VE482 - Homework 8

Introduction to Operating Systems

Kexuan Huang 518370910126

December 12, 2021

Ex.1 - Input/Output

1. Is it possible to scan a document and send it through a wireless network at full speed? Explain why, why not.

Is is possible to scan a document and send it through a wireless network at full speed is possible, only if the calculation and I/O are completely separated. While in real world scenario, scanning a document involves calculating, as a result, the data cannot be sent out though a wireless network at full speed.

2. How can an OS facilitate the installation of a new device without any need to recompile the OS or even to reboot it?

The new device can be installed without the need of recompilation of the OS with a loadable driver, which is a kind of modules that is able to be loaded and unloaded dynamically.

3. In the slides four I/O software layers are listed; in which layers are the following operations done:

a) Computing the track, sector, and head for a disk read

The Device Drivers layer.

b) Writing command to the device registers

The Device Drivers layer.

c) Checking is the user is allowed to use the device

The Device Independent OS Software layer.

d) Converting binary integers to ASCII for printing

The User Level IO Software layer.

4. A DMA controller has four channels. The controller is able to request a 32bit word every 100 nsec. A respond takes equally long. How fats should the bus be in order to avoid being a bottleneck?

$$\frac{32\text{ bit} \times 2}{100\text{ nsec} \times 4} = 1.6 \times 10^8\text{ bit/s} = 2 \times 10^7 B/s \approx 19.07\text{ MB/s}$$

5. Briefly explain what a thin client is.[1]

A thin client is a computer that runs from resources stored on a central server instead of a localized hard drive. Thin clients work by connecting remotely to a server-based computing environment where most applications, sensitive data, and memory, are stored.

6. Describe how a hard disk is working and give a few details regarding its geometry.[2]

A hard disk is an electro-mechanical data storage device that stores and retrieves digital data using magnetic storage and one or more rigid rapidly rotating platters coated with magnetic material. The platters are paired with magnetic heads, usually arranged on a moving actuator arm, which read and write data to the platter surfaces.

[1] Forcepoint
[2] Wikipedia

7. What are differences between RAID 0, 1, 5, 6, and 10?[3]

- **RAID 0**: In a RAID 0 system data are split up into blocks that get written across all the drives in the array. By using multiple disks (at least 2) at the same time, this offers superior I/O performance. This performance can be enhanced further by using multiple controllers, ideally one controller per disk.
- **RAID 1**: Data are stored twice by writing them to both the data drive (or set of data drives) and a mirror drive (or set of drives). If a drive fails, the controller uses either the data drive or the mirror drive for data recovery and continuous operation. You need at least 2 drives for a RAID 1 array.
- **RAID 5** is the most common secure RAID level. It requires at least 3 drives but can work with up to 16. Data blocks are striped across the drives and on one drive a parity checksum of all the block data is written. The parity data are not written to a fixed drive, they are spread across all drives, as the drawing below shows. Using the parity data, the computer can recalculate the data of one of the other data blocks, should those data no longer be available. That means a RAID 5 array can withstand a single drive failure without losing data or access to data. Although RAID 5 can be achieved in software, a hardware controller is recommended. Often extra cache memory is used on these controllers to improve the write performance.
- **RAID 6** is like RAID 5, but the parity data are written to two drives. That means it requires at least 4 drives and can withstand 2 drives dying simultaneously. The chances that two drives break down at exactly the same moment are of course very small. However, if a drive in a RAID 5 systems dies and is replaced by a new drive, it takes hours or even more than a day to rebuild the swapped drive. If another drive dies during that time, you still lose all of your data. With RAID 6, the RAID array will even survive that second failure.
- **RAID 10**: It is possible to combine the advantages (and disadvantages) of RAID 0 and RAID 1 in one single system. This is a nested or hybrid RAID configuration. It provides security by mirroring all data on secondary drives while using striping across each set of drives to speed up data transfers.

[3] https://www.prepressure.com/library/technology/raid

Ex. 2 - Multiprocessors

1. Suppose that the TSL instruction was not available for synchronizing a multiprocessor. Instead, another instruction, SWP was provided that atomically swapped the contents of a register with a word in memory. How could that be used to provide multiprocessor synchronisation?

It could be used by preloading a 1 into the register to be used. Then that register and the memory word are atomically swapped. After the instruction, the memory word is locked (i.e., has a value of 1). Its previous value is now contained in the register. If it was previously locked, the word has not been changed and the caller must loop. If it was previously unlocked, it is now locked.

2. Briefly describe virtualization and the main challenges that need to be overcome.[4] [5]

In computing, virtualization or virtualisation (sometimes abbreviated v12n, a numeronym) is the act of creating a virtual (rather than actual) version of something, including virtual computer hardware platforms, storage devices, and computer network resources. Virtualization is about using software to emulate functionality of hardware.

The main challenges are:

1. Resource distribution
2. VM Sprawl
3. Backward compatibility
4. Performance monitoring
5. Backup
6. Security

[4] Wikipedia
[5] TechAdvisory.org

Ex. 3 - File Systems

1. Assuming the current working directory is `/usr/share/doc/mutt/examples` what is the absolute path of `../../../doc/../../lib/mutt`?

`/usr/lib/mutt`

2. A Unix system has 1KB blocks and 4B disk addresses. What is the maximum file size if i-nodes contain 13 direct entries, and one single, double, and triple indirect entry each?

For each indirect block, there are

$$\frac{1\,\mathrm{KB}}{4\,\mathrm{B}} = 256\,\mathrm{blocks}.$$

- 13 direct entries will take 13 KB.
- one single indirect entry will take 256 KB
- double single indirect entry will take $256 \times 256 = 65536$ KB
- triple single indirect entry will take $256 \times 256 \times 256 = 16777216$ KB

As a result, the maximum file size is 16843021 KB, which is approximately 16 GB.

3. The time required to read a certain number of bytes can be approximated by summing the seek, rotation delay, and transfer times. Given a disk with mean seek time of 8 msec, a rotation rate of 15,000 rpm and 262,144 bytes per track. What are the data rates for block sizes of (i) 1 KB, (ii) 2 KB, and (iii) 4 KB?

The rotation delay $\approx \frac{1000\times60}{15000\times2} = 2\,\mathrm{msec}$.
The total time is:

(i) $8 + 2 + \frac{1024}{262144} = 10.004\,\mathrm{ms}$

(ii) $8 + 2 + \frac{2\times1024}{262144} = 10.008\,\mathrm{ms}$

(iii) $8 + 2 + \frac{4\times1024}{262144} = 10.016\,\mathrm{ms}$

The data rates can't be found unless the amount of data transfered are given.

Ex. 4 - Security

1. If using only lowercase letters how many possible 8 letter words can be chosen? Explain why a password should always contain lower and uppercase letters, numbers and symbols.

We can choose $26^8 = 208,827,064,576$ possible combinations with only lowercase letters. We can choose $52^8 = 53,459,728,531,456$ possible combinations with lowercase and uppercase letters, which is 256 times larger than only-lowercase combinations. If we take numbers and symbols into considerations, there're even more combinations. As a result, it takes much more times for attacker to crack the password.

2. Explain why memory dynamically allocated using `malloc` is sometimes random and some other times set to 0.[6]

When we call `malloc()`, one of two things will happen:

- It recycles memory that was previous allocated and freed from the same process.
- It requests new page(s) from the operating system.

1. In the first case, the memory will contain data leftover from previous allocations. So it won't be zero. This is the usual case when performing small allocations.
2. In the second case, the memory will be from the OS. This happens when the program runs out of memory, e.g. when we are requesting a very large allocation. When the OS gives us memory, it could have been freed from a different process. So that memory could contain sensitive information such as a password. So to prevent attackers reading such data, the OS will zero it before it gives it to us.

3. A computer system is made of several components each featuring a different level of security. How secure is this system?

According to the Wooden Barrel Theory, the security level of this system is determined by the weakest security level among the components. So, the system is probably not that secure.

[6] stackoverflow

4. Can security be achieved? Explain your answer.

No, we can't guarantee 100% security and every system has flaws which could probably be exploited by attackers. We can only try our best to improve the security level of a system to some extents and tremendous resources and efforts should be putted in, not to mention to build a error-free system. As an old problems being fixed, new problems might occur somewhere sometime.

Ex. 5 - Research

Write about a page on the topic of firewalls on Linux systems; include information on `iptables`, `ebtables`, and `arptables`. Do not forget to reference your sources of information.[7] [8]

A Linux firewall is defined as a solution or service that regulates, protects, and blocks network traffic as it passes to and from a Linux-based environment. Nearly 75% of the world's servers run on Linux, so these solutions are essential to provide secure access to users and end customers.

- **iptables** is used to set up, maintain, and inspect the tables of IP packet filter rules in the Linux kernel. Several different tables may be defined. Each table contains a number of built-in chains and may also contain user-defined chains. Each chain is a list of rules which can match a set of packets. Each rule specifies what to do with a packet that matches. This is called a 'target', which may be a jump to a user-defined chain in the same table.
- **ebtables** is an application program used to set up and maintain the tables of rules (inside the Linux kernel) that inspect Ethernet frames. It is analogous to the iptables application, but less complicated, due to the fact that the Ethernet protocol is much simpler than the IP protocol.
- **arptables** is used to set up, maintain, and inspect the tables of ARP packet filter rules in the Linux kernel. Several different tables may be defined. Each table contains a number of built-in chains and may also contain user-defined chains. Each chain is a list of rules which can match a set of packets. Each rule specifies what to do with a packet that matches. This is called a 'target', which may be a jump to a user-defined chain in the same table.

[7] Toolbox
[8] Linux man page

Ex. 6 - Linux

What is a kernel panic? Generate one on your Linux system and troubleshoot it. Provide all the details.[9]

A kernel panic is a safety measure taken by an operating system's kernel upon detecting an internal fatal error in which either it is unable to safely recover or continuing to run the system would have a higher risk of major data loss.

1. Enable `Sysrq` to trigger kernel panic: `echo 1 > /proc/sys/kernel/sysrq`
2. Trigger `Sysrq` panic event (c) to crash the system: `echo c > /proc/sysrq-trigger`

We can recover the system by rebooting it.

Ex. 7 - Course survey

Done :-)

[9] Wikipedia