Sistemas Operativos Avanzados

Objetivos (*Course outcomes*)
En el curso los alumnos:

- Analizarán temas avanzados de sistemas operativos, y
- Revisarán críticamente resultados de investigación del área.

Descripción

El curso introducirá brevemente los conceptos básicos de sistemas operativos y profundizará por medio de artículos de investigación. Algunos de los artículos contendrán los principios fundamentales de los sistemas operativos modernos. Otros artículos cubrirán trabajos recientes que exploran el estado del arte y muestran la evolución de los sistemas operativos. Los alumnos deberán estudiar de manera independiente todos los artículos y llegar muy bien preparados a la clase para discutirlos. Además, los alumnos presentarán en clase artículos científicos.

Temario

1. Introducción a los sistemas operativos

2. Sincronización y Concurrencia
    - Introducción a la comunicación y sincronización de procesos concurrentes
    - Artículos a Analizar:
        - FlexSC: Flexible system call scheduling with exception-less system calls [25].
        - Eraser: A Dynamic Data Race Detector for Multithreaded Programs [16].
        - Scheduler Activations: Effective Kernel Support for the User-Level Management of Parallelism [2].
        - Experience with processes and monitors in Mesa [10]
        - Threads Cannot be Implemented as a Library [4]
        - Eliminating Receive Livelock in an Interrupt-Driven Kernel [12]

3. Planificación
    - Planificación monoprocesador.
    - Planificación multi-procesador y en tiempo real.
    - Artículos a Analizar:
        - The spring kernel: A new paradigm for real-time systems [17]
        - A fair share scheduler [9]
        - Lottery scheduling: flexible proportional-share resource management [20]
        - The Linux scheduler: a decade of wasted cores [23]

4. Administración de la Memoria Virtual
    - Introducción a la administración de memoria virtual
    - Artículos a Analizar:
        - Practical, transparent operating system support for superpages [13]
        - Memory Resource Management in VMware ESX Server [19]
        - Native Client: a sandbox for portable, untrusted x86 native code [21]

5. Sistemas de archivos
    - Introducción a los sistemas de archivos
    - Artículos a analizar:
        - The Google File System [7]
        - Bigtable: A Distributed Storage System for Structured Data [6]
        - Design and Implementation of the Second Extended Filesystem [5]
        - Journaling the Linux ext2fs Filesystem [18]
        - Rethink the Sync [14]
        - Leases: An Efficient Fault-Tolerant Mechanism for Distributed File Cache Consistency [8]
        - Venti: a new approach to archival storage [15]
        - ShieldFS: a self-healing, ransomware-aware filesystem [22]

6. Virtualización
    - Tipos de virtualización
    - Virtualización de CPU, almacenamiento y red.
    - Artículos a Analizar:
        - Xen and the Art of Virtualization [3]
        - A comparison of software and hardware techniques for x86 virtualization [1]
        - Container-based operating system virtualization: a scalable, high-performance alternative to hypervisors [28].
        - Containers and virtual machines at scale: A comparative study [26]

7. Diseño de sistemas operativos.
    - Artículos a Analizar
        - Hints for Computer System Design [11]
        - LegoOS: A Disseminated, Distributed OS for Hardware Resource Disaggregation [27].

Referencias

Artículos:

[1] Adams, K., and Agesen, O. "*A comparison of software and hardware techniques for x86 virtualization*" ACM Sigplan Notices, 41, 11, 2006, 2-13.

[2] Anderson, T. E., Bershad, B. N., Lazowska, E. D., Levy, H. M. "*Scheduler activations: effective kernel support for the user-level management of parallelism*", In Proceedings of the thirteenth ACM symposium on Operating systems principles. ACM, New York, NY, USA, 1991, 95-109.

[3] Barham, P., Dragovic, B., Fraser, K., Hand, S., Harris, T., Ho, A., Neugebauer, R., Pratt, I., and Warfield, A., "*Xen and the art of virtualization*" In Proceedings of the nineteenth ACM symposium on Operating systems principles, ACM, New York, NY, USA, 2003, pp. 164-177.

[4] Boehm, H.J. "*Threads cannot be implemented as a library"*. SIGPLAN Not. 40, 6, 2005, 261-268.

[5] Card, R., Ts'o T., and Tweedie, S. "*Design and Implementation of the Second Extended Filesystem*" Proc. of First Dutch International Symposium on Linux, ISBN 90-367-0385-9, 1994.

[6] Chang, F., Dean, J., Ghemawat, S., Hsieh, W. C., Wallach, D. A., Burrows, M., Chandra, T., Fikes, A., and Gruber, R. E., "*Bigtable: a distributed storage system for structured data*" In Proceedings of the 7th

USENIX Symposium on Operating Systems Design and Implementation - Volume 7 (OSDI '06), Vol. 7. USENIX Association, Berkeley, CA, USA, 2006, pp. 15-15.

[7] Ghemawat, S., Gobioff, H., and Leung, S.-T, "*The google file system*". In proceedings of the nineteenth ACM symposium on Operating systems principles (New York, NY, USA, 2003), ACM, pp. 29–43.

[8] Gray C., and Cheriton, D. "*Leases: an efficient fault-tolerant mechanism for distributed file cache consistency*" SIGOPS Oper. Syst. Rev. 23, 5, 1989, pp. 202-210.

[9] Kay J., and Lauder, P., "*A fair share scheduler*" Commun. ACM 31 (1), 1988, 44-55.

[10] Lampson B. W., and Redell, D.D. "*Experience with processes and monitors in Mesa*" Commun. ACM 23, 2, 1980, 105-117.

[11] Lampson, B.W. "*Hints for Computer System Design*" In Proceedings of the Ninth ACM Symposium on Operating Systems Principles, USA, 1983, pp. 33-48.

[12] Mogul, J. C., and Ramakrishnan, K. K. "*Eliminating receive livelock in an interrupt-driven kernel*". ACM Trans. Comput. Syst. 15, 3, 1997, 217-252.

[13] Navarro, J., Iyer, S., Druschel, P., and Cox, A. "*Practical, transparent operating system support for superpages"*. In Proceedings of the 5th symposium on Operating systems design and implementation, ACM, New York, NY, USA, 2002, pp. 89-104.

[14] Nightingale, E. B., Veeraraghavan, K., Chen, P. M., and Flinn, J. "*Rethink the sync*" In Proceedings of the 7th symposium on Operating systems design and implementation (OSDI '06). USENIX Association, Berkeley, CA, USA, 2006, 1-14.

[15] Quinlan S., and Dorward S. "*Venti: A New Approach to Archival Storage*". In Proceedings of the Conference on File and Storage Technologies (FAST '02), Darrell D. E. Long (Ed.). USENIX Association, Berkeley, CA, USA, 2002, pp. 89-101.

[16] Savage, S., Burrows, M., Nelson, G., Sobalvarro, P., and Anderson, T, "*Eraser: a dynamic data race detector for multithreaded programs*" ACM Trans. Comput. Syst. 15(4), 1997, 391-411.

[17] Stankovic, J.A., and Krithi R. "*The spring kernel: A new paradigm for real-time systems.*" IEEE Software, 8, 3, 1991, 62-72.

[18] Tweedie, S. C. "*Journaling the Linux ext2fs Filesystem"* in proceedings of the 4th Annual LinuxExpo, Durham, NC., 1998.

[19] Waldspurger, C.A., "*Memory resource management in VMware ESX server*". SIGOPS Oper. Syst. Rev. 36, SI, 2002, pp. 181-194.

[20] C. A., and Weihl, W. E., "*Lottery scheduling: flexible proportional-share resource management*" In Proceedings of the 1st USENIX conference on Operating Systems Design and Implementation, USENIX Association, Berkeley, CA, USA, 1994, Article 1.

[21] Yee, B., Sehr, D., Dardyk, G., Chen J. B., Muth, R., Ormandy, T., Okasaka, S., Narula, N., and Fullagar, N., "*Native Client: a sandbox for portable, untrusted x86 native code*" Commun. ACM 53, 1, 91-99, 2010.

[22] Continella, A., Guagnelli, A., Zingaro, G., De Pasquale, G., Barenghi, A., Zanero, S., & Maggi, F. (2016, December). *ShieldFS: a self-healing, ransomware-aware filesystem*. In Proceedings of the 32nd Annual Conference on Computer Security Applications (pp. 336-347). ACM

[23] Lozi, J. P., Lepers, B., Funston, J., Gaud, F., Quéma, V., & Fedorova, A. (2016, April). *The Linux scheduler: a decade of wasted cores*. In Proceedings of the Eleventh European Conference on Computer Systems (p. 1). ACM

[24] Lozi, J. P., Lepers, B., Funston, J., Gaud, F., Quéma, V., & Fedorova, A. (2016, April). *The Linux scheduler: a decade of wasted cores*. In Proceedings of the Eleventh European Conference on Computer Systems (p. 1). ACM.

[25] Soares, L., & Stumm, M. (2010). *FlexSC : Flexible System Call Scheduling with Exception-Less System Calls*. In 9th USENIX Symposium on Operating Systems Design and Implementation (OSDI 10).

[26] Sharma, P., Chaufournier, L., Shenoy, P., & Tay, Y. C. (2016, November). Containers and virtual machines at scale: A comparative study. In Proceedings of the 17th international middleware conference (pp. 1-13).

[27] Shan, Y., Huang, Y., Chen, Y., & Zhang, Y. (2018). LegoOS: A Disseminated, Distributed OS for Hardware Resource Disaggregation. In 13th USENIX Symposium on Operating Systems Design and Implementation (OSDI 18) (pp. 69-87).

[28] Soltesz, S., Pötzl, H., Fiuczynski, M. E., Bavier, A., & Peterson, L. (2007, March). Container-based operating system virtualization: a scalable, high-performance alternative to hypervisors. In Proceedings of the 2Nd ACM SIGOPS/EuroSys european conference on computer systems 2007 (pp. 275-287).

Libros de consulta:

- William Stallings, *Operating Systems: Internals and Design Principles*, Prentice Hall, 6th edition, 2009.
- Andrew S. Tanenbaum, *Modern Operating Systems*, Prentice Hall; 3rd. edition, 2007.
- Silberschatz, Galvin, and Gagne, *Operating Systems Concepts*, Wiley; 8th. edition (July 29, 2008)

Vínculos web:

- Stanford's Advanced Topics in Operating Systems Course, http://www.stanford.edu/class/cs240/
- CMU's Advanced and Distributed Operating Systems Course, http://www.cs.cmu.edu/~15712/,
- Cornell's Advanced Course in Computer Systems, http://www.cs.cornell.edu/courses/cs614/2003sp/,
- Texas A&M's Operating Systems Course http://faculty.cs.tamu.edu/bettati/Courses/613/2011A/course_homepage.html,
- Xen project, http://www.xen.org
- KVM project, http://www.linux-kvm.org/page/Main_Page
- D.C. Marinescu, "Cloud Computing and Computer Clouds", Lecture Notes, Computer Science Division, Department of Electrical Engineering & Computer Science, University of Central Florida, 2012, https://www.cs.ucf.edu/~dcm/Teaching/CDA5532-CloudComputing/LectureNotes.pdf
- Ontko, R., Reeder, A., Tanenbaum, A. (2001). Modern Operating Systems Simulators. http://www.ontko.com/moss/