

doi: 10.7690/bgzdh.2024.08.010

基于框架的 Web 服务软件自动化测试技术

黎 晖¹, 于宏宇¹, 张绍平², 林柯军¹

(1. 陆军研究院五所, 江苏 无锡 214035; 2. 常熟市亚邦船舶电气有限公司, 江苏 常熟 215500)

摘要: 为弥补传统手动 Web 服务软件测试周期长、效率低等缺点, 设计一种基于框架的 Web 服务软件自动化测试技术。介绍测评体系结构, 设计 Web 服务自动化测试框架; 基于该框架进行综合安全监管系统软件的自动化测试技术应用研究, 在软件需求分析的基础上进行测试数据设计, 并自动生成测试用例; 利用自动化测试框架实现数据驱动脚本自动测试, 列出测试用例执行结果并分析软件问题。结果表明, 该技术能缩短软件测试周期、提高测试效率和测试覆盖率。

关键词: 软件测试; Web 服务; 自动化测试框架; 需求分析

中图分类号: TP23 **文献标志码:** A

Framework-based Automation Testing Technology for Web Services Software

Li Hui¹, Yu Hongyu¹, Zhang Shaoping², Lin Kejun¹(1. *The Fifth Institute of Army Academy, Wuxi 214035, China;**2. Changshu YBTECH Marine Electric Co., Ltd., Changshu 215500, China)*

Abstract: In order to make up for the shortcomings of traditional manual Web services software testing, such as long cycle and low efficiency, a framework-based automation testing technology for Web services software is designed. This paper introduces the evaluation architecture and designs the Web services automation testing framework. Based on this framework, the application research of automation testing technology for integrated safety supervision and management system software is carried out. Based on the software requirement analysis, the test data is designed and the test cases are automatically generated; The automation testing framework is used to implement data-driven script automation testing, and the results of test case execution are listed and the software problems are analyzed. The results show that the technology can shorten the software testing cycle, improve the testing efficiency and increase the test coverage.

Keywords: software testing; Web services; automation testing framework; requirement analysis

0 引言

Web 服务是一个基于可编程的 Web 应用程序, 作为开发分布式的互操作应用程序的新平台, 具有平台独立、低耦合、数据重用及无缝连接等优点。无需附加第三方专用软件或硬件, 就能实现基于不同部署平台、通信协议、编程语言开发的 Web 应用软件之间的数据同步交互和信息集成^[1]。Web 服务逐渐成为当前主流的网络分布式应用软件开发技术, 已在国防建设与国民生活等领域得到广泛应用。

为提高 Web 服务软件的可靠性, 需要进行软件测试以发现并消除可能存在的软件缺陷, 从而在最大程度上提高软件产品的质量^[2]。Web 服务软件由于自身的特点, 使其在进行软件测试时需要服务端、服务注册中心和客户端同时参与。客户端作为 Web 服务的直接使用者, 使得软件功能、性能等需求都是基于客户端实现的, 在客户端进行软件测试才能够真实反映 Web 服务工作时的功能需求满足情况, 以及增加用户数量或网络负载时的性能需求满足情

况; 因此, 测试主要针对客户端进行^[3]。由于 Web 服务端只对客户端提供服务调用接口, 因此, 针对客户端进行的 Web 软件测试主要建立在服务接口测试上。

由于 Web 服务软件的层级接口众多、操作相似性与重复性繁多, 采用手动方式进行软件测试不仅周期长、效率低, 而且较难覆盖所有需求点。手动测试已经不能满足日益增加的回归测试量^[4]。自动化测试是利用软件测试工具开发测试脚本, 驱使机器自动完成软件测试, 很大程度上节约了时间、人力、物力, 并可提高测试覆盖率与效率^[5]。笔者设计一种基于框架的 Web 服务软件自动化测试技术, 并以综合安全监管系统软件作为测试对象来验证其可行性。

1 软件测评体系结构

Web 服务自动化测试框架是在测评体系结构下建立的, 如图 1 所示, 该测评体系主要包括自动化

收稿日期: 2024-04-19; 修回日期: 2024-05-23

第一作者: 黎 晖(1984—), 男, 江西人, 博士。

测试平台、测试管理平台以及通用测试平台。

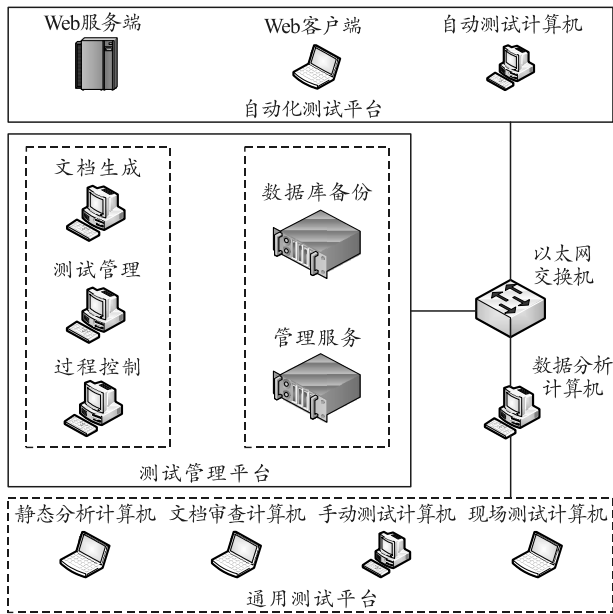


图 1 软件测评体系结构

自动化测试平台主要由 Web 服务端、Web 客户端以及自动测试计算机组成。自动测试计算机主要根据需求分析与测试用例设计结果进行仿真测试环境的搭建、测试数据文件的设计，以及测试脚本的开发与调试工作，利用 SoapUI 工具将固化后测试脚本生成对应的自动测试用例，将测试环境通过以太网导入实际 Web 客户端，在客户端通过服务接口调用方式，完成 Web 服务软件自动化测试。

测试管理平台的服务端包括数据库备份服务与管理服务，客户端包括测试文档生成、测试过程控制与测试管理维护客户端，过程控制端用于完成测评项目的需求分析与策划、测试设计与实现、测试执行与统计总结，测试管理端完成项目的质量保证、配置管理与跟踪控制。

通用测试平台则可完成 Web 服务软件的手动补充测试、软件文档审查、静态分析以及现场测试。

数据分析计算机用于将自动化测试平台与通用测试平台得到的测试记录进行数据规格化，并备份到测试管理平台的数据库备份服务端的对应数据库中。以太网交换机用于自动化测试平台、测试管理平台和通用测试平台之间的数据传输。

2 Web 服务自动化测试框架设计

2.1 Web 服务自动化测试框架

软件测评体系结构的自动化测试平台主要基于 Web 服务自动化测试框架，实现测试脚本开发与维护、测试用例生成与执行、测试结果验证与报告生

成，最终完成自动化测试任务。Web 服务自动化测试框架设计结构如图 2 所示，测试框架采用模块化分层方式，主要由测试数据模块、测试用例模块、参数配置模块、测试执行模块和统计报告模块等组成。

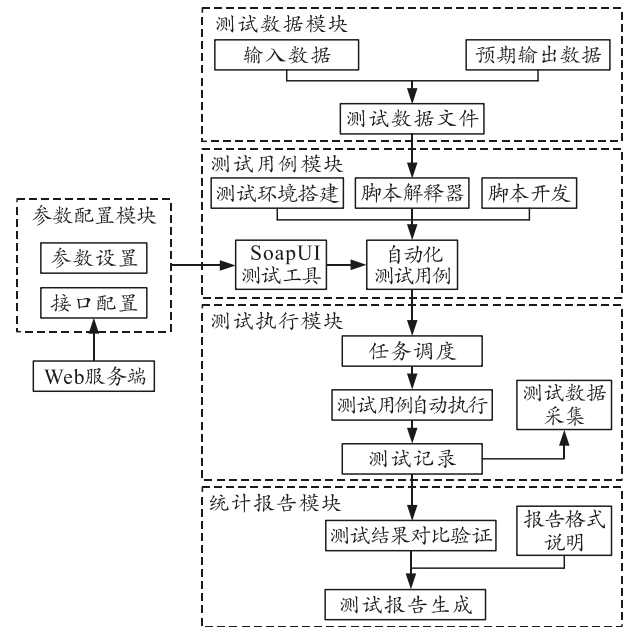


图 2 Web 服务自动化测试框架

Web 服务自动化测试框架基于数据驱动技术与模块化技术进行设计，将测试数据与测试脚本独立开发封装，可以提高它们的复用性和维护性^[6]。在测试数据模块中，测试数据文件包括输入数据与预期输出结果数据，测试数据文件保存在测试数据库中，当执行测试用例时通过变量形式读取对应的数据文件；在测试用例模块中，主要根据搭建的测试环境进行测试脚本的开发，通过编程并利用 SoapUI 测试工具将测试脚本转换为对应的自动化测试用例；参数配置模块主要用于分析测试软件的通讯协议及不同类型的数据变量，并设置测试客户端与 Web 服务端的接口与参数配置，以完成测试端与服务端的数据通讯；测试执行模块在任务调度安排下完成测试用例的自动化执行，并将测试结果记录进行采集与保存；统计报告模块则将实际测试结果与预期输出数据进行对比验证，并生成软件测试报告。

2.2 工作原理

Web 服务自动化测试过程是通过各功能模块的互相配合完成的，其主要工作原理为：对目标 Web 服务软件进行需求分析，确定软件的功能测试与性能测试覆盖范围，并对 Web 服务接口通讯协议进行

解析。在自动化测试计算机上，依据需求分析结果进行测试数据文件的设计，并完成测试脚本的开发。利用测试工具进行接口参数配置，利用脚本解释器将测试脚本转换成自动测试用例^[7]。然后执行自动化测试用例，并验证测试脚本的可行性。将固化后测试环境导入真实客户端，基于服务接口调用方式，完成测试用例的自动执行，数据分析计算机将测试记录数据规格化后备份到对应的数据库。同时将预期输出数据与测试用例实际执行结果进行对比验证，结果一致则用例通过，不一致则产生对应的软件问题。自动测试过程均生成对应的测试日志，统计软件问题，并按照规定格式输出软件测试报告，最终完成了 Web 服务软件的自动化测试。

3 Web 服务自动化测试技术应用

综合安全监管系统软件是用于信息安全领域的非嵌入式 Web 服务软件，首先基于软件需求规格说明与软件设计说明等技术文档，对测试软件进行需求分析与策划；然后基于 Web 服务自动化测试框架，进行综合安全监管系统软件的自动化测试。

3.1 软件需求分析

综合安全监管系统软件主要用于对信息网络环境中的软硬件资产状态、运行环境和系统运行中产生的信息进行集中统一监测与安全管理。综合安全监管系统软件采用 B/S 结构设计，服务器端程序采用 JSP 标准实现，采用 MYSQL 数据库并通过 JDBC 访问，服务通信接口采用 HTTP 协议完成。综合安全监管服务接收的原始日志来源于采集代理采集的监管对象日志信息。综合安全监管系统软件完成的主要功能如图 3 所示。

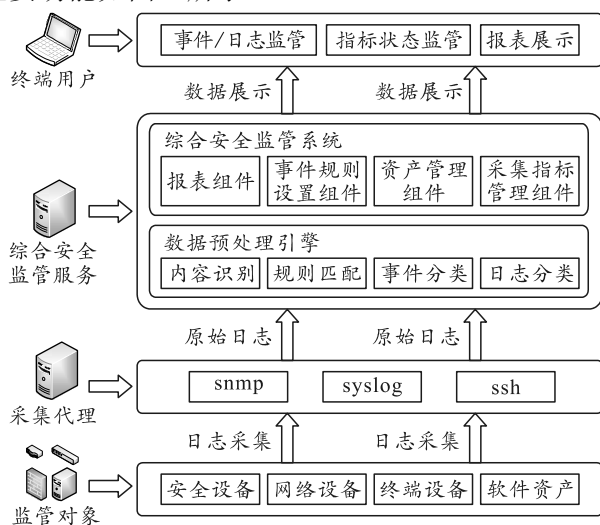


图 3 综合安全监管系统功能结构

1) 指标状态监管：软件基于配置的采集指标，能够实时展现监测到的网络设备、安全设备、计算机设备、软件资产等性能与状态信息；

2) 事件监管：软件基于数据库读取事件信息，能够在安全事件、异常事件页面中展示网络防病毒、入侵监测系统、设备异常行为等事件信息；

3) 报表展示：软件能够显示系统常规报表与自定义报表，并通过表格与图形方式展示报表具体内容，且能够将报表进行导入导出。

3.2 测试数据设计与用例生成

限于篇幅原因，笔者针对软件的指标状态监管功能点进行对应测试数据设计。服务端接收代理采集发送的采集指标原始日志，经过数据处理后在状态管理页面显示监管对象的设备类型、IP、指标参数与采集时间等信息。设计监管对象类型为防火墙，采集指标为 CPU 使用率。由于 CPU 使用率显示值在 0~100 之间，当采集指标不在此范围内，或者输入 IP 格式不正确时，软件判定接收数据格式错误，并给出对应异常状态提示。根据等价类划分法设计的正常与异常的测试数据如表 1 所示。

表 1 采集指标测试数据

IP	KpiType	Value	Date	reValue	reIP
192.168.201.11	CPU	-1.0	13:00:07	null	1
192.168.201.12	CPU	-0.1	13:00:23	null	1
192.168.201.13	CPU	0	13:00:53	0	1
192.168.201.14	CPU	0.1	13:00:23	0.1	1
192.168.201.15	CPU	10.0	13:01:02	10	1
192.168.201.16	CPU	100.0	13:01:36	100	1
192.168.201.17	CPU	100.1	13:01:58	null	1
192.168.201.18	CPU	101.0	13:02:14	null	1
192.168.201.	CPU	110.0	13:02:46	10	0
192.168.201	CPU	a	13:03:19	null	0
192.168.2012	CPU	+	13:05:01	null	0

注：Date 列的日期均为 2022-2-8

表中：IP 为监管设备 IP 号；KpiType 为采集指标类型；Value 为发送的指标参数值；Date 为指标采集时间；reValue 为指标参数的预期返回结果，当发送正常指标参数时返回值为发送参数，当发送异常指标参数时，软件应判断异常并返回 null；reIP 为 IP 状态的预期返回结果，当发送正常 IP 号时返回 1，当发送异常数据时，软件应判断异常并返回 0。其他采集指标数据设计方法相同，本文中不再累述。

设计的测试数据表保存在数据库中，自动测试用例生成时调用数据库相关测试数据表的输入数据，即通过变量赋值形式导入测试脚本当中^[8]。测试用例每次循环执行都调用表中下一行数据，执行

结束时将实际测试结果与测试数据表的预期输出数据进行比较，从而判断该测试用例对应功能点是否满足软件需求。针对指标状态监管功能点的 CPU 使用率指标测试用例的测试数据包含在上表内，该测试用例包括 12 个测试步骤，每个步骤由一个 HTTP 发送和对应的 HTTP 输出响应组成，输出结果与数据表对应预期结果相一致，则认为该步骤成功执行，所有步骤成功执行则认为该条用例通过。

3.3 自动化测试实现

利用 SoapUI 测试工具模拟采集代理解析并调用安全监管系统软件的 Web 服务接口，SoapUI 自带的 DataSource 功能可用于实现数据驱动脚本自动化测试技术，只需添加并配置存有测试数据文件的 MYSQL 数据库，将测试脚本中的变量与测试数据表中对应参数相互关联，便可在数据驱动下进行测试用例自动化执行^[9]，基于 SoapUI 的自动化测试如图 4 所示。

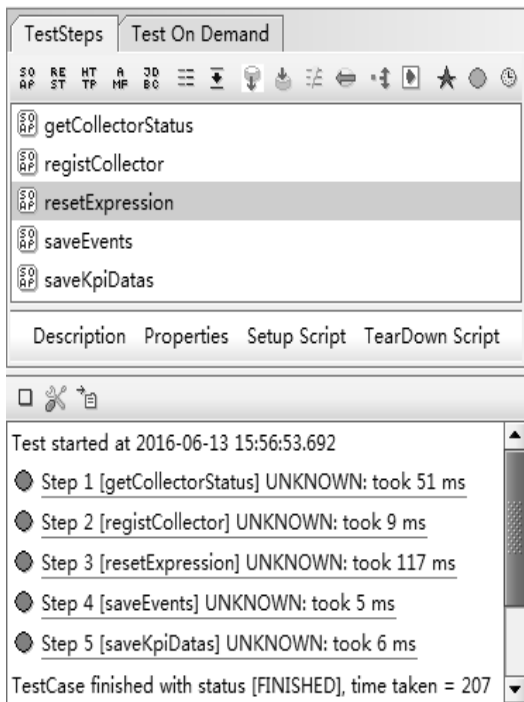


图 4 SoapUI 自动化测试过程

自动化测试过程中，每个循环都对监管系统软件所有 Web 服务接口进行一次发送与返回。以指标状态监管功能点为例进行说明，该功能点的服务接口名称为 saveKpiDatas，其对应接口发送请求的主要 XML 测试脚本如下：

```
<hur:save KpiDatas>
  <beans>
  </item>
```

```
<date>2016-6-8 10:31:08</date>
<ip>192.168.201.11</ip>
<kpiType>cpu</kpiType>
<value>66</value>
</item>
</beans>
</hur:saveKpiDatas>
```

date、IP、kpiType 与 value 等变量与测试数据表中对应参数相互关联，每次循环执行，变量值都在对应测试数据表中逐行读取。当读取完数据表末尾行时，该接口结束调用。接口输出响应的主要 XML 测试脚本如下：

```
<ns1:save KpiDatas Response xmlns:ns1="http://hurricane">
  <reValue>66</ reValue>
  <reIp>1</ reIp>
  <result>Success</result>
</ns1:saveKpiDatasResponse>
</hur:save KpiDatas>
```

返回的 reValue 和 reIP 值与测试数据表中对应的预期输出结果进行比较，每次循环返回值都与预期结果相一致，该用例才执行通过。所有测试用例自动化执行完毕后，将生成每次执行对应的测试日志，并生成最终的测试报告。

3.4 测试结果分析

经过自动化测试后，得到安全监管系统软件的指标状态监管功能点的 CPU 指标监管测试用例执行结果如表 2 所示。

表 2 测试用例执行结果

输入 IP	输入 Value	实际 reValue	预期 reValue	实际 reIP	预期 reIP
192.168.201.11	-1.0	-1	null	1	1
192.168.201.12	-0.1	-0.1	null	1	1
192.168.201.13	0	0	0	1	1
192.168.201.14	0.1	0.1	0.1	1	1
192.168.201.15	10.0	10	10	1	1
192.168.201.16	100.0	100	100	1	1
192.168.201.17	100.1	100.1	null	1	1
192.168.201.18	101.0	101	null	1	1
192.168.201.	110.0	110	null	1	0
192.168.201	a	a	null	1	0
192.168.2012	+	+	null	1	0

从上表中可知，利用测试工具模拟采集代理向综合安全监管系统软件发送异常 IP 号与指标值 Value 的 save KpiDatas 接口数据时，软件没有对错误 IP 格式以及超限或特殊字符的 Value 值进行判断处理。在事件监管功能点对应的 save Events 接口中已发现类似问题。

4 结束语

Web 服务软件已在国防建设与国民生活等领域得到广泛应用。为提高 Web 服务软件的可靠性，需要进行软件测试以提高软件产品质量。由于 Web 服务软件具有层级接口众多与操作重复性多等特点，采用手动方式进行软件测试不仅周期长、效率低，且较难覆盖所有需求点。为解决上述问题，笔者基于自动化测试框架，提出一种 Web 服务软件自动化测试技术的具体实现方法，并以综合安全监管系统软件作为测试对象进行应用研究。首先基于软件需求规格说明等技术文档，对被测软件进行测试需求分析与策划，然后利用 Web 服务自动化测试技术进行综合安全监管系统软件的自动化测试。结果表明：采用自动测试框架的 Web 测试技术。能够提高软件测试效率并增强软件测试充分性。

参考文献：

- [1] 侯俊, 周红, 马春燕, 等. 面向 WEB 服务的测试用例自动化生成方法[J]. 西北工业大学学报, 2018, 36(1): 149-155.
- [2] 丁志军, 周泽霞. Web 服务组合测试综述[J]. 软件学报, 2018, 29(2): 299-319.
- [3] 杨利利, 李必信. Web 服务测试问题综述[J]. 计算机科学, 2008, 35(9): 258-265.
- [4] 朱建华. 基于 Spring 的 Web 自动化测试平台设计与实现[D]. 大连: 大连理工大学, 2019.
- [5] 胡铮. 软件自动化测试工具实用技术[M]. 北京: 科学出版社, 2011: 6-11.
- [6] 樊付星, 黄大庆, 周末. 基于 Web 的自动化测试框架的研究与实现[J]. 电子设计工程, 2012, 20(20): 36-38.
- [7] SAMER H, MALCOLM M. An approach for specification-based test case generation for web services[C]//IEEE/ACS International Conference Computer Systems and Applications. Washington: 2007: 16-23.
- [8] 姜瑛, 辛国茂, 单锦辉, 等. 一种 Web 服务的测试数据自动生成方法[J]. 计算机学报, 2005, 28(4): 568-576.
- [9] 罗作民, 朱燕, 程明. Web 服务测试工具 SOAPUI 及其分析[J]. 计算机应用与软件, 2010, 27(5): 155-158.