基于申威众核处理器的 Office 口令恢复技术

董本松[1,2],赵荣彩[1,2],张 恒[1,2]
(1. 中原工学院 计算机学院,河南 郑州 450007;
2. 中原工学院 前沿信息技术研究院,河南 郑州 450007)

摘 要:随着口令恢复的计算需求不断增加,传统的口令恢复工具不能满足实际的计算需求。基于神威 OpenACC 二次代码开发的优化策略,提出了较好的解决方案。采用申威众核处理器,对 Office 口令恢复程序进行移植和优化,充分利用众核处理器的从核计算性能和局部存储优势,在加速循环并行化、优化全局访存操作、提高数据传输效率三个方面进行了改进与优化,并通过实验对该方法的正确性、有效性进行了分析和验证。用 Office DOC 2007 的加密文档作为测试案例,经过多种数据规模测试,并与传统的口令恢复工具和开源的 Hashcat 口令恢复工具进行了实验对比。实验结果与性能分析表明,该方法能够较好地发挥申威众核处理器的优势,在申威众核处理器上实现 Office 口令恢复,具有良好的加速比,能有效满足实际的应用需求。

关键词:口令恢复;众核处理器;OpenACC;CPE;LDM
中图分类号:TP391 **文献标识码**:A **文章编号**:1673-629X(2021)05-0137-06
doi:10.3969/j.issn.1673-629X.2021.05.024

Office Password Recovery Technology Based on Sunway Many-core Processor

DONG Ben-song[1,2],ZHAO Rong-cai[1,2],ZHANG Heng[1,2]
(1. School of Computer Science,Zhongyuan University of Technology,Zhengzhou 450007,China;
2. Research Institute of Frontier Information Technology,Zhongyuan University of Technology,
Zhengzhou 450007,China)

Abstract:With the increasing demand of password recovery,the traditional password recovery tools cannot meet actual computing needs. Based on the optimization strategy of Sunway OpenACC secondary code development,a better solution is proposed. The Sunway many-core processor is applied to transplant and optimize the Office password recovery program. By making full use of the advantage of many-core processor in the performance of the kernel computation and local storage,improvements and optimizations are made in three aspects: accelerating loop parallelization,optimizing global memory access operations and improving data transmission efficiency. The correctness and effectiveness of the method are analyzed and verified through experiments. With the encrypted document of Office DOC 2007 as a test case,a variety of data scale tests and experimental comparisons with traditional password recovery tools and open-source Hashcat password recovery tools are carried out. The experimental results and performance analysis show that the proposed method can make use of the advantages of Sunway many-core processor and realize Office password recovery on Sunway many-core processor. It has well speedup and can effectively meet actual application needs.

Key words:password recovery;many-core processor;OpenACC;CPE;LDM

0 引 言

当前,很多数据的存储和传输都经过各种方式加密[1],使数据更加安全。但是,加密口令一旦丢失,将给用户带来巨大的损失;与此同时,敌对分子也会利用各种加密手段,隐藏不法行为信息,给安全部门及时获取有效情报带来了挑战。随着加密算法的进步以及密钥位数的不断增加,口令恢复的计算规模呈指数级增长[2]。针对大规模计算问题,需要一种新型的高性能口令恢复方案,以实现大规模计算资源的动态聚合,进而实现对密数据的高效分析与恢复。

国内外许多学者对 Office 口令恢复进行了大量的研究,开发了许多实用的恢复工具。AOPR(advanced

收稿日期:2020-07-20 **修回日期**:2020-11-23
基金项目:国家重点研发计划项目子课题(2016QY07X1503)
作者简介:董本松(1993-),男,硕士研究生,研究方向为高性能计算;赵荣彩,博士,教授,博导,研究方向为高性能计算、并行编译、反编译。

office password recovery)单机版是一款专业的 Office 口令恢复工具,在口令恢复过程中使用 CPU,但不支持 GPU 加速。EDPR(Elcomsoft distributed password recovery)是一款商用的口令恢复工具,支持 GPU 加速,但是具有平台局限性,仅能恢复 Windows 上的软件程序和文件。另外,从经济成本的角度来看,该软件完成一次恢复代价较高。Hashcat 自称是世界速度最快、完全免费且开源的分布式口令恢复工具,但并不支持在 SW26010 众核处理器上部署。如文献[3-9]所示,尽管学者们对 Office 口令恢复做了大量的研究工作,但是口令恢复在吞吐量、效率、开销和速度方面仍有改进的空间。

为了解决口令恢复的大规模计算问题,该文提出了基于申威众核处理器的 Office 口令恢复优化方法。通过实验对该方法的正确性、有效性进行了分析和验证。实验结果表明,该方法具有并行性强、恢复速度快、吞吐量大等特点。从个人安全层面来看,有效解决了因遗忘口令造成重要数据丢失的问题。从国家安全层面上来看,可以帮助安全部门从大规模数据中迅速获取有效信息,对危害民族团结和国家安全的行为进行严厉打击。

该文完成了以下工作:

(1)基于申威众核处理器的一个主核,对 Office 口令恢复程序进行了一系列的分析和测试,找出了 Office 口令恢复程序的性能瓶颈;

(2)结合申威众核处理器的特点,提出了三种 Office 口令恢复并行机制。如加速循环并行化、优化全局访存操作、提高数据传输效率;

(3)实现了申威众核处理器的并行 SW-Office 口令恢复并进行了全面评估。实验结果表明,基于申威众核处理器的 Office 口令恢复程序具有良好的加速比。

1 相关技术

1.1 申威众核处理器

与传统多核处理器相比,国产自主可控申威众核处理器具有不可替代的优势。SW26010 众核处理器采用具有自主知识产权的申威 64 位 RISC 核心指令集和片上异构融合架构,构建片上众核多维并行数据通信和层次化存储体系,有效地缓解了众核处理器“通信墙”和“存储器墙”两大难题[10]。主核和从核的频率均为 1.45 GHz,主核和从核均支持 256 位向量化指令,峰值性能每秒超过 3 万亿次双精度浮点运算,优异的性能优势为研发提供了硬件基础。

图 1 显示了 SW26010 异构众核处理器的体系结构。它由 4 个核组,共 260 个核心组成,其中每个核组包含一个运算控制核心(management processing element,MPE)和一个运算核心阵列(computer processing elements,CPEs)。运算核心阵列由 64 个运算核心组成,并且运算核心之间是独立的,每个运算核心都有一个独立且容量为 64 KB 的局部存储(local direct memory,LDM),LDM 与主存之间支持以直接内存访问(direct memory access,DMA)方式传输数据。运算控制核心作为通用处理器核心,负责分配任务,协调运算单元;64 个运算核心作为加速计算核心,用来加速代码中计算密集部分。此外,运算核心与运算核心之间支持以寄存器通信方式进行通信。通过合理的硬件结构布局,SW26010 众核处理器不仅能有效地执行各种哈希算法,还能有效解决并行处理系统设计中所遇到的存储、互连和任务调度等问题[11]。

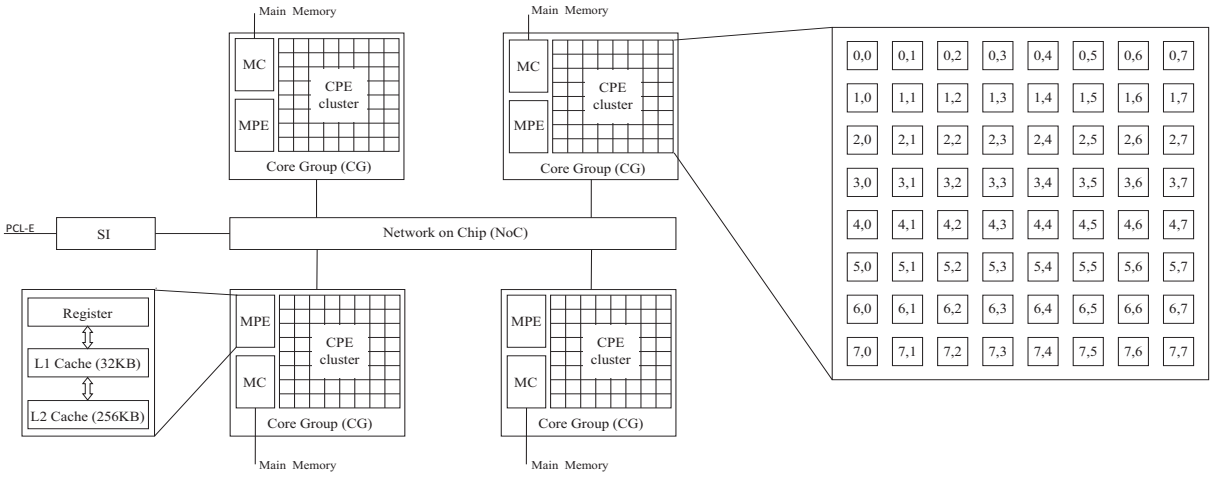

图 1 SW26010 异构众核处理器架构

1.2 Office 口令恢复的计算流程分析

该文以 Office DOC 2007 加密文档为例说明情况。口令恢复步骤主要包括 6 个模块,如 Office 文档解析、预处理、口令扩展、Hash 迭代、AES128、对比验证等。图 2 显示了口令恢复流程,对于口令恢复时间影响最大的是来自候选口令的散列计算,其计算量占申威小

型超级计算机的 96.7% 。

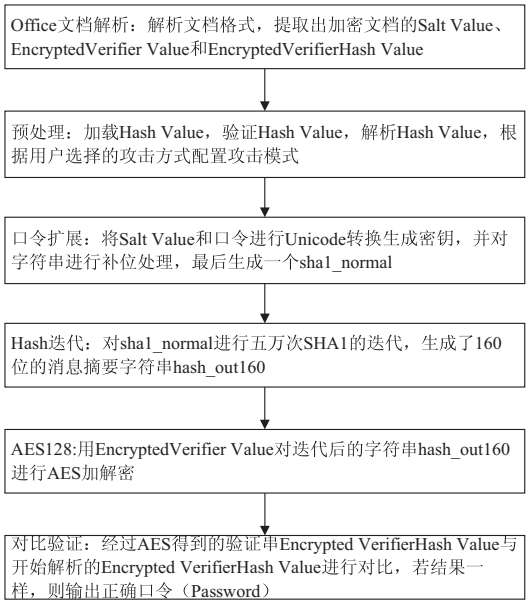

图 2　Office DOC 2007 加密文档口令恢复流程

Office 口令恢复程序的主要计算包含 Hash 迭代、AES128。Hash 的高次迭代是口令恢复的关键,主要通过 SHA1 高速迭代运算 5 万次,每一次迭代包含了 80 个子循环运算,最后生成 160 位的消息摘要。AES128 是以 Hash 高次迭代的结果和部分验证串作为输入,执行口令恢复操作,主要完成对字符串的拼接、比较、AES 加解密等运算。

1.3　Office 口令恢复的开销热点分析

该文首先在 SW26010 众核处理器的一个主核上移植实现了 Office 口令恢复程序。通过插桩方式对程序中各函数的运行时间进行测试和统计,找出运行时间位于前列的热点函数。缩短运行时间、提高运行速度是程序并行化的目的。程序中最耗时的是循环迭代计算部分,解决方法是将循环迭代步分布到多个不同的从核中同时进行计算[10],由主核负责分配任务,从核负责计算任务,合理地分配资源,使其效率最大化。

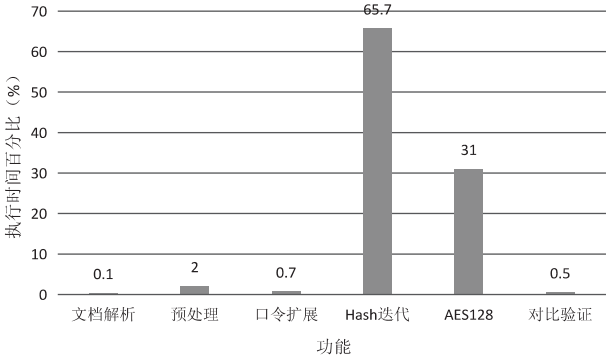

图 3　SW-Office 代码热点分析

如图 3 所示,Office 口令恢复程序的热点函数主要集中在 Hash 迭代、AES128 等模块。因此,该文分析了 Office 口令恢复程序的性能瓶颈,并基于国产自主可控平台,提出了加速循环并行化、优化全局访存操作、提高数据传输效率等优化方法。同时,结合申威众核处理器的高效并行执行能力,实现了并行 SW-Office 口令高效恢复。

2　基于申威众核处理器的 Office 口令恢复优化方法

为了解决 1.3 节中提出的性能瓶颈问题,该文采用申威 OpenACC 进行二次代码移植开发,充分利用申威平台的并行机制进行深度并行优化。以单核组版本为基础,分析验证优化方法的正确性和有效性。

2.1　加速循环并行化

根据 SW26010 众核处理器的体系结构设计,首先充分发挥 SW26010 从核的计算性能,将程序计算比较密集的部分映射到从核计算。申威 OpenACC 的研发是为了解决共享内存架构下的片内众核计算并行化问题[12]。申威 OpenACC 支持 gang、worker、vector 三级并行机制。gang 是粗粒度并行,worker 是细粒度并行,vector 是在 SIMD 或向量操作的指令级并行。通过对 Office 口令恢复程序的相关性分析和测试发现,该程序中存在循环嵌套函数,可利用申威 OpenACC 三级并行优化,但循环嵌套函数中存在直接和间接不可众核并行的部分。直接不可众核并行的部分是无法处理的,而间接不可众核并行的部分是可以通过技术处理的,使其能够成为众核并行的部分,因此在循环嵌套函数中,可直接循环并行化的部分相对较少。在程序设计过程中,如果程序完全依赖主核完成计算,在性能上将无法达到理想的效果,同时也严重浪费了 SW26010 从核的设计功能。因此,该文充分利用 SW26010 的从核计算机制,采用申威 OpenACC 三级并行优化,以主从加速并行模式执行程序,主核主要完成不可众核并行部分的计算以及通信,将开销较大的热点循环函数映射到从核加速并行计算,如算法 1 所示。

算法 1:加速循环并行化。

```
Input:PW 位数上限,字符种类上限。
Output:统计每一个字符出现的次数。
1. Begin
     //初始化 markov_stats_buf,获取第一个位置的字符值
2. #pragma acc parallel local(i,j,k)
3.   #pragma acc loop gang
4.    for i=0 to SP_PW_MAX by 1 do
5.       #pragma acc loop worker
6.          for j=0 to CHARSIZ by 1 do
7.             #pragma acc loop vector
8. for k=0 to CHARSIZ by 1 do
9. *out++ += *in++;
     //输出每一位字符出现的次数
```

```
10. end for
11.          end for
12.    end for
13. End
```

如算法 1 所示,通过使用申威 OpenACC 三级并行机制,加速循环并行化将开销较大的热点循环函数映射到 64 个计算从核上,每个从核由一个线程 worker 执行。实验结果表明,该方法以单核组为基础,验证了该方法的正确性和有效性,通过运行 64 个从核的循环并行函数,较 1 个主核性能提高了约 36 倍。然而,由于整个程序可直接循环并行化的部分相对较少,所以提升的效果并未呈现出 64 核的理想趋势,该实验的结果与主核相比,SW-Office 口令恢复整体性能提高了约 2.68 倍。

2.2 优化全局访存操作

利用 64 个计算从核后,充分利用从核 64 KB 局部存储,减少或避免从核直接访问主存。从核对主存的访问通常分为两种方式,一种是通过全局离散式(global load/store,gld/gst)访问主存,从核直接访问主存获取计算所需的数据,这种访问方式延迟很大,对程序的性能有一定的影响。另一种是通过 DMA 批量式访问主存,从核访问 LDM 的延迟很小,采用 DMA 传输方式,首先将从核计算所需的数据批量式地取到 LDM 中,当从核进行计算时,直接访问 LDM 获取所需的数据,从而大大减少数据的访存延迟,保证程序的性能。使用申威 OpenACC 制导语句访问主存时,都是通过 DMA 批量式访问方式,如果未指定制导语句,则默认采用全局离散访问方式。1 条 gld 指令的延迟约为 177 个周期,是 1 次访问便笺存储器(scratch pad memory,SPM)操作的 44 倍,属于低效的访存操作。另外,当 64 个从核同时发起 gld/gst 指令时,这些全局访存指令将排队依次执行[13],这会极大地降低并行效率,导致更多的资源浪费,因此需要尽量减少冗余的 gld/gst 指令。

在性能分析工具中,penv_slave2_gld_count 接口函数可以收集中间代码中的 gld/gst 指令数,帮助用户发现冗余指令。接口函数代码如下所示:

```
//主核代码部分,初始化接口
penv_slave2_gld_init();
⋮
//从核函数调用
⋮
//从核代码部分,统计接口
penv_slave2_gld_count(&ic1);
⋮
penv_slave2_gld_count(&ic2).
```

在使用了 penv_slave2_gld_count 接口函数后,在程序中发现冗余的 gld/gst 指令,主要是对保存在主存上的指针进行了直接的访问,从而导致程序的性能受到一定的影响。面对这种情况,首先在 SPM 上重新声明一个局存指针;然后在代码的初始化部分直接将地址赋值给 SPM 上的局存指针;最后,所有的指针操作直接访问 SPM,而不是通过 gld/gst 访问主存,从而避免了从核以 gld/gst 方式直接访问主存。在单核组条件下,该方法得到了有效的验证,在加速循环并行化的基础上,SW-Office 口令恢复整体性能进一步提高了约 1.73 倍。

2.3 提高数据传输效率

最后,通过提升 DMA 传输带宽来提高数据传输效率。对全局访存操作进行优化后,由于从核上的 LDM 容量有限,无法直接一次性将计算数据全部拷贝到加速线程 LDM 中存储,导致循环中涉及的数据往往无法批量式传输,因此只能按照单次循环涉及的计算数据进行传输,如果不能有效地优化数据传输[14],则无法提高数据传输效率。

根据 SW26010 众核处理器的体系结构设计,优化数据传输需要合理地利用 64 KB 的 LDM。LDM 的设计是为了减少访存的延迟,将计算过程中所涉及的数据传输至 LDM 的 SPM 上,通过 SPM 存储方式来减少 Cache 实现的控制开销,同时还避免众多运算核心间一致性处理带来的设计复杂性和性能下降[15]。为了解决因数据较大而无法一次性传输至 LDM 中存储的问题,该文利用申威 OpenACC 中的循环分块子句 tile。tile 制导语句将计算所需的数据按指定的块大小分割成两重循环,依次将从核计算中所需的数据传输至 LDM 中存储,方便从核计算时直接访问 LDM 获取数据,减少了从核直接访问主存的延迟,解决了因计算数据较大而无法直接拷入 LDM 中存储的问题,利用 DMA 实现批量式数据传输,从而提高数据的传输效率,如算法 2 所示:

算法 2:提高数据传输效率。

```
Input:任意长度的明文。
Output:拷贝到 SPM 中,计算输出 160 位的消息摘要。
1. Begin
2. #pragma acc parallel copy(inits, buffers) local(round, i)
     //对输入的任意明文进行分组,使得每一组的长度为 512
3. #pragma acc loop tile(512)
4. for block=0 to blockcount by 1 do
5. getwtschedule(&inputstring[block * 16],schedule);
     //将 16 份子明文分组扩展到 80 份
6. #pragma acc loop tile(32)
7. for round=0 to 80 by 1 do
8.          doRound(buffers,round,schedule[round]);
```

```
9. #pragma acc loop tile(32)
10. for i=0 to 5 by 1 do
11. buffers[i]+=inits[i]
12. inits[i]=buffers[i];
    //输出 160 位的消息摘要
13. end for
14.          end for
15.    end for
16. End
```

根据上述循环分块的原理[16],采用申威 OpenACC 的制导语句 tile 对程序中的计算数据进行循环分块。每一次 tile 分块的大小即从核计算所需的数据大小,通过 tile 子句循环分块后,将计算所需的数据提前存储在 LDM 中,方便从核计算时直接访问 LDM 获取数据。当 tile 尺寸增加时,执行时间缩短,性能得到优化。但是,由于 SPM 的空间容量有限,当计算 160 位消息摘要时,循环分块 tile 最大值为 512。在单核组下对该方法进行了正确性和有效性的验证,实验结果表明,SW-Office 口令恢复整体性能进一步提高了约 1.61 倍。

3　实验结果与性能分析

3.1　实验环境与测试程序

该文提供的实验平台是申威小型超级计算机系统,系统由 1 台管理节点计算机和 1 台双星服务器组成。其中双星服务器包含 1 颗 AST2400 作为 BMC(baseboard management controller)处理器,以及 1 颗 SW26010 申威众核处理器作为 1 个计算节点。实验环境具体如表 1 所示。

表 1　实验环境

名称	描述
服务器	申威小型超级计算机
操作系统	Deepin Linux Server 15.5
CPU	Intel(R) Core(TM) i5-7500 CPU @ 3.40 GHz 4 核
编程语言	C 语言
编译器	gcc
CPU 内存	8 GB
SW	SW26010 @1.45 GHz 260 核
编程语言	C 语言
编译器	swacc
SW 内存	32 GB
测试程序	SW-Office 口令恢复程序

3.2　实验结果与对比分析

以运行在 1 个主核上的 SW-Office 口令恢复程序作为基准测试,分别采用加速循环并行化、优化全局访存操作、提高数据传输效率三种优化方法,在单核组下对主从核进行性能测试对比。测试的加密文档选择 Office DOC 2007。每项测试实验做二十次,取平均值,保留整数。图 4 显示了性能测试的结果。

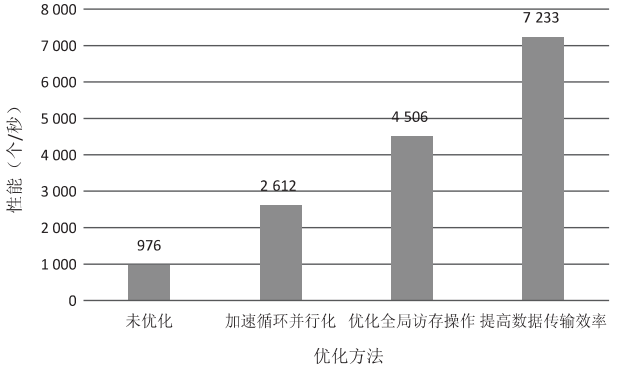

图 4　SW-Office 性能对比

如图 4 所示,基于 SW26010 众核处理器,通过加速循环并行化、优化全局访存操作、提高数据传输效率等三种方法对 Office 口令恢复进行了优化。实验结果表明,在加速循环并行化后,该程序的性能是原程序的约 2.68 倍;在优化全局访存操作后,性能是原程序的约 4.62 倍;在提高数据传输效率后,性能是原程序的约 7.41 倍。通过三种有效的方法优化,使得 SW-Office 口令恢复性能得到了一定的提高,同时验证了所提出的三种优化方法的正确性,从而进一步证明了三种优化方法对 Office 口令恢复程序的性能提高是有效的。

为了进一步评估优化方法的正确性和有效性,将与市场上的 Office 口令恢复软件进行性能对比分析。图 5 显示了 AOPR 单机版、EDPR、AccentSoft、Hashcat 和 SW-Office 口令恢复程序的性能,选择的测试文档为 Office DOC 2007。

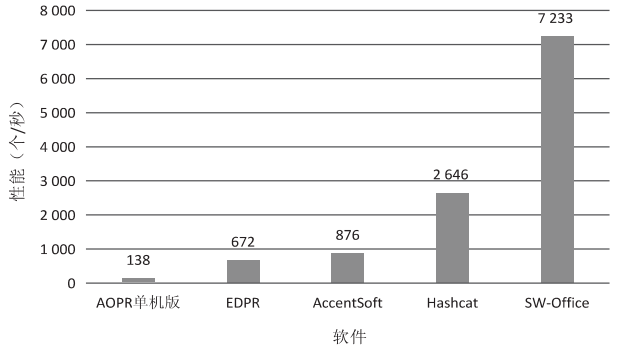

图 5　各软件的性能对比分析

如图 5 所示,使用不同的口令恢复软件对同一加密文档进行口令恢复时,SW-Office 的性能是 AOPR 单机版的约 52.41 倍,是 EDPR 的约 10.76 倍,是 AccentSoft 的约 8.26 倍,是 Hashcat 的约 2.73 倍。综上可得,基于 SW26010 异构众核处理器的 Office 口令恢复程序的性能表现较好,进一步证明了该优化方法

的正确性和有效性。

4 结束语

该文基于 SW26010 异构众核处理器实现了 Office 口令高效恢复。基于 SW26010 异构众核处理器,分析了 SW-Office 口令恢复程序,找出了 SW-Office 的热点函数,利用了申威众核处理器,提出了加速循环并行化、优化全局访存操作、提高数据传输效率等优化方法。解决了计算效率低、延迟长等问题,改进了细粒度的线程级并行。为了验证三种优化方法的正确性和有效性,首先对优化方法进行了实验测试,并对其性能进行了统计,得到的性能结果优于原程序。然后对优化后的性能与市场上同类软件的性能进行实验对比和分析,进一步验证了三种优化方法的正确性和有效性,从而使 SW-Office 口令恢复程序的性能与市场上同类软件相比有较好的性能改善,进而能够较好地满足实际的计算需求。

根据 Amdahl 定律可知,程序的加速比潜力取决于可以并行化部分的比例。目前只用了 1 个核组,验证了所提出方法的正确性和有效性。接下来将扩充到 4 个核组,同时,再进行向量化,充分挖掘代码的并行性,更有效地处理数据依赖性,充分发挥程序的加速比潜力,以尽可能发挥申威 26010 众核处理器的各种并行机制,最大限度地提升程序的执行效率。

参考文献:

[1] HENDI A Y, DWAIRI M O, AL-QADI Z A, et al. A novel simple and highly secure method for data encryption-decryption[J]. International Journal of Communication Networks and Information Security, 2019, 11(1): 232-238.

[2] TIWARI A, SHARMA N, KAUSHIK I, et al. Privacy issues & security techniques in big data[C]//2019 international conference on computing, communication, and intelligent systems (ICCCIS). Greater Noida, India: IEEE, 2019: 51-56.

[3] LI B, ZHOU Q, SI X. Mimic computing for password recovery [J]. Future Generation Computer Systems, 2018, 84: 58-77.

[4] QIU W, GONG Z, GUO Y, et al. GPU-based high performance password recovery technique for hash functions[J]. Journal of Information Science and Engineering, 2016, 32(1): 97-112.

[5] 张冬芳,管 磊,戴晓苗,等. 基于异构计算集群的密码口令破解系统设计与实现[J]. 网络空间安全,2019,10(6):95-101.

[6] 吕 辉. 基于 CUDA 架构的破解办公文件密码研究[D]. 郑州:解放军信息工程大学,2014.

[7] CHEN L, YANG Y, WANG J, et al. Word 2003 document password cracking based on the China supercomputer[C]// Proceedings of the 6th international Asia conference on industrial engineering and management innovation. Paris: Atlantis Press, 2016: 251-263.

[8] HONG J, CHEN Z, HU J. Analysis of encryption mechanism in office 2013[C]//2015 IEEE 9th international conference on anti-counterfeiting, security, and identification (ASID). Xiamen, China: IEEE, 2015: 29-32.

[9] 李丽平,周清雷,李 斌. 在多核 FPGA 上实现 Office 文档口令破解的方法[J]. 小型微型计算机系统,2019,40(5):929-934.

[10] 杨广文,赵文来,丁 楠,等. “神威·太湖之光”及其应用系统[J]. 科学,2017,69(3):12-16.

[11] DONGARRA J. Report on the Sunway Taihu light system: Uieecs-16-742[R]. Knoxville: University of Tennessee, 2016.

[12] 顾文静,孙 晨,王 彬. 基于 OpenACC 的高性能计算并行优化研究与应用[J]. 计算机技术与发展,2018,28(4):65-70.

[13] 王一超,林新华,蔡林金,等. 太湖之光上利用 OpenACC 移植和优化 GTC-P[J]. 计算机研究与发展,2018,55(4):875-884.

[14] 李雁冰,赵荣彩,韩 林,等. 一种面向异构众核处理器的并行编译框架[J]. 软件学报,2019,30(4):981-1001.

[15] 郑 方,许 勇,李宏亮,等. 一种面向高性能计算的自主众核处理器结构[J]. 中国科学:信息科学,2015,45(4):523-534.

[16] 李雁冰,赵荣彩,赵 博,等. 面向异构多核处理器的的循环分块[J]. 计算机工程与设计,2015,36(1):168-173.