基于不等长 counter 的存储器机密性和完整性保护方法

马海峰[1,2],姚念民[1],杜文杰[1]
(1.哈尔滨工程大学计算机科学与技术学院,黑龙江哈尔滨 150001;
2.黑龙江科技大学计算机与信息工程学院,黑龙江哈尔滨 150022)

摘 要: 针对计数器模式加密存在的 counter 存储开销大,容易溢出的问题,本文提出一种高效的数据机密性和完整性保护方法,它基于数据访问的局部特性,为内存访问频率不同的区域设置不同的 counter 长度,且 counter 长度可动态调整.分析和模拟实验表明,该方法可降低内存开销并减少溢出次数.

关键词: 计数器模式加密; 机密性; 完整性
中图分类号: TP393.08 **文献标识码:** A **文章编号:** 0372-2112 (2013) 12-2503-04
电子学报 URL: http://www.ejournal.org.cn **DOI:** 10.3969/j.issn.0372-2112.2013.12.029

Memory Confidentiality and Integrity Protection Method Based on Unequal Length Counter

MA Hai-feng[1,2], YAO Nian-min[1], DU Wen-jie[1]
(1. *College of Computer Science and Technology, Harbin Engineering University, Harbin, Heilongjiang* 150001, *China*;
2. *College of Computer and Information Engineering, Heilongjiang University of Science and Technology, Harbin, Heilongjiang* 150022, *China*)

Abstract: Focusing on the problem of high counter memory overhead and frequent overflow of counter mode encryption, this paper proposes an efficient scheme of protecting data confidentiality and integrity. Based on the locality character of data accessing, the scheme set different counter lengths for memory area of different accessing frequency and the counter lengths can be dynamic adjusted. Analysis and simulation results indicate the scheme can decrease memory overhead and the number of overflow.

Key words: counter mode encryption; confidentiality; integrity

1 引言

目前针对存储系统的攻击方式多种多样,主要分为软件攻击和硬件攻击,软件攻击是通过恶意软件、病毒等软件的手段来攻击系统,从而获得或破坏系统数据;另一类攻击是硬件攻击,它能直接修改物理操作,主要通过搭接设备的手段实施被动攻击和主动攻击.这种攻击已被证明是容易实施的.尽管有许多安全的软件包,但在防范此类攻击时难检测出.意识到这些威胁,提出了多种针对单处理器和多处理器的安全体系结构[1].一般在这类安全体系结构中,只有处理器片内是安全的,其它都是不安全的,其它部件都需要由处理器校验.

存储系统保护主要有两个方面:存储器完整性校验和加密.完整性校验(integrity verification)用于检测是否攻击者改变了正在运行的程序和数据的状态.完整性校验的主流方法是 hash 树[2],它的主要问题是存储开销和计算开销都很大,由此又提出了多种改进方法,有 CHTree[2]、LHash[2]、M-TREE[3]、PAT tree[4] 和 TEC Tree[5] 等.加密(Encryption)是保证内存中数据的私密性.存储加密的主流方法计数器模式(counter mode)加密.它的问题是存储开销大、容易溢出.针对此问题,本文提出一种内存机密性和完整性保护方法 MCIPIC(Memory Confidentiality and Integrity Protection based on Inequality Counter),它能降低保存 counter 的主存开销和 counter 溢出次数,并可用于所有基于 counter 的完整性和机密性保护方法中.

2 MCIPIC 原理和工作过程

本文提出的 MCIPIC 是经典计数器模式加密的改进方法,它根据内存访问频率来动态调整 counter 的长度.

收稿日期:2012-03-28;修回日期:2013-05-30;责任编辑:梅志强
基金项目:国家自然科学基金(No.61073047);黑龙江省教育厅科学技术研究项目(No.12533052);中央高校基本科研业务费专项资金(No.HEUCFT1202,No.HEUCF100609)

MCIPIC 主要通过三个过程实现:初始化、数据块加密解密和数据页迁移.为方便叙述,设 cache 行[6]和内存块大小相同.

2.1 初始化

MCIPIC 的原理如图 1 所示,上部是处理器,为安全区(trust domain),包含加密引擎(crypto engine)等部件,下部是内存,为不安全区(untrust domain).在初始阶段将内存划分为热区(hot area)和非热区(non-hot area),热区保存的是访问频率超过阈值的加密数据块,初始情况下热区不存放数据,除去热区外的其它区域为非热区,热区和非热区具有不同的密钥.再为内存中的每个页设置一个局部计数器,用来统计页内块的写回次数,非热区的计数器值(ctr)长度较小,热区的计数器值(ctr′)长度较大.另在处理器中维护一个结构,用来统计每个页内块的写回频率,并设置迁入阈值.当页内块每修改一次时,相应计数器值加 1,当计数器溢出时,更换密匙,并用新密匙重新加密页内所有数据块.

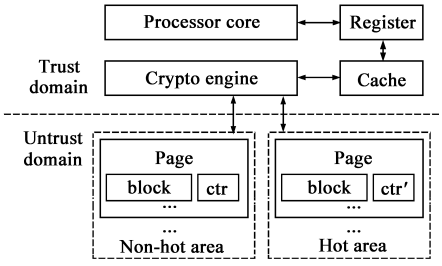


图1 MCIPIC原理图

需要说明的是,内存中加密块是连续存储的,ctr 存储在单独的加密块中,即 ctr 和加密块是分开存储的,因此没有语义问题.

2.2 数据块加密解密

当新产生的 cache 行写回内存时,要写到非热区;当修改后的数据块被写回内存时,要写到它读取的区.写到热区或非热区的步骤基本相同,区别在于计数器值(counter)长度不同.写回过程为:将对应 counter 加 1,将 counter 和块地址(addr)相连接,构成加密种子 seed,seed 具有唯一性,addr 保证不同的地址使用不同的 seed 加密;每次写回内存时计数器值加 1,保证写回同一地址时,seed 也唯一.接着用对应区的密匙和 seed 对 cache 行进行计数器模式加密,最后将加密的 cache 行和对应 counter 一起保存到内存.读取的过程是:先从内存热区或非热区读出加密的数据块和对应的 counter 到片内 cache,再将 counter 和 addr 相连接,构成解密种子 seed,接着用密匙和 seed 对加密数据块进行计数器模式解密,从而获得明文数据.

2.3 数据页迁移

数据页迁移有迁入到热区和迁出到非热区两个过程,简称迁入和迁出.当非热区中页的写回频率达到阈值时,要进行迁入,步骤为:将该页及对应的 counter 读入,用非热区密匙 non-hot-key 对页内数据块进行解密,接着热区的计数器 $\mathrm{Timer}_{\mathrm{hot}}$ 对这些数据块生成长度更长的新 counter,再用热区密匙 hot-key 对新 counter 和 addr 加密,最后将包含数据块和新 counter 的页映射到热区;如热区已满,在热区选择一个写回频率最低的页先迁出,再映射.迁出与迁入的步骤类似:将该页读入,用 hot-key 将其解密,然后用非热区的页计数器 $\mathrm{Timer}_{\mathrm{non\text{-}hot}}$ 对页内每个 cache 行生成一个长度更短的新 counter,再用 non-hot-key 对新 counter 和地址 addr 进行加密,最后将包含数据块和新 counter 的页映射到非热区.

3 MCIPIC 特点和存储增益分析

3.1 MCIPIC 的特点

相对于 counter mode,MCIPIC 有两个优点:一是降低了 counter 的存储开销,这是因为热区中 counter 较长,但热区占要保护内存的比例很小,非热区所占比例很大,因此总体来看 counter 占用的主存开销较小;二是减少了 counter 溢出的次数,这是由于程序的局部访问特性,在某段时间内,大部分的访问集中在热区,热区的局部 counter 增长很快,但 counter 足够长,不容易溢出;而非热区的数据块访问次数较少,因此虽 counter 较短也不容易溢出.

3.2 存储增益分析

下面分析 MCIPIC 的存储增益,比较的对象是 counter mode 加密.

分析存储增益需要的参数为:设内存总块数为 N,热区比例为 α,热区计数器值 counter 的长度为 L_{h},非热区 counter 的长度为 L_{n},数据块长度为 L,counter mode 的 counter 长度为 L_{c},参数值设为:$L_{\mathrm{n}}=16$ bits,$L=512$ bits,$L_{\mathrm{c}}=64$ bits,则 MCIPIC 的存储开销为:

$$\mathrm{Cost}=\alpha N\times(L_{\mathrm{h}}+L)+N(1-\alpha)\times(L_{\mathrm{n}}+L) \quad (1)$$

采用 counter mode 的存储开销为:

$$\mathrm{Cost}'=N\times(L_{\mathrm{c}}+L) \quad (2)$$

相对于 counter mode 的存储开销增益为:

$$\beta=(\mathrm{Cost}'-\mathrm{Cost})/\mathrm{Cost}' \quad (3)$$

将式(1)和式(2)代入式(3)中,代入参数值并化简,再分别设 L_{h} 12B、16B、24B、32B,得到存储性能增益如图 2 所示.

图中横坐标为热区比例 α,纵坐标为增益率 β.由图可见,当 L_{h} 不变时,随着 α 的逐渐提高,增益不断下降,这是因为热区的 counter 较长,热区增大导致总体占用的空间也增大;当 α 不变时,L_{h} 最小时增益率最大,随着 L_{h} 逐渐增大时,增益率下降,当热区的 counter 长

度超过一定程度时增益为负,这是因为当 L_h 较小时,热区占的存储开销也较小,增益明显,当 L_h 增大时,热区占的存储开销相应增大,导致总体存储开销增大,在某一时刻,将达到与 counter mode 相同的存储开销.

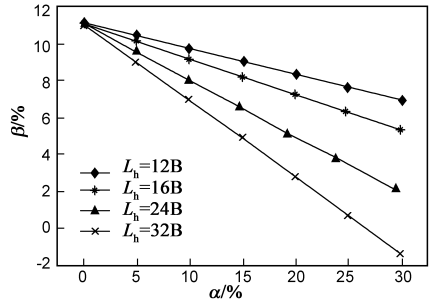


图2　存储性能增益

4　性能评估

下面评估 MCIPIC 的性能,将分为 counter 无溢出和有溢出两种情况讨论.

4.1　模拟环境和参数

仿真框架基于 Simplescalar tool set[7],该模拟器[1]能模拟分支预测和乱序执行的处理器.本文对 simplescalar 进行了修改,增加了 AES 加密机制、counter mode 和 MCIPIC 功能特性.实验基于 6 种 SPEC2000 CPU benchmarks[8]测试集:vortex、vpr、art、parser、mcf 和 gzip.为更准确捕捉程序特性,每个 benchmark 跳过了最初的 10 亿条指令,模拟执行 1 亿条指令,性能参数基于 IPC(Instruction Per Cycle).

4.2　无溢出时性能评估

首先评估在 counter 无溢出情况下 MCIPIC 的性能,比较的对象是基于 counter mode 和直接加密模式(Direct)的内存保护机制,评估基准是无加密保护的内存机制(Baseline).设 counter mode 中 counter 长度为 32 bits;MCIPIC 中非热区的 counter 长度为 8 bits,热区的 counter 长度为 64 bits;访问热区比例为 10%,非热区比例为 90%;MCIPIC 和 counter mode 在片内都有用于缓冲 counter 的 cache(seq cache),均设为 64K.

性能评估结果如图 3 所示,其中每个 benchmark 的 IPC 值都用 Baseline 进行了归一化.由图可见,不同加密模式性能降低程度不同:直接加密性能下降最多,counter mode 和 MCIPIC 性能下降较少.主要原因是 Direct 中加密解密延迟在关键路径上,无法隐藏这些延迟,从而造成性能下降;而 counter mode 和 MCIPIC 在片内 seq cache 中缓冲了部分 counter,在大部分情况下,解密时所需要的 counter 都能命中,这样就可直接从 seq cache 读取 counter 进行 AES 加密,同时从内存读加密块,这隐藏了解密延迟.在小部分情况下,解密时所需的 counter 未命中,这时就要先从内存取出 counter,再取数据块,解密延迟出现在关键路径上,但由于 counter 未命中数占总访问次数比例较少,因此性能下降较少.

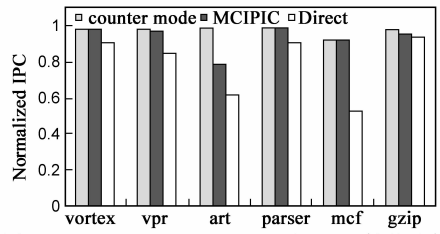


图3　Counter mode、MCIPIC和Direct性能比较

与 counter mode 相比,MCIPIC 的性能基本不变或有小幅下降,原因是 MCIPIC 在 counter mode 的加密基础上增加了换入和换出操作,即当非热区中某块访问频率到达阈值时,要换入到热区;当热区已满时,要选出访问频率最低的块换入到非热区,换入和换出都要进行 AES 操作,但由程序局部特性,当程序运行稳定后,换入和换出次数显著减少,对性能影响很小.

4.3　有溢出时性能评估

下面评估在 counter 发生溢出情况下 MCIPIC 的性能,比较的对象是 counter mode 和 Baseline,模拟程序基于 vortex 和 gzip.设 MCIPIC 中热区 counter 长度为 22 bits,非热区 counter 长度为 18 bits,counter mode 中 counter 长度设为 19 bits(与 counter mode 相比,MCIPIC 中 counter 占用的空间更小),访问热区比例为 10%,非热区比例为 90%,MCIPIC 中热区和非热区各有一个全局计数器.

程序的模拟结果如图 4(*a*)、(*b*)所示,图中横坐标 *t* 为时间(单位为秒),纵坐标为 IPC.由图可见,MCIPIC 的溢出次数小于 counter mode,这是因为非热区访问频率很低,即使非热区的 counter 长度相对较短,发生溢出的时间仍比 counter mode 长,即更少的溢出次数;而热区

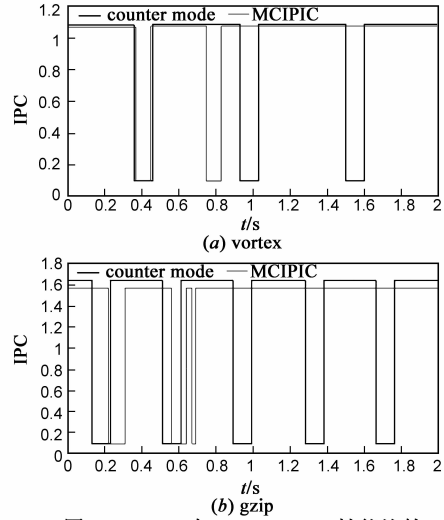


图4　MCIPIC与counter mode性能比较

counter 长度足够长,即使热区访问频率高,发生溢出的时间也比 counter mode 更长,溢出次数也更少.

另由图 4 可见,在发生溢出时,counter mode 的时间开销也比 MCIPIC 更大,这是由于 counter mode 溢出要更换密钥并重新加密所有要保护的内存区,这期间系统性能降到极低;而 MCIPIC 在溢出时只需加密溢出的热区或非热区,因此时间开销更少.

5 结论

本文提出了一种计数器模式加密的改进方法 MCIPIC,分析和模拟实验表明,其空间开销更低,溢出次数更少,可降低由内存保护引起的性能代价.下一步工作要研究 counter mode 加密在多处理器平台中的应用[9,10],并研究提高存储和计算性能优化的方法.

参考文献

[1] 管茂林,等.流体系结构指令存储器优化设计研究[J].电子学报,2012,40(7):1379 – 1385.
GUAN Mao-lin, et al. Optimized design research of instruction memory for stream architecture[J]. Acta Electronica Sinica, 2012,40(7):1379 – 1385.(in Chinese)

[2] G E Suh, D Clarke, B Gassend, M van Dijk, S Devadas. AEGIS: Architecture for tamper-evident and tamper resistant processing[A]. The 17th Annual International Conference on Supercomputing[C]. New York, USA: ACM Press, 2003. 160 – 171.

[3] Chenghuai Lu, et al. M-Tree. A high efficiency security architecture for protecting integrity and privacy of software[J]. Journal of Parallel and Distributed Computing, 2006, 66(9): 1116 – 1128.

[4] Hall, W E, Jutla, C S. Parallelizable authentication trees[A]. Lecture Notes in Computer Science[C]. Kingston, Canada: Springer Berlin Heidelberg Press, 2006. 95 – 109.

[5] Reouven Elbaz, et al. TEC-Tree. A low cost and parallelizable tree for efficient defense against memory replay attacks[A]. Lecture Notes in Computer Science[C]. Vienna, Austria: Springer Berlin Heidelberg Press, 2007. 289 – 302.

[6] 王超,等.异质存储系统中的高速缓存机制研究[J].电子学报,2011,39(6):1267 – 1271.
WANG Chao, et al. A study on cache mechanism in heterogeneous memory system[J]. Acta Electronica Sinica, 2011, 39(6):1267 – 1271.

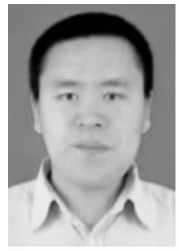
[7] Todd Austin, Eric Larson, Dan Ernst. SimpleScalar: An infrastructure for computer system modeling[J]. Computer, 2002, 35(2):59 – 67.

[8] Hussein Al-Zoubi, et al. Performance evaluation of cache replacement policies for the SPEC CPU2000 benchmark suite [A]. The 42nd ACM Southeast Regional Conference[C]. New York, USA: ACM Press, 2004. 267 – 272.

[9] 祝永志,等.基于 SMP 机群的层次化并行编程技术的研究[J].电子学报,2012,40(11):2207 – 2210.
ZHU Yong-zhi, et al. Research of parallel programming techniques of hierarchical model based on SMP clusters[J]. Acta Electronica Sinica, 2012, 40(11):2207 – 2210.

[10] Brian Rogers, et al. Single-level integrity and confidentiality protection for distributed shared memory multiprocessors[A]. International Symposium on Computer Architecture[C]. Salt Lake City, UT: IEEE Computer Society, 2008. 161 – 172.

作者简介



马海峰 男,1977 年出生,黑龙江鸡西人,博士研究生,黑龙江科技大学副教授,主要研究方向为存储安全,网络安全。

姚念民 男,1974 年出生,黑龙江大庆人,哈尔滨工程大学教授,博士生导师,主要研究方向为网络安全、网络存储、性能分析.

杜文杰 男,1986 年出生,硕士研究生,湖北黄冈市人,主要研究方向为数据存储.