

反病毒引擎硬件加速技术研究

肖梓航, 桑胜田, 肖新光

(安天实验室, 黑龙江哈尔滨 150020)

摘要: 本文首先指出软件型反病毒引擎无法满足高速网络数据处理需求的现状, 分析了硬件加速的历史和特点, 进而分别总结了通过 ASIC 和 GPU 两种硬件实现反病毒引擎加速的研究现状, 讨论了面临的问题。同时, 介绍了安天实验室在这两个领域中已取得的研究进展, 并分析了主要实现思路。

关键词: 反病毒引擎; 硬件加速; ASIC; GPU; FPGA

中图分类号: TP393.08 **文献标识码:** A **文章编号:** 1671-1122 (2011) 01-0036-04

Research on Hardware Speedup of Antivirus Engine

XIAO Zi-hang, SANG Sheng-tian, XIAO Xin-guang

(Antiy Labs, Ha'erbin Heilongjiang 150020, China)

Abstract: In this paper, firstly we point out that software-type antivirus engine can't supply the demand of nowadays high-speed network data processing. Then we introduce the history of hardware speedup technology and analyze its characteristics. Next we summarize research status of the ASIC and GPU speedup method in antivirus engine, and discussed existing problems. Lastly we present Antiy's research results in these two fields and explained the central implementation method.

Key words: antivirus engine; hardware speedup; ASIC; GPU; FPGA

0 反病毒引擎现状

反病毒技术是信息安全产业的核心技术, 反病毒引擎则是反病毒技术这个核心中的核心。

传统的反病毒引擎一般是软件型引擎^[1], 构建在通用处理器和体系结构之上, 通过高级语言程序实现引擎逻辑和数据处理。软件型引擎的优点在于: 1) 通用性和普适性: 基于 X86 处理器和体系结构的软件型引擎, 可以衍生出个人杀毒软件、企业级安全软件等, 运行在绝大部分个人计算机之上; 2) 设计和实现的成本不高: 引擎的设计和实现都以编码的工作形式体现, 无论是人力成本和设备成本, 还是后期维护(需求变更、Bug 修复、版本升级等), 需要投入的时间和资金相对都不多。

因此, 软件型引擎具有较低的准入门槛, 被各类安全厂商广泛使用^[2]。

但随着信息技术的飞速发展, 软件型引擎也开始暴露出不足的一面, 即处理速度。

互联网及其承载的信息量爆炸增长, 带来了海量的数据。从个人用户的角度, 一个反病毒引擎能在几个小时扫描 1TB (1024GB) 的文件, 就已经可以满足; 但站在企业和网络运营商的角度, 核心网和骨干网的吞吐量以 10Gbps 甚至更高的速率为基本单位, 这就对运行在网络设备中的反病毒引擎处理速度提出了更高的要求。

然而, 基于通用处理架构的软件型反病毒引擎难以达到这样的超高处理速度。一方面, 软件在从高级语义逻辑执行到物理元器件状态变化的等价转换过程中产生了大量的非最优指令, 导致多余的时间开销; 另一方面, 通用型处理器在硬件设计上主要考虑对单一指令流的分支预测、对存储器访问的缓存加速等问题, 因而较适合一般的逻辑事务处理, 而不适合大规模高并发的计算。

此外, 目前受限于功耗、散热、信号稳定性等诸多问题, 通用处理器中单个处理单元的计算能力已经接近物理极限, 摩尔定律在单一处理单元上已经难以生效, 各大处理器厂商开始走上多处理单元(即多核)的道路。无论是多核并行, 还是多处理器并行, 在反病毒引擎的处理速度上实际上依然存在上面所说的两个问题, 而且传统架构处理单元的并行还存在较大的同步、通信开销, 硬件成本优势也将随着并行度的提高而逐渐消失。

● 收稿时间: 2010-12-23

作者简介: 肖梓航 (1988-), 男, 湖南, 副总工程师, 本科, 主要研究方向: 安全事件应急处理技术、病毒分析与检测、反病毒引擎技术; 桑胜田 (1976-), 男, 黑龙江, 主任, 讲师, 博士, 主要研究方向: 微电子技术、反病毒引擎家族技术、安全技术与硬件结合技术研究; 肖新光 (1974-), 男, 吉林, 首席技术架构师, 高工, 主要研究方向: 反病毒引擎、反病毒体系架构、计算机犯罪取证。

面对这样的困境,人们开始将目光投向硬件加速技术。

在大规模网络数据处理的应用中,硬件加速已经取得了极为显著的成效。事实上,目前 Internet 上的网络流量所流经的路由设备和交换设备,绝大部分都不是基于通用硬件架构。所谓的网络处理器(Network Processor, NP)即是这种技术的代表,它是一种专门为处理数据包而设计的可编程处理器,能够直接完成简单的网络数据分析转发,并通过微指令编程和并行运行实现高速的高层应用^[3]。

事实上,在路由器^[4, 5]、防火墙、IDS(Intrusion Detection Systems, 入侵检测系统)^[6, 7]、IPS(Intrusion Prevention System, 入侵防护系统)、UTM Unified Threat Management(统一威胁管理)等网络和信息安全领域,硬件加速已经被广泛使用并取得相当好的效果,成为中高端市场产品的主流技术。

在与反病毒引擎在技术特点上比较接近的 IDS 等领域,硬件加速的主要问题是高速模式匹配,尤其是正则匹配(及其衍生的深度包检测技术)。这类问题涉及复杂的程序逻辑和数据读写操作,并在综合特征检测、异常检测、DoS 检测、溢出检测中大量使用,因此是整体性能的瓶颈所在^[7]。近几年来,模式匹配的硬件加速问题成为学术界和产业界共同的研究热点,各种场景模型、匹配算法和加速技术下的优秀结果层出不穷^[8-16]。

但是这些技术只能供反病毒引擎借鉴,而无法直接使用。与上述应用相比,反病毒技术有更为复杂的处理流程、几十倍的规则库、繁琐的预处理过程、较低的误报要求^[17]。因此,一直以来,反病毒引擎的硬件加速都是一个棘手的问题。

无论如何,从整个产业界的实际需求和以往的历史发展规律来看,我们可以肯定硬件加速将会是反病毒引擎的性能发力点和主要发展方向之一。随着专用硬件本身能力的不断增长,反病毒技术将逐步满足现有需求的要求,并因为高速而开拓出更广阔的新应用和新市场。

1 基于 ASIC 的硬件加速

根据使用的硬件是否专门为应用场景设计,可以将硬件加速技术粗略地分为两种:基于专用芯片的加速和基于通用硬件的加速。

专用芯片一般是指 ASIC(Application Specific Integrated Circuit)^[18],即专门为某一应用而设计的集成电路。基于 ASIC 的加速技术,需要深入分析应用的需求、自行设计和制造集成电路,然后用于嵌入式设备之中。与通用芯片相比,具有体积小、功耗低、可靠性高、保密性高等优点,最关键的是通过硬件单元直接处理流程逻辑,使性能大幅提高。

然而,ASIC 的不足之处也很明显,一是设计成本高:即便有所谓半定制设计,即使用标准的基本处理组件,但要实

现反病毒引擎复杂逻辑,难度依然很大;二是难以升级维护,设计一旦完成,在原有产品上完全无法实现功能上的修改。

因此,一种折中方案出现,即可编程 ASIC,其中的典型代表是 FPGA(Field-Programmable Gate Array, 现场可编程门阵列)^[18],它是一种半可定制 ASIC,通过硬件描述语言(Verilog 或 VHDL)完成电路设计,然后烧录到 FPGA 芯片上,从而获得需要的芯片。

与传统 ASIC 相比,FPGA 的处理速度会有所减慢,并且难以完成复杂的逻辑设计,还消耗更多的电能、具有较高的造价,但它具有相对较低的时间成本和设计成本,并且拥有一定意义上的可维护性,因此更为适合于反病毒引擎^[4, 11]。

基于 ASIC 的反病毒引擎加速技术主要有两方面的思路:

1) 实现网络数据包的处理。要实现深度网络数据分析和检测,就需要对数据包做预处理,包括荷载提取、流还原、数据重组、任务分拣等,这些操作在高速网络环境下具有高度复杂性,没有任何一款智能网卡可以实现上述功能,而通用处理器无法实现较高的速度,因此考虑用 ASIC 完成。

2) 实现病毒检测。病毒检测的主要时间开销发生在哈希计算和模式匹配。在对这两个模块做细化分析后,发现其主要问题出现在存储器读写、任务并发数量、关键算法的执行等方面,可以通过 ASIC 实现专门的处理,超越 CPU 的性能。

安天实验室综合考虑了上述因素,对两个思路均进行了尝试。安天正在设计的 AVL SDK Scope 计划采用 ASIC 处理网络数据包,并处理大量的常规病毒检测工作,而将待检测数据中非常规的部分(异常数据),以及逻辑复杂的预处理部分交由主处理器计算,因此主处理器不再成为性能瓶颈。此外,AVL SDK Scope 突出考虑任务的高并发性,在芯片设计上,实现了多个并行处理单元,并且可以实现芯片级的并行性多处理,即多芯片组合成芯片阵列,因此具备了较高的处理能力可定制性。

AVL SDK Scope 芯片内部结构如图 1 所示,其核心是四个同样的微引擎单元(MicroEngine, ME),负责包解析、重组,规则匹配;控制单元(Control Unit)负责整体的协调、对 ME 的配置和与片外主处理器通讯;填充单元(Fill Unit)将输入界面到来的包预处理后按 ME 所需的内部数据结构提交智能缓存(IntelliBuffer)。

智能缓存是 AVL SDK Scope 的获得高性能的关键技术之一,它的工作原理基于对这样一种事实的观察,即对于网络安全应用,流量中绝大多数数据是良性数据,仅有少数需要深度分析。智能缓存采用多哈希算法,过滤掉 90% 以上的正常数据,仅将不到 10% 的“疑似”问题数据保存至缓冲区。

保存至缓冲区的数据等待 ME 处理,必要时缓存内容可暂存于片外高速缓存。四个 ME 并行工作,按照配置的规则从缓存中取出重组后的内容与匹配库比较,根据匹配结果,ME

修改并填写数据结构中相应的域。提取单元 (Extraction Unit) 从缓存中提取具有特定标志的包, 简单处理还原后发送到输出界面。控制单元监测微引擎单元、填充单元、提取单元的状态, 这些状态信息可通过主处理器界面反映给主处理器。ME 处理过程中遇到需要特殊处理的情况可将数据写入片外 SDRAM, 并通过控制单元通知主处理器做进一步分析。

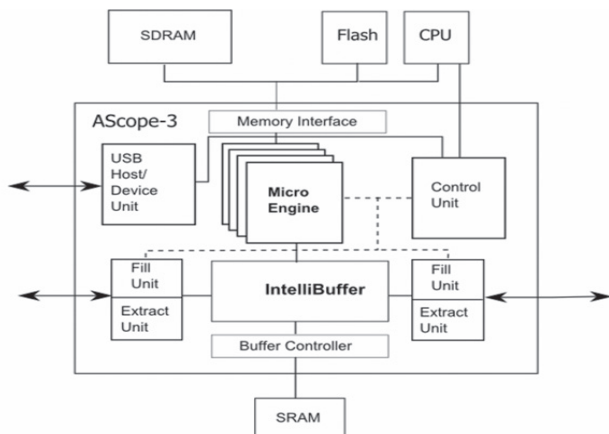


图1 AVL SDK Scope芯片内部结构

在 20 万条规则下, 单片 AVL SDK Scope 的设计能力为, 多片集成则可以基本满足企业网和核心网的检测需求, 适用于与非 X86 的 ARM、MIPS 等架构结合, 实现对低端网络设备的病毒加速过滤。目前该研究计划正在进行中。

2 基于 GPU 的硬件加速

基于通用硬件的加速技术在早年一般是指基于网络处理器或 FPGA 的加速, 但从目前的趋势来看, 在现在和未来相当长一段时间, GPU 加速将成为发力点和主要方向^[19]。

2.1 GPU 加速的优点

近年来, 由于对图形图像处理的需求逐渐扩大, 用于图像技术的专门处理器 GPU (Graphics Processing Units, 图形处理器) 呈现出极快的发展, 出现了用于通用高性能计算的 GPU 计算模式。相对于 CPU, GPU 具有更多的并行处理单元、更高的存储器访问带宽和更强的浮点运算能力。从技术架构上来看, CPU 具有大量 Cache 存储器和逻辑控制单元, 适合于逻辑控制和一般计算任务; 而 GPU 则包含数以千计的并发执行单元和少量控制单元, 借以实现超高性能的图形计算和大规模并行通用计算。因此, CPU+GPU 的运算结构应运而生, 由 CPU 负责逻辑控制和任务协调、GPU 负责核心计算的方式, 成为目前高性能计算领域的主要计算模式。通用 GPU 计算逐渐走向实用, 在气候预测、石油勘测、航空航天、制药、海洋环境、金融工程、地震预报等领域都取得了令人瞩目的成果^[20]。

和 ASIC 加速相比, GPU 加速几乎就是一种软件性加速,

它充分依托硬件架构上的高度并行处理能力, 通过与体系结构相关的高级语言程序开发实现加速, 具有以下优点:^[21-22]

1) 实现成本低: GPU 加速回归了高级语言设计开发的方式, 技术上的准入门槛也低; 2) 开发周期短: 无论是从头开始设计还是现有代码移植, GPU 都具有较快的开发周期, 新功能的迭代周期则更比 ASIC 要短; 3) 可移植性好: 由于 CPU+GPU 的架构, 硬件兼容性变得宽松, 而 GPU 计算的目标代码通过显卡设备驱动实现物理无关的封装, 因此代码移植性也得到保证; 4) 可扩展性好: 硬件上, 从最简单的单 CPU+单 GPU, 到数百 CPU 与数百 GPU 的超级计算集群, 都易于实现; 软件上, 功能的扩展可以简单地通过编码实现。

2.2 GPU 加速需突破的技术

网络安全方面, 目前也可实际使用 GPU 加速的成熟产品还很少。在学术界这是目前研究的一个重点, 基于开源软件 Snort 的 GPU 加速 IDS^[6, 23]、基于开源软件 ClamAV 的 GPU 加速反病毒引擎^[17], 在国外的学术期刊和学术会议上屡屡出现, 其结果往往较普通 CPU 计算有十余倍的性能提升。

然而, 这些结果依然难以应用到实际场景。一部分原因是源于原型系统与实际的差异, 例如规则库的规模, 实际场景中的情况将导致理论方法的结果呈现雪崩式的下降。

安天实验室根据对反病毒引擎的理解, 对 GPU 加速反病毒引擎进行了多方位的实验和研究, 认为主要需要突破的技术有以下几部分。

1) 网络数据包处理。对高速处理引擎而言, 无论硬件架构如何, 这始终是需要面对的问题。与网卡同在 PCI-E 总线上的 GPU 设备并不擅长于处理复杂的网络数据包, 在 GPU 加速模式下, 该项工作应通过修改网卡驱动来实现底层数据包的重组等工作^[7]。

2) 数据拷贝。GPU 计算中, 要考虑数据在网卡缓存、内存、显存中的存留, 以及通过 PCI-E 总线在三者之间的拷贝。其中主要的瓶颈出现在 PCI-E 总线带宽、内存读写速率。对前者, 可以通过流计算模型实现总线拷贝与 GPU 计算的拷贝^[24], 从而使总线全时间使用; 对后者, 应固定分配非换页内存, 消除内存换页对 DMA 速率的影响^[21]。

3) 预处理过滤。为了减少数据拷贝, 应该建立简单的预处理过滤方法, 尽量避免待检测数据被拷贝到显存中^[17]。预处理过滤没有统一的方法, 但需和反病毒引擎的核心流程紧密结合。传统的软件引擎可能不会含有预处理过滤这一步骤, 这就需要从头设计。

4) CPU 和 GPU 任务的划分。前面指出, CPU 适合处理逻辑复杂的计算, 而 GPU 适合处理逻辑简单、能高度并行的计算。在反病毒引擎中, 哈希计算、模式匹配适合 GPU 来完成, 而各环节的衔接调度、预处理、解包等操作, 还是适合 CPU 执行。

5) 多线程的合理调度。注意 CPU 与 GPU 可以并发工作, 在 CPU 端有线程的调度, 在 GPU 端有流模型和上下文调度, 合理安排 CPU 线程、GPU 流、GPU 内核函数, 可以使整个计算系统得到充分利用。

6) GPU 优化^[25]。GPU 计算存在特有的并行模型和存储模型, 在这些模型下进行硬件层面的算法优化和代码优化, 是有一定必要的。

目前的学术研究大都将注意力集中到 GPU 优化之上, 而用 GPU 加速反病毒引擎将是一个全局性的系统工程, 上述每一点都应考虑。

在上述优化思路下, 安天实验室实现了自身 AVL SDK 反病毒引擎基于的 GPU 加速, 该系统在单 GPU 设备上, 采用全规则集可以达到 40Gbps 的处理速度, 并且可以线性地扩展到多设备集群上, 完全满足核心网和骨干网的处理需求, 并具有极低的硬件成本。

3 总结

高速反病毒引擎目前已经成为一种现实需求。从来自于其他网络应用的经验来看, 硬件加速是最有希望的解决途径。

基于 ASIC 的硬件加速大规模生产的成本较低。其中 FPGA 还具有一定的可编程性。这类基于专用芯片的加速方案可以用于中低端网络设备之中, 成为反病毒功能模块。

基于 GPU 的加速可以认为是目前最好的高端解决方案, 既具有硬件的高速特点, 又具有软件开发的灵活性, 性价比也有很高的竞争力。长期来看, GPU 设备和总线技术在硬件上还有较大的提升空间, 因而带来进一步加速。因此, GPU 加速将是反病毒引擎技术的一个重要发展方向。

在各类硬件加速下, 反病毒引擎完全可以达到目前 Internet 主干网级的吞吐量, 从而实现主干网的深度病毒检测, 这既是一个新兴市场, 其现实意义还将给信息安全产业的发展带来无法预估的深远影响。● (责编 张岩)

参考文献:

- [1] Peter Szor. The Art of Computer Virus Research and Defense[M]. Addison Wesley, 2005.1.
- [2] 李兆星. 反病毒引擎的自主创新与发展[J]. 信息网络安全, 2010, (05): 56-57.
- [3] 谭章熹, 林闯, 任伟源等. 网络处理器的分析与研究[J]. 软件学报, 2003, 14(02): 253-267.
- [4] Chan, P.K., Schlag, M.D.F. Acceleration of an FPGA Router[C]. The 5th Annual IEEE Symposium on FPGAs for Custom Computing Machines, 1997. Issue 16-18, 175-181.
- [5] S. Han et al. PacketShader: A GPU-accelerated Software Router[C]. Proceedings of ACM SIGCOMM 2010, Sep 2010.
- [6] G. Vasiladis et al. Gsnort: High Performance Network Intrusion Detection Using Graphics Processors[C]. In Proceedings of the 11th International Symposium on Recent Advances In Intrusion Detection. Sep. 2008.

[7] Abhishek Mitra et al. Compiling PCRE to FPGA for Accelerating SNORT IDS[C]. ACM/IEEE Symp. on Architecture for Networking and Communication Systems, 2007.

[8] Michael C. Schatz and Cole Trapnell. Fast Exact String Matching on the GPU[EB/OL]. <http://www.cbcb.umd.edu/software/cmatch/Cmatch.pdf>, 2010-12-22.

[9] Jiangfeng Peng, Hu Chen. CUGrep: A GPU-based high performance multi-string matching system. International Conference on Future Computer and Communication[C]. 2010, Vol.1, Issue 21-24:77-81.

[10] Yi-Hua E. Yang, Viktor K. Prasanna. Automatic Construction of Large-Scale Regular Expression Matching Engines on FPGA[C]. International Conference on Reconfigurable Computing and FPGAs, 2008. Issue 3-5:73-78.

[11] Yi-Hua E. Yang et al. Compact Architecture for High-Throughput Regular Expression Matching on FPGA[C]. In Proc. of 2008 ACM/IEEE Symposium on Architectures for Networking and Communications Systems, 2008.

[12] 张庆丹, 戴正华, 冯圣中等. 基于 GPU 的串匹配算法研究[J]. 计算机应用, 2006, 26(07): 1735-1737.

[13] 潘志浩, 杨博文, 曹炳尧. 基于网络处理器的深度包检测系统的研究[J]. 微计算机信息, 2009, (27): 115-116.

[14] 王恩海. 特征匹配引擎设计与实现[J]. 计算机系统应用, 2010, 19(9): 115-119.

[15] 王烁. 字符串模式匹配的硬件加速研究[D]. 中国科学技术大学, 2008.

[16] 周宗伟. 高性能大规模模式匹配算法研究[D]. 清华大学, 2008.

[17] G. Vasiladis, S. Ioannidis. GrAVity: A Massively Parallel Antivirus Engine[C]. In Proceedings of the 13th International Symposium on Recent Advances In Intrusion Detection. Sep. 2010.

[18] S. K. Tewksbury. Application-Specific Integrated Circuits(ASICS) [C]. Chapter in The Electrical Engineering Handbook, CRC Press, R. Dorf (Ed). (1998).

[19] K. Fatahalian, M. Houston. A closer look at GPUs[J]. Communications of the ACM, 2008, Vol.51, No.10, 50-57.

[20] NVIDIA Corporation. CUDA in Action[EB/OL]. <http://www.nvidia.com/cuda>, 2010-10-01/2010-12-20.

[21] 张舒, 褚艳利. GPU 高性能计算之 CUDA[M]. 北京: 中国水利水电出版社, 2009.10: 2-13.

[22] John D. Owens et al. A Survey of General-Purpose Computation on Graphics Hardware[C]. Computer Graphics Forum, 2007,26(1):80-113.

[23] N. Jacob, C. Brodley. Offloading IDS Computation to the GPU[C]. 22nd Annual Computer Security Applications Conference, 2006. 371-380.

[24] L. Costa. GPU Support for Batch Oriented Workloads[C]. IEEE 28th International Performance Computing and Communications Conference, 2009. 231-238.

[25] NVIDIA Corporation. CUDA C Best Practices Guide[EB/OL]. <http://developer.download.nvidia.com>, 2010-08-20/2010-12-15.



全民阅读 报刊刊

根据中华人民共和国公安部发布的《关于转发〈关于开展“全民阅读报刊刊”活动的通知〉的通知》(公宣[2010]107号)文件的精神, 为进一步拓宽公安民警的阅读视野, 本刊编辑部特向全国公安民警推荐《信息安全》杂志。