知网个人查重服务报告单(全文对照)

报告编号:BC20250505131453572448429 9　　检测时间:2025-05-05 13:14:53

篇名：嵌入式JS引擎模糊测试方法研究

作者：雷璟锟

检测类型：毕业设计

比对截止日期：2025-05-05

检测结果

去除本人文献复制比：6.2%　　去除引用文献复制比：5.9%　　总文字复制比：6.2%

单篇最大文字复制比：2%（基于深度学习的JavaScript引擎模糊测试方法研究）

重复字符数：[1849]　　单篇最大重复字符数：[590]　　总字符数：[29899]

4%(420)	4%(420)	嵌入式JS引擎模糊测试方法研究.doc_第1部分（总10396字）
9.2%(973)	9.2%(973)	嵌入式JS引擎模糊测试方法研究.doc_第2部分（总10575字）
5.1%(456)	5.1%(456)	嵌入式JS引擎模糊测试方法研究.doc_第3部分（总8928字）

（注释：无问题部分　　文字复制部分　　引用部分）

1. 嵌入式JS引擎模糊测试方法研究.doc_第1部分　　总字符数：10396

相似文献列表

去除本人文献复制比：4%(420)　　去除引用文献复制比：3.3%(340)　　文字复制比：4%(420)

1	面向嵌入式JavaScript引擎的差分模糊测试方法研究 姚厚友(导师：牛进平;汤战勇) - 《西北大学硕士论文》- 2021-06-01	1.9%（194） 是否引证：否
2	面向QuickJS的JSDOM浏览器测试运行环境的移植与实现 陈宇畅 - 《大学生论文联合比对库》- 2023-06-12	1.4%（146） 是否引证：否
3	1711380_王继铭_JavaScript执行引擎漏洞分析技术研究 王继铭 - 《大学生论文联合比对库》- 2021-06-07	0.8%（80） 是否引证：否

	原文内容	相似内容来源
1	此处有 48 字相似 于各类型设计丰富的变异策略与自调用表达式，最终生成了大量语法语义正确、高覆盖率的测试用例。（3）基于上述方法，本文 设计并实现了原型系统EJSFuzz，使用该系统对主流的嵌入式JavaScript引擎进行模糊测试 ，对发现的缺陷进行案例分析。通过与Fuzzilli，AFL等先进模糊测试工具的对比实验，验证了EJSFuzz在代码覆盖率	面向嵌入式JavaScript引擎的差分模糊测试方法研究　姚厚友 -《西北大学硕士论文》- 2021-06-01（是否引证：否） 1.避免分析重复测试结果造成人力资源浪费。通过测试用例精简和测试结果过滤可以极大地降低测试结果的分析成本。（4）设计并实现 JSDiff 原型系统基于面向嵌入式 Java Script 引擎的差分模糊测试方法设计并实现了 JSDiff 原型系统。使用 JSDiff 原型系统对嵌入式 Java Script 引擎进行测试
2	此处有 48 字相似 心技术。AFL++、WinAFL、LibFuzzer等交互友好、操作简单的工具相继涌现，极大地促进了模糊测试技术的普及。 下面介绍一些最新的模糊测试技术相关的安全研究。 *Christian Holler等人提出了一种* 基于语法和代码片段重组的模糊测试框架LangFuzz[1]，利用上下文无关的语法随机生成有效程	面向嵌入式JavaScript引擎的差分模糊测试方法研究　姚厚友 -《西北大学硕士论文》- 2021-06-01（是否引证：否） 1.测编译器安全缺陷的主流方法。模糊测试是通过检测软件崩溃，并根据软件崩溃是否可用于安全攻击而决定其是否是安全缺陷。下面将介绍几种最新的利用模糊测试技术检测安全缺陷的方法。Gro? S[7]提出了一种字节码级别的测试用例变异方法 Fuzzilli。通过自定义

	序，确保输入通过语法检查。	一种字节码级别的中间语言 Fuzz IL，并在此中间代码上
3	此处有 80 字相似 组生成新测试用例，提高触发异常的概率。LangFuzz在Mozilla JavaScript引擎中发现105个高危漏洞。 *Suyoung Lee等人将神经网络语言模型（NNLM）Montage[2]用于JavaScript引擎模糊测试。* Montage创新地将JS代码的抽象语法树（AST）分解为深度为1的子树片段序列，并基于LSTM（长短期记忆网络，Long Short-Term Memory）模	1711380_王继铭_JavaScript执行引擎漏洞分析技术研究 王继铭 -《大学生论文联合比对库》- 2021-06-07（是否引证：否） 1.L[18]上进行扩展，实现了代码覆盖率反馈，最终发现了 XML和 JavaScript执行引擎的22个安全缺陷。Suyoung Lee 等人[7] 实现了一个基于神经网络语言模型(NNLM)的模糊测试工具 Montage。Montage将JavaScript测试用例的抽象语法树转换为一系列抽象34第五章相关工作语法子树用于训练 NNLM，使用NNLM训练得到的新的抽象语法子树替换原来
4	此处有 48 字相似 码在语法和语义上均有效。该方法无需手动编写语法规则，自动学习JS语义，凸显了语义感知方法在漏洞挖掘中的优势。 1.3 本文研究内容 本文通过对嵌入式JavaScript引擎运行原理与模糊测试技术的研究，提出了一种基于参数类型的测试用例变异方法，通过分析原始测试用例得到代码段的抽象语法树，提取语义信息，确定可进行变异的参数，根据数据	面向嵌入式JavaScript引擎的差分模糊测试方法研究 姚厚友 -《西北大学硕士论文》- 2021-06-01（是否引证：否） 1.种缺陷的修复方法也非常简单，只需要在条件满足时跳出循环即可，修复此类缺陷不会引入新的缺陷即不具有入侵性。 本文研究内容本文通过对嵌入式 Java Script 引擎和差分模糊测试技术的深入研究，提出了一种面向嵌入式 Java Script 引擎的差分模糊测试方法。通过对获取的测试用例进行变异得到可用于测试嵌入式 Ja
5	此处有 50 字相似 yScript是三星公司开源的轻量级JavaScript引擎，目前已应用于三星物联网生态与华为鸿蒙生态。 2.1.2 嵌入式JavaScript引擎运行原理 JavaScript引擎负责解释和执行JavaScript代码，了解其内部架构对测试工作有重要意义。QuickJS 是Fabrice Bellard继FFmpeg和QEMU的又一	面向嵌入式JavaScript引擎的差分模糊测试方法研究 姚厚友 -《西北大学硕士论文》- 2021-06-01（是否引证：否） 1.顺利接入互联网，极具网络编程优势的 Java Script 成为嵌入式开发的首选编程语言，因此也出现了许多优秀的嵌入式 Java Script 引擎。Java Script 引擎负责解释并执行 Java Script语言编写的程序，其功能的正确性决定了 Java Script 程序是否能正确运行。Java Script引擎是执行
6	此处有 47 字相似 的架构与工作流程。 图2-1 QuickJs引擎架构 如图2-1所示，架构最顶层的是QJS与QJSC，其中QJS负责命令行参数解析、引擎环境的初始化、依赖模块的加载以及对JavaScript文件的读取与解释执行；QJSC则负责对JavaScript源代码进行编译，输出可执行的字节码。 QuickJS的整体架构以中间层为核心，其	面向QuickJS的JSDOM浏览器测试运行环境的移植与实现 陈宇畅 -《大学生论文联合比对库》- 2023-06-12（是否引证：否） 1.天大学毕业设计（论文 ）第 8 页如图 2.2 所示，QuickJS 架构中，其中最上层包含 qjs 和 qjsc，qjs 包含命令行参数处理，引擎环境创建，加载模块和 js 文件读取解释执行等。qjsc 能够编译 js 文件成字节码文件，生成的字节码可以直接被解释执行，性能上看是省掉了 js 文件解析的消耗。
7	此处有 99 字相似 封装标准功能集，OS Module则暴露文件操作、时间处理等系统级接口，共同构成完整的运行时能力体系。最底层是基础， JS_RunGC 使用引用计数来管理对象的释放。 JS_Exception 会将 JSValue 返回的异常对象存储在 JSContext 中，通过 JS_GetException 函数提取异常对象。 Memory Control负责管理 JS运行时的全局内存。Stack Control负责管理JS运行时的堆栈数据结构。	面向QuickJS的JSDOM浏览器测试运行环境的移植与实现 陈宇畅 -《大学生论文联合比对库》- 2023-06-12（是否引证：否） 1. 等。扩展模块 Std Module 和 OS Mod-ule，提供标准能力和系统能力，比如文件操作和时间操作。 JS_RunGC 使用引用计数来管理对象的释放。 JS_Exception 会把 JSValue 返回的异常对象存在 JSContext 里，通过 JS_GetException 函数取出异常对象。内存管理控制 js运行时不同内存使用函数不同，全局内存分配上限使用的是 JS_SetMemoryLimit 函数，

2. 嵌入式JS引擎模糊测试方法研究.doc_第2部分

总字符数：10575

相似文献列表

去除本人文献复制比：9.2%(973)　　去除引用文献复制比：9.2%(973)　　文字复制比：9.2%(973)

1	基于深度学习的JavaScript引擎模糊测试方法研究 刘艺玮(导师：李玉军) - 《电子科技大学硕士论文》- 2024-03-15	5.6%（590） 是否引证：否
2	面向嵌入式JavaScript引擎的差分模糊测试方法研究 姚厚友(导师：牛进平;汤战勇) - 《西北大学硕士论文》- 2021-06-01	2.3%（240） 是否引证：否
3	JavaScript高级程序设计：2. - 《互联网文档资源（https://www.360docs.）》- 2024	1.4%（143） 是否引证：否
4	六百音乐盒设计与实现 吴琪瑶 - 《大学生论文联合比对库》- 2022-05-17	1.3%（142） 是否引证：否
5	170747008 赵晓蕊 信本171 赵小鹏 论文正文 赵晓蕊 - 《大学生论文联合比对库》- 2021-04-30	1.1%（116） 是否引证：否
6	天然气管道分输站站控PLC程序的测试用例生成方法 陈安均 - 《大学生论文联合比对库》- 2024-06-17	0.6%（67） 是否引证：否
7	201908010611_梁菁蕾 梁菁蕾 - 《大学生论文联合比对库》- 2023-05-31	0.4%（40） 是否引证：否

原文内容		相似内容来源
1	此处有 363 字相似 2.2 模糊测试理论 模糊测试是一种自动化的软件测试方法，其核心在于向软件系统输入大量异常或随机生成的数据，并监测系统对这些输入的响应，以发现潜在的缺陷、漏洞或异常。通常，测试过程中会使用专门的软件工具（称为用例生成模型）来生成测试用例，并将其输入至目标程序执行。在程序运行期间，测试工具会持续监控其状态，并在检测到崩溃或异常时记录相关输入数据和执行结果。随后，测试人员对这些信息进行分析，以确定问题的根本原因并评估系统的安全性。相较于其他漏洞检测手段，模糊测试具有高度自动化的特点，无需深入的领域知识，同时具备良好的适应性和扩展能力，因而在软件安全测试中被广泛应用。 图2-3 模糊测试流程图 如图2-3所示，模糊测试的基本流程分为四个阶段：测试用例生成、执行测试用例、异常监视以及漏洞确认。 （1） 测试用例生成。 为了有效测试目标软件，测试用例模型需要不断生成新的测试用例，其生成质量直接影响测试的效果。目前，主流的用例生成方式分为生	201908010611_梁菁蕾　梁菁蕾 -《大学生论文联合比对库》- 2023-05-31（是否引证：否） 1.FL 相关研究模糊测试（Fuzz Testing）是一种自动化测试技术，旨在发现软件系统中的漏洞和错误。模糊测试通过向软件系统输入大量的随机数据，以检测系统对异常输入的处理能力，从而发现潜在的漏洞和错误。模糊测试已经成为软件测试领域中最常用的测试技术之一。模糊测试可以快速生成大量测试用例，对于发现简单的缺陷 基于深度学习的JavaScript引擎模糊测试方法研究　刘艺玮 -《电子科技大学硕士论文》- 2024-03-15（是否引证：否） 1.能显现。因此，想要产生合法的测试输入，不仅要考虑 Java Script 代码的语法正确性，还要考虑语义正确性。2.2 模糊测试理论模糊测试是一种自动化的软件测试技术，通过向软件系统输入大量异常或随机生成的数据，并监测软件对这些数据的处理反应，来识别潜在的错误、漏洞或异常行为。模糊测试通常会设计一种特殊的软件工具（也被称为模糊器）生成测试用例，将其输入到待测的软件程序并执行，在执行的过程中监视目标程序的运行状态。第二章 理论基础与关键技术17当被测的软件系统发生崩溃或出现异常行为时，测试人员记录下输入数据和运行结果，分析导致问题的原因，从而确定目标程序中是否存在漏洞。与其他漏洞检测方法相比，模糊测试高度自动化，不需要过多的领域知识，有良好的扩展性和应用性。模糊测试的核心在于利用算法生成的大量随机的测试用例，其中随机性意味着测试输入的多样性和不可预见性。因此，模糊测试能够探索软件行为的广泛空间，覆盖到软件开发人员可能没有考虑到的异常场景，发现传统测试手段忽略的缺陷和漏洞。图 2-3 模糊测试流程如图 2-3 所示，模糊测试包括以下四个阶段：生成测试用例、执行测试用例、异常监视、漏洞确认。（1）测试用例生成。模糊测试的输入用例应尽可能满足测试程序对输入格式的要求。即应保证输入用例在语法、语义上的正确性，以便通过解释器的检查 天然气管道分输站站控PLC程序的测试用例生成方法　陈安均 -《大学生论文联合比对库》- 2024-06-17（是否引证

		：否） 1.自动化生成测试用例的功能，其中，较为常见的测试方法有模糊测试和基于模型的测试。2.2.1 模糊测试的测试用例生成方法模糊测试是一种自动化的测试方法，即通过向系统输入大量随机、异常或非预期的数据，来发现软件系统中的潜在缺陷和安全漏洞。在进行模糊测试时，首先需要确定要进行模糊测试的目标，可以是整个系统、特定功能模块、接口或协议等。确保明确了测试的范围
2	此处有 227 字相似 反应。为了提高测试效率，该过程通常会自动化运行，通过在短时间内输入并执行大量的测试用例，从而快速发现可能存在的异常情况。 （3） 异常监视。测试用例执行过程中，Fuzzer 会监控目标程序的运行状态，记录异常行为，如程序崩溃或功能异常。当检测到异常时，相关测试用例会被标记并存储，以便后续分析。监控方式主要包括基于进程的监控和基于插桩的监控。进程监控方式下，Fuzzer作为父进程启动目标程序，并通过分析进程返回码和信号来判断是否发生异常。而插桩方法则在程序源码或二进制文件中嵌入额外代码，以在运行过程中收集执行信息，进一步分析程序行为。插桩可以分为源代码插桩和二进制插桩 ，其中二进制插桩适用于无源码的第三方程序。 （4）漏洞确认。在完成测试和异常监视后，研究人员需要对收集到的异常数据进	基于深度学习的JavaScript引擎模糊测试方法研究 刘艺玮 -《电子科技大学硕士论文》- 2024-03-15（是否引证：否） 1.作为输入数据，交给待测的目标程序执行。根据测试需要，这个步骤可能会自动化进行，以便在短时间内执行大量的测试用例。（3）异常监视。在执行测试用例的过程中，监控目标程序的行为和响应，收集任何异常执行情况的信息，如程序崩溃、功能错误。当监控到异常时，模糊器记录相应的测试用例以供后续的分析。监控的方式可以分为基于进程监控的方法和基于插桩的方法。在进程监控方式中，模糊测试器作为父进程，启动目标程序为子进程，控制其执行并监控其运行状况。模糊器根据子进程的返回码和信号判断是否出现异常。基于插桩的方法通过对目标程序和输入用例进行插桩来获取运行过程中电子科技大学硕士学位论文18的信息。插桩即在程序的源代码或二进制执行文件中插入额外的代码，以便在程序运行时收集信息。插桩也可以分为源代码插桩和二进制代码插桩。源代码插桩直接在源代码层面添加额外的代码行或命令来收集所需的信息。二进制插桩在编译后的二进制文件中插入监测或检测
3	此处有 69 字相似 擎。 3 基于类型推断的嵌入式JS引擎模糊测试方法 3.1 嵌入式JavaScript引擎模糊测试方法设计思路 在自动化的模糊测试中，验证测试方法有效性主要依赖两个关键因素：一方面是测试用例的生成质量，另一方面是缺陷检测方法的准确性。高质量的测试用例是发现潜在缺陷的基础，而高效的缺陷检测手段则决定了缺陷是否能够被及时且准确地识别。基于这一认识，本文提出了一种基于类型推断	面向嵌入式JavaScript引擎的差分模糊测试方法研究 姚厚友 -《西北大学硕士论文》- 2021-06-01（是否引证：否） 1. 第三章 以性能缺陷检测为导向的差分模糊测试方法研究17第三章 以性能缺陷检测为导向的差分模糊测试方法研究自动化测试中保证测试方法有效性的手段主要有两个，其一是测试用例生成质量，其二是判断是否是存在缺陷的缺陷检测方法。高质量的测试用例是触发缺陷的关键，缺陷检测方法决定了能否检测出缺陷。本章将针对嵌入式 Java Script 引擎的性能缺陷检测的
4	此处有 143 字相似 果传入错误的参数会导致触发类型错误而提前中止测试流程，导致代码覆盖率降低，因此参数类型推断是执行精准变异不可或缺的一步。 JavaScript中共有8种基本的数据类型，其中值类型有6种：字符串（String）、数字(Number)、布尔(Boolean)、空（Null）、未定义（Undefined）、Symbol。引用数据类型有3种：对象(Object)、数组(Array)、函数(Function)。 在类型推断模块中，通过静态文本分析推断参数可能的数据类型，以图3-3为例讲解。 图3-5 运行结果图 左侧的代码展示	JavaScript高级程序设计：2. - 《互联网文档资源（https://www.360docs.）》- （是否引证：否） 1.- 跨平台性：JavaScript可以在不同的操作系统和浏览器上运行。2.JavaScript中的数据类型有哪些？JavaScript中的数据类型包括：- 基本数据类型：数值（number）、字符串（string）、布尔值（boolean）、null和undefined。- 引用数据类型：对象（object）、数组（array）、函数（function）。3.JavaScript中的变量声明有哪些方式？JavaScript中的变量可以使用var、let或const进 170747008 赵晓蕊 信本171 赵小鹏 论文正文 赵晓蕊 -《大学生论文联合比对库》- 2021-04-30（是否引证：否） 1.和计算，常常用来为网页上添加各种不同的动态功能，主要被用来向HTML页面添加交互的行为，同时也可以直接加入HTML页面。值类型(基本类型)：字符串（String）、数字(Number)、布尔(Boolean)、对空（Null）、未定义（Undefined）、Symbol。引用数据类型有三种：对象(Object)、数组(Array)、函数

		(Function)。利用JAVA语言编写出来的程序代码在经过不断的编译过后会自动生成JAVA字节码，而JVM虚拟机又会自动对这些文字和字节码 六百音乐盒设计与实现　吴琪瑶 -《大学生论文联合比对库》- 2022-05-17（是否引证：否） 1.式各样的动态功能,为用户提供更流畅美观的浏览效果。通常JavaScript脚本是通过嵌入在HTML中来实现自身的功能的。JavaScript有值类型(基本类型)：字符串（String）、数字(Number)、布尔(Boolean)、对空（Null）、未定义（Undefined）、Symbol,大数值类型(BigInt)以及引用数据类型：对象(Object)、数组(Array)、函数(Function)、日期(Date)。3.4 开发工具Visual Studio Code是一个轻量且强大的跨平台开源代码编辑器（
5	此处有 37 字相似 循环执行多次变异策略，最后拼接这些变量，生成完整的变异代码并返回，最后去重测试用例列表，确保返回的测试用例唯一。 4 系统设计与实验评估 通过对嵌入式JavaScript引擎与相关测试方法的研究，本文旨在设计一种高效的差分模糊测试方法，并基于此方法实现了原型系统EJSFuzz。 4.1 系统设计概述 EJ	面向嵌入式JavaScript引擎的差分模糊测试方法研究　姚厚友 -《西北大学硕士论文》- 2021-06-01（是否引证：否） 1.发现缺陷的单位成本。 西北大学硕士专业学位论文40 第五章 原型系统设计与实验评估41第五章 原型系统设计与实验评估通过对嵌入式 Java Script 引擎和差分模糊测试方法的深入分析，本文引入了嵌入式 Java Script 引擎的差分模糊测试方法。前两章分别描述了嵌入式 Java Scr
6	此处有 35 字相似 出收集与对比等步骤。 图3-6 差分模糊测试流程 (4) 测试结果处理模块 测试结果处理模块包括测试结果精简与测试 结果过滤模块。经过差分模糊测试后，不同引擎持续输入的测试用例会产生大量测试结果，这些结果需经人工比对才能确认是否为未知缺陷。然而，人工分析时受重复测试结果的显著干扰，导致发现未知缺陷的成本大	面向嵌入式JavaScript引擎的差分模糊测试方法研究　姚厚友 -《西北大学硕士论文》- 2021-06-01（是否引证：否） 1.nt(a * b)5. print(a - b);6. print(a / b);图 1 测试用例示例2.3.2 测试结果过滤方法在差分模糊测试中，随机输入的测试用例会重复触发大量已经发现过的缺陷，重复产生的测试结果会影响测试人员从测试结果中寻找未知缺陷的效率。因此，将测试结果中重复的结果过
7	此处有 67 字相似 重难度大幅提高。为了解决这个问题，本文设计了一种过滤方法来降低重复结果的测试的影响。具体设计思路如下：通过提取异常信息和 执行结果中的引擎名称、异常 API 以及错误类型等内容，并对其进行标准化处理，可将其作为结果过滤的关键特征。基于这些特征对测试结果进行筛选：若当前结果与已有记录重复，则予以丢弃；否则，保留该结果，以便后续确认其是否触发了未知缺陷。(5) 系统界面设计	面向嵌入式JavaScript引擎的差分模糊测试方法研究　姚厚友 -《西北大学硕士论文》- 2021-06-01（是否引证：否） 1.要判断其测试结果是否改变。若测试结果发生改变，则恢复规范化前的测试用例。（4）从所有引擎执行规范化后的测试用例的执行结果中提取引擎名称，异常的API 以及关键的异常信息等内容，同时对异常信息进行标准化即可到用于测试结果过滤的关键特征。 （5）使用到这些关键特征对测试结果进行过滤，对比执行测试用例后引擎的异常状态，出错的 API 以及引擎抛出的异常信息等关键信息判断触发缺陷的测试结果是否重复
8	此处有 32 字相似 型等内容，并对其进行标准化处理，可将其作为结果过滤的关键特征。基于这些特征对测试结果进行筛选：若当前结果与已有记录重复， 则予以丢弃；否则，保留该结果，以便后续确认其是否触发了未知缺陷。 (5) 系统界面设计 图3-7 EJSFuzz系统主界面 打开EJSFuzz系统，首先会进入系统的主界面，如图3	面向嵌入式JavaScript引擎的差分模糊测试方法研究　姚厚友 -《西北大学硕士论文》- 2021-06-01（是否引证：否） 1.，出错的 API 以及引擎抛出的异常信息等关键信息判断触发缺陷的测试结果是否重复。若当前测试结果是重复发现的测试结果，则丢弃当前测试结果。否则记录当前测试结果以便手动确认其是否触发了未知缺陷。第四章 以高精度为导向的用例精简与结果过滤方法研究39 本章小结本章主要介绍了差分模糊测试结果自动化处理

3. 嵌入式JS引擎模糊测试方法研究.doc_第3部分

总字符数：8928

相似文献列表

去除本人文献复制比：5.1%(456)　　去除引用文献复制比：5.1%(456)　　文字复制比：5.1%(456)

1	软件工程-2019116022-杲时雨 软件工程 - 《大学生论文联合比对库》- 2023-05-06	3.0%（267） 是否引证：否
2	面向嵌入式JavaScript引擎的差分模糊测试方法研究 姚厚友(导师：牛进平;汤战勇) - 《西北大学硕士论文》- 2021-06-01	1.4%（125） 是否引证：否
3	李连胜 202006030237 赵新宇 一种基于参数变异的JavaScript引擎模糊测试框架 赵新宇 - 《大学生论文联合比对库》- 2024-05-21	0.7%（64） 是否引证：否

	原文内容	相似内容来源
1	此处有 26 字相似 方法 评估模糊测试包括多种指标，常见的有以下几种： 1) 代码覆盖率：代码覆盖率是衡量模糊测试有效性的关键指标之一， 指测试用例执行过程中覆盖的代码路径占总代码的百分比。 常见的覆盖率类型包括：语句覆盖率（测试是否执行了所有的代码语句）、分支覆盖率（测试是否覆盖了所有条件分支）、路径覆盖率（	李连胜 202006030237 赵新宇 一种基于参数变异的JavaScript引擎模糊测试框架 赵新宇 -《大学生论文联合比对库》- 2024-05-21（是否引证：否） 1.。重现成功后，还需进一步判断该crash是否可被利用。2.2.2 Fuzzing评价分析代码覆盖率：代码覆盖率是指在测试过程中被执行到的代码占总代码量的百分比。它是衡量测试充分性的一个重要指标。高代码覆盖率意味着测试数据能够触及更多的代码路径，从而增加发现潜在缺陷和漏洞的可能性。
2	此处有 168 字相似 糊器在单位时间内生成有效测试用例的能力，通常以（用例数/秒）衡量。高效的模糊器应具备高吞吐量、低冗余性等特点。 3) 漏洞发现率：漏洞发现率用于衡量软件测试中漏洞检测技术的有效性，表示特定技术或工具检测到的已知漏洞所占的百分比。简单来说，漏洞发现率是某一技术或工具检测到的漏洞数量与被测软件中已知漏洞总数的比值。例如，如果某系统存在50个已知漏洞，而某工具检测出其中10个，则该工具的漏洞发现率为20%。越高的漏洞发现率表明该检测工具越能够高效的识别漏洞 。 为了进一步说明本文所提出方法的有效性，需要引入对比实验，将EJSFuzz与其他模糊测试工具进行对比十分必要。本文选	软件工程-2019116022-杲时雨 软件工程 -《大学生论文联合比对库》- 2023-05-06（是否引证：否） 1.报告；④分析测试报告，评估JIT的漏洞和安全性。4.1.4评估指标模糊测试技术包含很多实验指标，例如：？漏洞发现率[30]：漏洞发现率是软件测试中用来衡量漏洞检测技术有效性的指标， 它是特定技术或工具检测到的已知漏洞的百分比。简单来说，漏洞发现率是一种技术或者工具可以检测到的漏洞数量与被测软件中存在的漏洞总数的比值。例如，某系统存在50个已知漏洞，而一个工具可以检测出其中20个已知漏洞，那么此工具的漏洞检测率为40%。高检测率代表着此检测工具能高效的识别漏洞，表明了它识别软件漏洞的能力。？漏洞类型[31]：在软件测试中，攻击者可以利用系统本身存在的弱点，针对于某一软件漏洞
3	此处有 58 字相似 tzero/fuzzilli 通过分析测试结果与对比实验，能准确评估EJSFuzz系统的模糊测试能力，并根据评估指标与 具体漏洞，改进原型系统，进一步提升其模糊测试的能力。 4.3 实验结果分析 本文根据“基于类型推断的差分模糊测试 方法”，对各大主流嵌入式JavaScript引擎进行模糊测试，以下从测试用例变异、模糊测试结果评估、对比实验结果等方面对	软件工程-2019116022-杲时雨 软件工程 -《大学生论文联合比对库》- 2023-05-06（是否引证：否） 1.g模糊测试技术，记录两者评估指标之间的差异。通过分析这些数据，本文将能够评估JVM内部JIT编译器的部分漏洞和实现差异，并针对具体漏洞，进一步改进差分模糊测试框架。4.2 实验结果分析本文通过使用“基于相关性矩阵的差分模糊测试框架”检验JIT模块，在JVM选项配置和测试用例代码特性的多种搭配情况下测试出各编译器的一些异常问题。以下从测试用例变异
4	此处有 38 字相似 9.84% 15.32% 20.54% 实验组 86.71% 84.04% 90.10% 93.65% 89.79% 实验结果如表4.3所示，由于对照组采用随机策略，导致其类型准确率普遍偏低， 范围为9.94%~21.28%。而在实验组中，各数据类型的准	李连胜 202006030237 赵新宇 一种基于参数变异的JavaScript引擎模糊测试框架 赵新宇 -《大学生论文联合比对库》- 2024-05-21（是否引证：否） 1.ring对照组 18.7% 7.5% 14.6% 22.4%实验组 95.7% 94.1% 92.3% 96.8%实验的结果如表5-3所示。由于对照组使用的是随机策略，不具有逻辑性，因此类型的精确率普遍偏低，由 7.5%到 22.4%. 使用了本文所提出的参数类型获取方法，实验组的类型精确率最高达到了96.8%. 在两组实验

	确率明显高于对照组，Array类型的准确率为4.07倍，Bo	
5	此处有 35 字相似 模糊测试系统的主要目的是有效地触发被测试引擎中的缺陷，本节使用EJSFuzz系统对4.1.1介绍的实验对象进行模糊测试， 通过分析EJSFuzz系统所检测到的引擎缺陷数量，来说明本文方法的有效 性。具体的触发情况如表4.4所示： 表4.4 引擎缺陷触发情况 引擎名称缺陷数量 JerryScript 2 M	面向嵌入式JavaScript引擎的差分模糊测试方法研究 姚厚友 -《西北大学硕士论文》- 2021-06-01（是否引证：否） 1.给定的关键信息而决定是否进行精简。 差分模糊测试效果评估缺陷检测系统的目标是对真实的软件系统进行缺陷检测。首先，通过分析 JSDiff系统所检测出的引擎缺陷数量及状态，说明本文提出的缺陷检测方法能有效检测出嵌入式 Java Script 引擎的缺陷。其次，还统计分析本文方法检测出的缺陷类型及数量，证明本文方法对
6	此处有 50 字相似 ，然而自增操作会影响左操作数num值变为2，而后进行减法运算得到错误结果1。正确的操作顺序为将左操作数num=1压入栈或 存入寄存器，而后对num变量执行赋值自增操作，此时对num的修改不会影响到已经存入栈或寄存器里的值， 且右操作数为先赋值后自增，它的值也为1。最后取出左右操作数和操作符，计算1-1得到正确结果0。 4.3.3 对比实验分	面向嵌入式JavaScript引擎的差分模糊测试方法研究 姚厚友 -《西北大学硕士论文》- 2021-06-01（是否引证：否） 1.因此 Duktape 执行测试用例后的输出为 0。然而，正确的执行顺序应该是将“>>>”左边变量的初始值 4 拷贝一份存入寄存器，再执行操作 a = 2，此时对变量 a 的修改结果不会影响运算符“>>>”左边已经存入寄存器的值，最终第 3 行演变为计算 4 >>> 2 的执行结果，因此正确的输出结果应该为 1。此测试用例中，无符号右移操作中加
7	此处有 41 字相似 大的开发者社区，正逐步向嵌入式领域扩展。这一趋势催生了众多优秀的嵌入式JavaScript引擎，随着各类引擎的广泛应用与 功能性的不断提升，嵌入式生态对JavaScript引擎安全性与可靠性的要求也在快速 提高，作为嵌入式生态的基础设施，JavaScript引擎的缺陷会导致大量的嵌入式设备面临风险，一个小的功能缺陷会让整个程	软件工程-2019116022-呆时雨 软件工程 -《大学生论文联合比对库》- 2023-05-06（是否引证：否） 1.VM）是目前软件工程领域一种常用的虚拟机，它是Java应用程序能正常开发、生产和发行的关键基础部件。然而随着软件复杂度和功能性的不断提升，业界对JVM的可靠性和安全性的要求也在快速提升。作为JVM中最重要的动态编译模块，JIT（Just-In-Time）编译器负担着越来越多代码优化、程序性能提升方面的工作
8	此处有 40 字相似 升，嵌入式生态对JavaScript引擎安全性与可靠性的要求也在快速提高，作为嵌入式生态的基础设施，JavaScript 引擎的缺陷会导致大量的嵌入式设备面临风险，一个小的功能缺陷会让整个程序运行错误， 性能缺陷会让本就资源受限的嵌入式设备难以正常工作，而安全缺陷甚至会成为物联网设备被攻破的入口。为了对嵌入式JavaScr	面向嵌入式JavaScript引擎的差分模糊测试方法研究 姚厚友 -《西北大学硕士论文》- 2021-06-01（是否引证：否） 1.擎。开发人员水平参差不齐等因素会导致嵌入式 Java Script 引擎存在安全缺陷、功能缺陷及性能缺陷等问题。引擎的安全缺陷会使嵌入式设备面临安全风险，引擎的功能缺陷会影响Java Script 程序的正确运行，引擎的性能缺陷不仅会白白增加计算资源匮乏的嵌入式设备的计算量，还会严重增加低功耗嵌入式设备的能耗。本文通过对现有缺陷检测方法进