

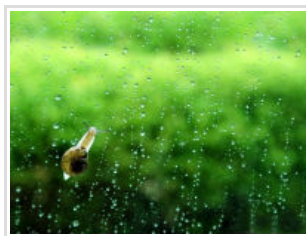
wuhui_gdnt的专栏

目录视图

摘要视图

RSS 订阅

个人资料



wuhui_gdnt

访问： 295371次

积分： 8541

等级： 

排名： 第769名

原创： 549篇 转载： 0篇

译文： 30篇 评论： 43条

[博客Markdown编辑器上线啦](#) [那些年我们追过的Wrox精品红皮计算机图书](#) [PMBOK第五版精讲视频教程](#) [火星敏捷开发1001问](#)

C、C++中未定义行为的指引， 第2部分

分类： C++语言

2013-03-20 14:25

595人阅读

评论(0)

收藏

举报

目录(?)

[+]

作者： John Regehr

原作： <http://blog.regehr.org/archives/226>

当像边界检查GCC，Purify，Valgrind这样的工具第一次出现时，在它们下面运行任意一个UNIX应用程序是有趣的。检查器的输出显示这些应用程序，尽管工作得很好，执行了大量的内存安全性错误，比如使用未初始化数据、数组访问越界等。仅运行grep或不管什么将导致数以十计或百计的这些错误发生。

发生了什么？基本上，C/UNIX执行环境的附带属性使得这些错误（通常）是温和的。例如，由malloc()返回的块通常在之前及/或之后包含一些填充字节；这些填充字节可以吸收越界储存，只要它们不是离分配区域太远。值得消除这些bug吗？是的。首先，一个带有不同属性的执行环境，比如一个提供减少填充字节的嵌入系统的malloc()，会把

文章搜索

文章分类

[ELF对线程局部储存的处理](#) (7)[GCC-3.4.6源代码学习笔记](#) (209)[GCC后端及汇编发布](#) (50)[Linux内核Modutils工具系列](#) (16)[Studying note of GCC-3.4.6 source](#) (202)[GCC's back-end & assemble emission](#) (17)[GRUB手册及Multiboot规范](#) (7)[autoconfig手册](#) (2)[dwarf调试格式](#) (2)[C++语言](#) (8)[LLVM](#) (22)

文章存档

[2015年01月](#) (3)[2014年11月](#) (3)[2014年10月](#) (3)[2014年09月](#) (3)[2014年07月](#) (4)[展开](#)

温和的近失 (near-miss) 数组写变为凶险的堆讹误bug。其次，在不同的情形下，同一个温和的bug甚至可能，在同一个执行环境里，导致一个崩溃或讹误。开发者通常发现这些类型的争论是令人信服的，目前大多数UNIX程序是相对Valgrind净化的。

用于查找整数未定义行为的工具不像内存不安全性检查器那么成熟。在C及C++中坏的整数行为包括有符号数溢出、除0、偏移超出比特宽度等。近年来，这些成为了更加严重的问题，因为：

- 整数缺陷是严重安全问题的一个来源
- 在利用整数未定义行为来产生高效代码方面，C编译器已经变得大为激进

最近我的学生Peng Li实现了一个整数未定义行为的检查工具。使用它，我们发现许多程序包含这些bug。例如，超过半数的SPECINT2006基准测试执行了或这或那的整数未定义行为。今天在许多方面，整数bug的情形与1995年左右内存bug的情形类似。说得更明确点，整数检查工具确实存在，但看起来它们没有广泛使用，而且它们中的许多工作在2机制文件上，太晚了。在编译器有机会利用未定义行为之前，你必须看一下源代码——然后消除它。

本贴的余下部分探讨我们在LLVM：一个中等大小 (~800KLOC) 的开源C++代码库中发现的几个整数未定义行为。当然我这里不是挑剔LLVM：它是质量非常高的代码。想法是通过看一下，在这个经过良好测试的代码中，未检测出的、潜藏的某些问题，我们可以有望学会在将来然后避免写出这些bug。

作为一个无目的的注解 (asa random note)，如果我们把LLVM代码视为C++0x而不是C++98，那么会出现大量额外的偏移相关的未定义行为。在后续贴中我将谈及新的偏移限制 (它等同于C99中的限制)。

我稍微整理了工具的输出以提高可读性。

整数溢出#1

错误消息：

UNDEFINED at <BitcodeWriter.cpp, (740:29)> :

阅读排行

[GCC-3.4.6源代码学习笔](#) (4730)[DWARF调试格式的简介](#) (2650)[GCC-3.4.6源代码学习笔](#) (1913)[DWARF调试格式的简介](#) (1858)[GCC-3.4.6源代码学习笔](#) (1788)[C++的词法闭包](#) (1469)[ELF对线程局部储存的处](#) (1389)[GCC后端及汇编发布 \(4](#) (1186)[C、C++中未定义行为的:](#) (1165)[Modutils工具源码分析之](#) (1163)

评论排行

[GCC-3.4.6源代码学习笔](#) (15)[Current content index of](#) (5)[GCC-3.4.6源代码学习笔](#) (4)[GCC后端及汇编发布 \(6](#) (3)[DWARF调试格式的简介](#) (2)[GCC-3.4.6源代码学习笔](#) (2)[GCC-3.4.6源代码学习笔](#) (2)[Studying note of GCC-3](#) (2)[Modutils工具源码分析之](#) (1)[GCC-3.4.6源代码学习笔](#) (1)

Operator: -

Reason: Signed Subtraction Overflow

left (int64): 0

right (int64): -9223372036854775808

代码:

```
int64_t V = IV->getSExtValue();
```

```
if (V >= 0)
```

```
    Record.push_back(V << 1);
```

```
else
```

```
    Record.push_back((-V << 1) | 1); <<----- bad line
```

在运行在2进制编码机器上的所有现代C/C++变种中，对值是INT_MIN（或在这个情形里，INT64_MIN）的int求负是未定义的行为。对这个情形，修改是添加一个显式的检查。

编译器会利用这个未定义的行为吗？它们会：

```
[regehr@gamow ~]$ cat negate.c
```

```
int foo (int x) __attribute__ ((noinline));
```

```
int foo (int x)
```

```
{
```

推荐文章

* CSS变量试玩儿

* **【Android开发经验】兼容不同的屏幕大小**

* Cocoa Core Competencies_1_Accessibility

* QtAndroid详解(3): startActivity 实战Android拍照功能

* PHPer都应该关注的服务端性能问题-听云Server试用笔记

最新评论

GCC后端及汇编发布 (6)

wuhui_gdnt: 你可以参考一些llvm的在线文档，llvm.org。在编译优化阶段，程序以中间形式出现，这时每条中间...

182. 结束语及预告

韩成涛: 楼主好毅力，佩服~

DWARF调试格式的简介 (续完)

brauceunix: 翻译得 得很好，很受用。谢谢。

GCC后端及汇编发布 (6)

wuhui_gdnt: delay slot跟流水线结构有关。跟在跳转指令后的指令就称为delay slot。因为执行跳转后...

GCC后端及汇编发布 (6)

Lazy_Linux: 请问楼主，define_split主要应用于什么情况？所谓的delay slot指的是什么？谢谢！...

Studying note of GCC-3.4.6 source code
chenleich: 有gcc3.4.6的软件

```
if (x < 0) x = -x;
```

```
return x >= 0;
```

```
}
```

```
#include <limits.h>
```

```
#include <stdio.h>
```

```
int main (void)
```

```
{
```

```
    printf ("%d\n", -INT_MIN);
```

```
    printf ("%d\n", foo(INT_MIN));
```

```
    return 0;
```

```
}
```

```
[regehr@gamow ~]$ gcc -O2 negate.c -o negate
```

```
negate.c: In function 'main':
```

```
negate.c:13:19: warning: integer overflow in expression [-Woverflow]
```

```
[regehr@gamow ~]$ ./negate
```

吗，或者arm-none-eabi的就是mingw软件

[Studying note of GCC-3.4.6 source code](#)
[chenleich](#): 为什么要学习源码呢？

[GCC-3.4.6源代码学习笔记](#) 当前
[yuanlinhu](#): 请问下 楼主 分析源码我想用vc++ 2008工具， 能不能把makefile文件 转换成...

[GCC-3.4.6源代码学习笔记 \(1\)](#)
[yuanlinhu](#): 谢谢 楼主分享 强烈支持， 希望楼主多多分享下底层的知识

[DWARF调试格式的简介](#)

[warriorpaw](#): @warriorpaw:不过这翻译， 额实在是，，，，， 有些地方需要对照原文才能明白是干啥的， 呵呵

C++

[C++ ABI summary](#)

[The C++ Standards Committee](#)

[C++ Standard Core Language Issue](#)

GCC

[GCC官方网站 \(GCC official site\)](#)

[Illumos 基于GCC的底层虚拟机 \(GCC-based low-level virtual machine\)](#)

[GCC Wiki](#)

-2147483648

1

在C编译器有矛盾的想法中，-INT_MIN即是负的又是非负的。如果第一个真正的AI（译注：人工智能）以C或C++编写，我想它会立即推导出自由即是奴役、爱即是恨、和平即是战争。

整数溢出#2

错误消息：

UNDEFINED at <InitPreprocessor.cpp, (173:39)> :

Operator: -

Reason: Signed Subtraction Overflow

left (int64): -9223372036854775808

right (int64): 1

代码：

```
MaxVal = (1LL << (TypeWidth - 1)) - 1;
```

在C/C++中，像这样计算最大有符号整数值是非法的。有更好的方式，比如创建一个全1的向量，然后清除高位的比特。

整数溢出#3

错误消息：

Linux

[Modutils v2.4下载点](#)**UNDEFINED at <TargetData.cpp, (629:28)> :****Operator: *****Reason: Signed Multiplication Overflow****left (int64): 142998016075267841****right (int64): 129**

代码:

Result += arrayIdx * (int64_t)getTypeAllocSize(Ty);

这里分配的大小是貌似合理的，但对于任何可想象的数组，数组索引超出了边界。

偏移超出比特宽度#1

错误消息:

UNDEFINED at <InstCombineCalls.cpp, (105:23)> :**Operator: <<****Reason: Unsigned Left Shift Error: Right operand is negative or is greater than or equal to the width of the promoted left operand****left (uint32): 1****right (uint32): 63**

代码:

```
unsigned Align = 1u << std::min(BitWidth - 1, TrailZ);
```

这完全是一个bug：BitWidth被设置为64，但应该是32。

偏移超出比特宽度#2

错误消息：

```
UNDEFINED at <Instructions.h, (233:15)> :
```

```
Operator: <<
```

```
Reason: Signed Left Shift Error: Right operand is negative or is greater than or equal  
to the width of the promoted left operand
```

```
left (int32): 1
```


```
right (int32): 32
```

代码：

```
return (1 << (getSubclassDataFromInstruction() >> 1)) >> 1;
```

当getSubclassDataFromInstruction()返回在范围128-131内的一个值时，左移的右实参值为32（译注：上面的代码应是getSubclassDataFromInstruction()>> 2）。（在任一方向）移动这个比特宽度或更大是一个错误，因此这个函数要求getSubclassDataFromInstruction()返回不大于127的值。

结论

使得某些程序行动错误，但没有给开发者任何方式来告知他们的代码是否执行这些行动，如果是在何处，基本上是邪恶的。C的设计点之一是“相信程序员  st and then there's trust）。

我的意思是，我信任我5岁的孩子，但
或保密性关键的代码，在编程方面等同

或C++创建一大段安全性关键



上一篇 C、C++中未定义行为的:

下一篇 C、C++中未定义行为的:

主题推荐

c++ 应用程序 人工智能 编译器 开发者

猜你在找

将 Win32 CC++ 应用程序迁移到 POWER 上的 Linux第 1
C++中的未定义行为undefined behavior
通过 Visual C++ 的编程模型和编译器优化增强您的应用
CC++开发者必不可少的15款编译器+IDE
您的应用程序架构中的Ext JS 4第2部分

将Win32CC++应用程序迁移到POWER上的Linux第3部分
C++中的未定义行为
CC++开发者必不可少的15款编译器+IDE
CC++开发者必不可少的15款编译器+IDE
我的应用程序崩溃现在怎么办 - 第2部分

准备好了么？跳 吧！

更多职位尽在 CSDN JOB

C/C++软件开发工程师	我要跳槽	高级C/C++开发工程师	我要跳槽
南京康瑞思信息技术有限公司	3-6K/月	陕西埃普沃企业管理咨询有限公司	10-15K/月
C/C++开发工程师	我要跳槽	C/C++高级软件工程师	我要跳槽
恒安嘉新（北京）科技有限公司	6-20K/月	同花顺	12-25K/月

监控系统下载

权限管理表设计

地税ca证书

4级准考证号

logo设计

变频器原理

电脑病毒下载

android教程

电路图仿真软件

验证码识别算法

语音广告制作

查看评论

暂无评论

您还没有登录,请[\[登录\]](#)或[\[注册\]](#)

* 以上用户言论只代表其个人观点，不代表CSDN网站的观点或立场

核心技术类目

全部主题 Hadoop AWS 移动游戏 Java Android iOS Swift 智能硬件 Docker OpenStack
VPN Spark ERP IE10 Eclipse CRM JavaScript 数据库 Ubuntu NFC WAP jQuery
BI HTML5 Spring Apache .NET API HTML SDK IIS Fedora XML LBS Unity
Splashtop UML components Windows Mobile Rails QEMU KDE Cassandra CloudStack
FTC coremail OPhone CouchBase 云计算 iOS6 Rackspace Web App SpringSide Maemo
Compuware 大数据 aptech Perl Tornado Ruby Hibernate ThinkPHP HBase Pure Solr
Angular Cloud Foundry Redis Scala Django Bootstrap

公司简介 | 招贤纳士 | 广告服务 | 银行汇款帐号 | 联系方式 | 版权声明 | 法律顾问 | 问题报告 | 合作伙伴 | 论坛反馈

网站客服

杂志客服

微博客服

webmaster@csdn.net

400-600-2320 | 北京创新乐知信息技术有限公司 版权所有 | 江苏乐知网络技术有限公司 提供商务支持

京 ICP 证 070598 号 | Copyright © 1999-2014, CSDN.NET, All Rights Reserved

