

## Unit OSB: Comparing the Linux and Windows Kernels

### B. Quiz

Windows Operating System Internals - by David A. Solomon and Mark E. Russinovich with Andreas Polze

## Copyright Notice

© 2000-2005 David A. Solomon and Mark Russinovich

- These materials are part of the *Windows Operating System Internals Curriculum Development Kit*, developed by David A. Solomon and Mark E. Russinovich with Andreas Polze
- Microsoft has licensed these materials from David Solomon Expert Seminars, Inc. for distribution to academic organizations solely for use in academic environments (and not for commercial use)

## Kernel Architecture

Windows has what type of kernel architecture?

- a) Microkernel
- b) Modified-Monolithic
- c) Monolithic
- d) Modular microkernel

3

## Linux Tasks

The Windows equivalent of a Linux task is:

- a) Thread
- b) Process with no threads
- c) Process with one thread
- d) Task

4

## Scheduling Priorities

The Linux equivalent of the Windows “real time” priority range is:

- a) Normal
- b) Fixed FIFO
- c) Fixed Round Robin
- d) Both (b) and (c)

5

## Linux/Windows Differences

Which of the following is true for Linux, but not for Windows:

- a) Uses form of the “clock” working set management algorithm
- b) Supports more than 64 processors
- c) Supports 4/4 virtual address mode
- d) Both c and d
- e) None of the above

6

## Scheduler

Which of the following are differences between the Windows scheduler and Linux scheduler?

- a) Biases scheduling decisions against processes with many threads
- b) Performs load balancing in SMP scheduling
- c) Maximum timeslice is less than 200ms
- d) Has fixed and dynamic priority ranges

7

## Linux Features

Which of the following do not apply to Linux?

- a) Fully asynchronous I/O
- b) Re-entrant kernel
- c) Pre-emptible kernel
- d) Has an  $O(n)$  scheduler

8

## SendFile/TransmitFile

The SendFile API in Linux and TransmitFile API in Windows provide for:

- a) Wake-one semantics
- b) One-copy network I/O
- c) Zero-copy network I/O
- d) Thundering herd semantics

9

## CriticalSection/Futex

The difference between a Windows Critical Section and Windows Futex is:

- a) Futexes always transition to kernel mode
- b) Futexes have prioritized waits
- c) CriticalSections have prioritized waits
- d) Critical sections work across processes

10