

网络中间件的测试方法研究

陈琳¹, 孙富元²

(1. 联想研究院, 北京 100085; 2. 天津理工学院 信息技术研究所, 天津 300191)

摘要: 全面介绍软件接口测试项目。通过实例给出一种面向测试自动化的网络中间件性能测试, 网络状况模拟和网络设备批量测试的设计和实现方法。

关键词: 测试; 网络中间件; 软件接口函数(API)

中图分类号: TP391.76 文献标识码: A 文章编号: 1001-3695(2005)03-0080-02

Study on Test Methods of Network Middleware

CHEN Lin¹, SUN Fu-yuan²

(1. *Lenovo Corporate R & D, Beijing 100085, China;* 2. *Institute of Information Technology, Tianjin University of Technology, Tianjin 300191, China*)

Abstract: The software API test is introduced completely. Test methods on network middleware are also given through examples, which designed and realized the performance test, network simulating and batch test.

Key words: Test; Network Middleware; API

对软件中间件提供的接口函数的测试是软件测试过程中十分重要的部分,它在单元测试和集成测试中都有应用,只有保证接口函数的稳定性,才能从根本上保证基于接口函数开发的其他软件程序的稳定、可靠。基于网络实现的网络中间件软件又有别与其他普通的中间件软件。本文介绍如何进行网络中间件的接口测试,并且给出一些具体测试的例子。

1 网络中间件测试架构

从图 1 可以看出网络中间件包含两个外部接口:网络接口和 API 接口。一个网络中间件通过网络接口和其他网络中间件交换网络消息。同时,它通过 API 接口和本机上基于网络中间件开发的应用程序进行交互。

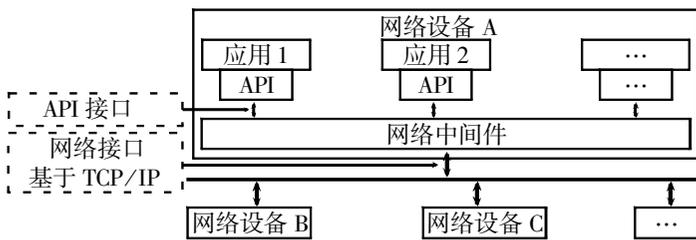


图 1 网络中间件框架结构

网络设备 A, B, C 上都运行相同的网络中间件或者运行着基于同一个标准而由不同厂商实现的不同网络中间件。这些设备处于同一个物理网络中时就能够相互发现、相互识别对方是同类网络设备。在经过安全认证后这些网络设备就能够发现并且使用其他网络设备共享出来的资源,如计算资源、存储资源、网络通信资源等。

2 软件接口测试评述

软件接口测试是一项复杂的测试工作,有如下几种:

(1) 功能性测试。功能性测试主要验证。网络中间件提供的 API 函数是否能够相互配合完成网络中间件设计目标。

在功能性测试时,主要考虑几方面的测试:当输入合法的参数时,验证 API 函数能否返回预期的正确结果;当输入非法的参数时,验证 API 函数能否正确处理非法的输入参数;

验证 API 函数在本地执行时(不需要经过网络与别的网络中间件发生交互)是否正确;验证 API 函数执行过程需要别的网络中间件配合时,是否正确;验证运行在不同平台上的网络中间件之间是否能够正确交互、并且返回正确响应结果。

(2) 一致性测试。它主要验证按照同一个标准文本实现的不同网络中间件之间是否能够正常交互信息且能够正常处理这些交互信息。例如基于 Bluetooth 蓝牙协议文本由不同厂商开发的不同蓝牙设备之间是否能够相互识别对方、相互连接在一起并且相互交换数据信息均属于一致性测试。

(3) 网络测试。网络测试包括:运行网络中间件的设备自身网络状况不稳定时的测试;运行中间件的设备之间通过不同的网络连接方式连接在一起进行测试,验证通过中间件的消息能否通过网络设备到达目标设备。

(4) 安全性测试。网络中间件软件对网络安全自然有很高的要求,因此需要进行安全测试,其目标是验证符合安全规则的网络设备之间的交互能够被正确处理,对于不符合安全规则的交互作出相应的响应。在测试时需要重点模拟一些网络攻击,包发送给网络中间件,验证网络中间件是否能够正确处理这些攻击(此时会出现网络中间件资源占用率可能上升、处理性能也可能降低、不能及时处理所有的数据包而丢掉部分的数据包),同时验证在攻击结束后网络中间件是否能够恢复正常运行。

(5) 性能测试。它需要测试网络中间件处理单个函数调用需要的时间,以及在一定软硬件环境下网络中间件软件对网

络带宽、内存资源、CPU 资源等的占用情况。

(6) 压力测试。它需要模拟环境中存在很多设备, 而某个设备的资源被其他多个设备持有, 或大数据量访问时, 网络中间件软件是否仍然能够正确处理这些请求, 同时需要测试出在压力情况下网络中间件的性能数据。

(7) 兼容性测试。兼容性测试就是验证被测软件是否能够在目标平台上稳定运行。它包括两个方面, 即硬件兼容性和软件兼容性(软件兼容性包括 OS 软件兼容性和其他应用软件的兼容性)。

(8) 批量测试。一般而言, 网络软件产品在投入市场之前需要进行一项批量测试。就是将一定数量的设备按照某种方式连接成一个物理网络, 在这个网络中进行设备的功能性测试和性能测试。以验证当一定数量的网络设备存在于网络中时这些设备是否仍然能够正常运行。

3 网络中间件的测试示例

下面给出在实际测试过程中, 是如何组织、设计这些测试的。这里提出了一种网络中间件的测试方法。为便于说明, 我们重点介绍性能测试、网络测试、安全测试、批量测试。当然在测试过程中需要给出测试结果(如测试记录、测试配置、实际数据等)。

3.1 性能测试流程

对网络中间件的性能测试过程是通过监控网络中间件, 测试程序对系统资源的占用(包括 CPU 计算资源占用、存储空间占用、网络带宽的占用), 记录每个 API 函数执行时间、网络中间件启动/停止需要的时间。最后绘出一个图形化显示的结果。图 2 给出了进行性能测试的步骤。

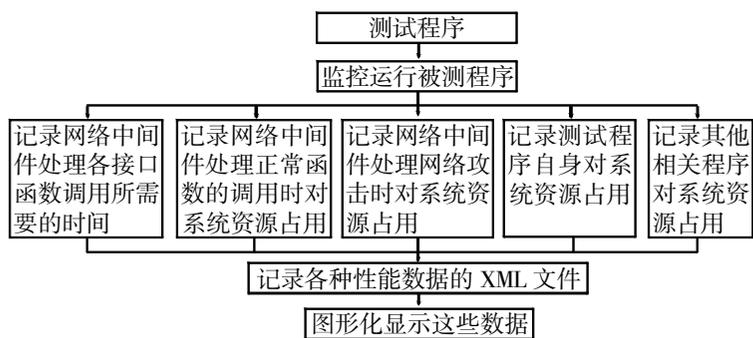


图 2 性能测试流程

利用不同 OS 平台提供的一些监控函数, 测试程序就可以很方便地监控并记录正在运行的各种程序对系统资源的占用。记录结果可以保存成一个或几个 XML 文件。利用相关软件对 XML 文件进行离线分析, 将这些记录的数据进行整理和分类, 最后以图形化的方式显示性能记录结果。

3.2 网络状况的模拟测试

模拟网络状况对网络中间件进行网络安全测试就是通过测试程序实现以一定的间隔启用/禁用网卡, 给网卡增加多个 IP 地址, 以一定的间隔发送不同大小的数据包, 从而统计出在网络不正常时, 网络中间件是否仍然能够正确处理这些情况, 并且在网络恢复正常后网络中间件能够自动恢复正常运行。

对 Microsoft Windows 平台上的网络中间件, 测试程序可以通过 MS DDK 中提供的 Windows Setup API 和 CFGMGR API 来实现对网卡的启用和禁用。

被调用的主要函数就是 setupapi.h 提供的函数:

```

WINSETUPAPI
BOOL
WINAPI
SetupDiGetDeviceRegistryPropertyA(
    IN HDEVINFO DeviceInfoSet,
    IN PSP_DEVINFO_DATA DeviceInfoData,
    IN DWORD Property,
    OUT PDWORD PropertyRegDataType, OPTIONAL
    OUT PBYTE ProperBuffer,
    IN DWORD PropertyBufferSize,
    OUT PDWORD RequiredSize, OPTIONAL
);

DWORD WINAPI
AddIPAddress(
    IPAddr Address,
    IPMask IpMask,
    DWORD IfIndex,
    PULONG NTEContext,
    PULONG NTEInstance
);

DWORD WINAPI
DeleteIPAddress(
    ULONG NTEContext);

```

通过 iphlpapi.h 提供的添加/删除 IP 地址函数就能够进行模拟添加/删除 IP 地址的操作。

3.3 批量测试

对网络中间件的批量测试是中间件产品投入市场前的一项重要工作。在进行批量测试时由于工作量大并且测试时间短, 因此需要其他执行人员配合, 而这些人员往往对被测对象可能不是特别了解, 所以要求在设计批量测试用例时要尽量简化、容易执行、容易判断是否通过测试。

图 3 给出了网络中间件的批量测试网络结构图。其中 RT1, RT2 是两个路由器, SW1 ~ SW8 是八个三层以太网交换机, G1 ~ G7 是七个网络设备组。每个设备组分别有八台运行被测网络中间件的网络设备(共 56 台不同类型的基于网络中间件开发的网络设备)。这 56 台网络设备依据所在组依次编号为 D11 ~ D18, D21 ~ D28, D31 ~ D38, D41 ~ D48, D51 ~ D58, D61 ~ D68, D71 ~ D78。在每个组中设备都分为两类, 即资源提供者和资源使用者。简单起见, 每个设备组指定编号 1 的设备为资源提供者(提供基于网络中间件开发能够被其他网络中间件设备使用的资源, 如提供网络访问资源、屏幕显示资源、存储资源、计算资源等)。其余设备为资源的使用者, 可以同时使用这些资源。在 G1 ~ G5 组中, 资源使用者使用本组内资源提供者的共享资源。在 G6, G7 组中, G6 组中的资源使用者使用 G7 组资源提供者的共享资源, 而 G7 组中的资源使用者使用 G6 组资源提供者的共享资源, 这样设计的目的是用于验证在跨过 2 级路由器后, 网络设备之间是否仍然能够进行资源的使用。

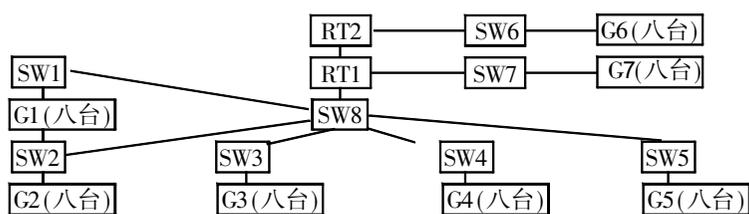


图 3 网络中间件的批量测试网络结构图

批量测试需要验证如下几点:

- (1) 一定数量的网络设备通过某种方式连接 (下转第 84 页)