RELIABLE OPERATING SYSTEMS

Research Summary

1st EuroSys Doctoral Workshop
October 23, 2005 – Brighton, UK

Jorrit N. Herder
Dept. of Computer Science
Vrije Universiteit Amsterdam
PERCEIVED PROBLEMS

• **Weak security and reliability**
  - Computer crashes
  - Digital pests (viruses, worms, etc.)

• **Complexity**
  - Hard to maintain and configure
  - Too large for embedded and mobile computing
TYPICAL OS STRUCTURES

(a) Monolithic kernel

(b) Multiserver Hybrid kernel

(c) Single-server Minimal kernel
INHERENT PROPERTIES

- **Fundamental design flaws in monolithic kernels**
  - All code runs at highest privilege level (breaches POLA)
  - No proper fault isolation (any bug can be fatal)
  - Huge amount of code in kernel (1-20 bugs per 1000 LOC)
  - Untrusted, 3rd party code in kernel (70% driver bugs)
  - Hard to maintain and configure (limited portability)

- **Lack of modularity causes problems**
  - Proper OS design can solve above problems
DESIGN OF A RELIABLE OS: MINIX 3

- Recent work
  - Design and implementation of the MINIX 3 operating system
  - Transformation into a minimal kernel design (< 3800 LOC)
  - All servers and drivers run in a separate user-mode process

- Current research
  - Additional reliability properties
MINIX 3: ACHIEVING RELIABILITY

● **Design principles**
  - Simplicity
  - Modularity
  - Least authorization
  - Fault tolerance

● **How this helps**
  - Number of fatal bugs is reduced
  - Damage that bugs can do is limited
  - Recovery from common failures is possible
MINIX 3: STRUCTURAL MEASURES

- Stable minimal kernel (< 3800 LOC) reduces # fatal bugs
- Isolated, user-mode processes in private address space
- Reliable IPC: small, fixed-size message passing
- Deadlock avoidance and deadlock detection
- Buffer overruns prevented and damage limited
- Bad pointers in OS are caught with MMU hardware
- Scheduler detects and tames infinite loops in OS
- Monitor and restart malfunctioning OS services
MINIX 3: PER-PROCESS POLICIES

- IPC only possible if type and target are allowed
- Only exported list of kernel calls can be called
- Access to individual I/O ports can be restricted
- Access to remote memory, e.g., video RAM
- Scheduling priority and quantum size
- Period for reincarnation server status checks
MINIX 3: REINCARNATION SERVER

- **Start servers and drivers**
  1. Encapsulate in new process
  2. Assign only needed privileges
  3. Start in controlled environment

- **Monitor services**
  a. Immediate crash detection
  b. Periodically check status

- **Fix problems**
  - Kill and restart fresh copy
SUMMARY & CONCLUSION

- Different OS structures and properties
  - Fundamental problems with monolithic systems
  - Inherent benefits of modular systems

- OS reliability *is* possible: **MINIX 3**
  - Multiserver OS with minimal kernel (< 3800 LOC)
  - Improvements over other operating systems
    - We reduce the number of fatal bugs
    - We limit the damage bugs can do
    - We can recover from common failures
QUESTIONS?

• The MINIX 3 team
  - Jorrit Herder
  - Ben Gras
  - Philip Homburg
  - Herbert Bos
  - Andy Tanenbaum

• More information
  - Web: www.minix3.org
    - As of tomorrow!
  - News: comp.os.minix
  - Mail: jnherder@cs.vu.nl
PERFORMANCE ISSUES

• **Historical fear: modularity incurs overhead**
  - Communication overhead
  - Copying of data

• **Times have changed ...**
  - New insights reduced performance penalty (only 5-10%)
  - Absolute performance penalty is minimal these days
  - *Users gladly sacrifice some performance for reliability*
MINIX 3: SOME NUMBERS

• Performance measurements
  – Time from multiboot monitor to login is under 5 sec.
  – The system can do a full build of itself within 4 sec.
  – Run times for typical applications: 6% overhead
  – File system and disk I/O performance: 9% overhead
  – Networking performance: Ethernet at full speed

• Code size statistics
  – Kernel is 3800 LOC; rest of the OS is in user space
  – Minimal POSIX-conformant system is 18,000 LOC