Análise do Sistema Operacional Windows 10

Jorge Luiz Andrade, Ana Carolina, and Matheus Castro

***Abstract*—aa aaa[1]**

I. Introdução

II. Windows 10

A. Arquitetura

B. Gerência de memória

O Windows 10 estabelece 4GB como limite para memória física em versões 32-bits e 2TB em versões 64-bits, com exceção da versão Home, que possui limite de 128GB em sua versão de 64-bits [2].

A memória física pode ser dividida em [2]:

- **Reservada para o Hardware**: armazena drivers de hardware que devem sempre permanecer na memória física, não estando disponível para uso do gerenciador de memória.
- **Em uso:** É a memória em uso por todos os processos em execução, *kernel* do SO e *drivers*.
- **Modificada:** É a memória de páginas que foram modificadas em processos que ficaram em espera. Os dados anteriores são escritos em disco, mas facilmente recuperados.
- **Em espera:** É a memória que estava alocada em processos que terminaram normalmente. O gerenciador de memória mantém os dados em memória como uma espécie de cache para arquivos usados recentemente. A memória em espera está disponível para alocação, mas suas páginas são classificadas de 0 a 7, sendo as páginas com menores valores usadas primeiro.
- **Livre:** É a memória que ainda não foi alocada ou que retornou para o gerenciador de memória por um processo que foi terminado.

O gerenciador de memória do Windows 10 faz parte do Windows executive, uma porção em baixo nível do seu *kernel*, residindo no arquivo *Ntoskrnl.exe*. É responsável, entre outras funções, por [3]:

- Alocar, desalocar e gerenciar a memória virtual, que em sua maior parte está exposta por meio da API do Windows ou de interfaces para drivers de dispositivos em modo kernel;
- Garantir que processos não acessem regiões a que não possuem permissão;

O ambiente Windows, de modo geral, utiliza o conceito de espaço de endereçamento virtual para um processo, sendo este o conjunto de endereços da memória virtual que esse processo tem acesso. O espaço de endereçamento é privado e não pode ser acessado por outros processos que não o compartilhem [1].

O espaço de endereçamento em versões 32-bits do Windows é de até 4GB, dividido em uma partição para o processo e outra para uso do sistema. Versões 64-bits do sistema suportam endereçamento em modo usuário de até 8TB [2].

Assim como todos os componentes do *Windows executive*, o gerenciador de memória é totalmente reentrante, ou seja, pode executado novamente antes que a execução anterior tenha sido concluída, e suporta execução simultânea em sistemas multiprocessados. Isso permite que duas ou mais *threads* adquiram recursos de forma que seus dados não sejam corrompidos [3].

C. Gerência de processos

D. Gerência de arquivos

Assim como ocorre desde a versão 3.1, o *Windows* 10 utiliza o *NTFS*(*New Technology File System*) como seu sistema de arquivos padrão em ambientes domésticos, suportando volumes e arquivos de até 256TB quando utilizado o tamanho do *cluster* padrão de 64KB e até 2^{32}-1 arquivos por volume e pasta [4].

O NTFS foi desenvolvido de forma a incluir funcionalidades necessárias em sistemas de arquivos empresarias. Isso inclui integridade e recuperação de dados, proteção à informações sensíveis, redundância de dados e tolerância a falhas [3].

- **Integridade e recuperação de dados**: Modificações no sistema de arquivos são realizadas em operações atômicas, ou seja, toda a operação deve ser completada ou nenhuma parte dela o será.
- **Segurança**: Arquivos e diretórios são associados à um arquivo oculto de segurança contendo as informações de permissão. Assim que um processo tenta utilizar um arquivo, suas permissões são checadas, e seu acesso só é permitido se autorizado pelo administrador do sistema ou pelo dono do arquivo.
- **Redundância e tolerância a falhas**: O NTFS garante que o sistema de arquivos permaneça acessível após uma falha do disco, mas não garante integridade dos arquivos em si. Essa integridade, entretanto, é alcançada utilizando-se RAID 1 e 5.

O sistema NTFS não tenta evitar fragmentação de arquivos durante suas alocações. Entretanto, além de sua própria ferramenta, o Windows inclui uma API que permite o desenvolvimento de ferramentas de desfragmentação de terceiros, que permite que dados de arquivos sejam movidos de forma que ocupem *clusters* contíguos, possuindo como única limitação o impedimento da desfragmentação em arquivos de paginação e de logs do sistema NTFS [3].

E. Gerência de E/S

O gerenciador de Entrada/Saída do *kernel* do Windows 10 realiza a comunicação entre o sistema operacional e os *drivers* de dispositivos por meio de *IRPs*(*I/O request packets*,

ou pacotes de requisição de E/S). Isso permite que *threads* individuais operem em múltiplas chamadas de E/S de forma concorrente [3].

Devido aos dispositivos operarem em velocidades diferentes daquela do sistema operacional, a comunicação IRP se assemelha a pacotes de redes, sendo passados do sistema operacional para um *driver* de dispositivo, e de um *driver* para outro, por meio do gerenciador de E/S. O sistema de E/S do Windows possui um modelo em camadas, ou pilha, onde cada controlador na pilha envia e recebe IRPs [5].

Além da criação e distribuição de IRPs, o gerenciador também possui código comum a diferentes controladores, facilitando a criação e utilização de *drivers* individuais de dispositivos.

F. Interrupções

G. Kernel

H. Suporte a threads

I. Segurança

III. Conclusões

References

[1] Mark Russinovich, David Solomon, and Alex Ionescu. *Windows Internals Part 2*. Microsoft Press, 6 edition, 2012.

[2] Sushovon Sinha. Physical and virtual memory in windows 10. http://answers.microsoft.com/en-us/windows/forum/windows_10-performance/physical-and-virtual-memory-in-windows-10/e36fb5bc-9ac8-49af-951c-e7d39b979938. Acessado em 22 de novembro de 2016.

[3] Mark Russinovich, David Solomon, and Alex Ionescu. *Windows Internals Part 2*. Microsoft Press, 6 edition, 2012.

[4] Default cluster size for ntfs, fat, and exfat. https://support.microsoft.com/en-us/kb/140365. Acessado em 26 de novembro de 2016.

[5] Windows kernel-mode i/o manager. https://msdn.microsoft.com/en-us/library/windows/hardware/ff565734. Acessado em 26 de novembro de 2016.