

非对称加密算法的保密性比较好,它消除了最终用户交换密钥的需要,但加密和解密花费时间长、速度慢,它不适合于对文件加密而只适用于对少量数据进行加密。

1. 数字签名

数据加密只解决了传输数据机密性的问题，如果需要确认发信者的身份，则需要使用数据签名的技术。数字签名是建立在公开密钥体制基础上,它是公开密钥加密技术的另一类应用。它的主要方式是，报文的发送方从报文文本中生成消息摘要，发送方用自己的私有密钥对这个消息摘要进行加密来形成发送方的数字签名，然后,这个数字签名将作为报文的附件和报文一起发送给报文的接收方。报文的接收方首先对接收到的原始报文生成消息摘要，接着再用发送方的公开密钥来对报文附加的数字签名进行解密。如果两个值相同、那么接收方就能确认该数字签名是发送方的。

使用数字签名，能确认以下两点：第一，信息是由签名者发送的；第二，信息自签发后到收到为止未曾作过任何修改。

1. SSL协议

SSL（安全套接字）协议利用对称加密技术和公开密钥加密技术，在两点之间交换私钥建立了安全的连接。该协议提供了以下的安全服务：

1.认证用户和服务器，确保数据发送到正确的客户机和服务器；

2.加密数据以防止数据中途被窃取；

3.维护数据的完整性，确保数据在传输过程中不被改变。

1. 应用服务的安全策略
2. 基于JAVA的移动电商平台安全性风险
3. 单点服务故障
4. 缓存击穿与缓存雪崩
5. SQL注入风险
6. 第三方服务消息泄露
7. 总结

参考文献

［1］快消品移动电商…