|  |
| --- |
| استاد درستی << بسم الله الرحمن الرحیم >> گزارش کار اول  گزارش کار آزمایشگاه سیستم عامل  فاطمه کریمی 4011231054  سارا سعادت 4011231033    **نصب ابنتو در :vmware**  راه‌اندازی اوبونتو (Ubuntu) در یک ماشین مجازی ایجادشده با VMware به چند مرحله ساده نیاز دارد.  پیش‌نیازها:   1. نصب VMware Workstation / VMware Player 2. دریافت فایل ISO اوبونتو   مراحل راه‌اندازی اوبونتو در VMware:   1. **ایجاد ماشین مجازی جدید:**  * VMware را باز میکنیم * از منوی بالا روی **"Create a New Virtual Machine"** کلیک می کنیم * گزینه **Typical (recommended)** را انتخاب کرده و روی **Next** می زنیم   **2- انتخاب فایل ISO:**   * در بخش Installation Media:   + گزینه **Installer disc image file (iso):** را انتخاب میکنیم   + فایل ISO اوبونتو را که دانلود کشده را انتخاب میکنیم * روی **Next** کلیک میکنیم   **3-تنظیم نوع سیستم‌عامل:**   * VMware به طور خودکار تشخیص می‌دهد که اوبونتو است، ولی اگر نشناخت:   + Guest Operating System: **Linux**   + Version: **Ubuntu (64-bit)**   **4-ام و مسیر ذخیره ماشین مجازی:**   * یک نام برای ماشین وارد میکنیم، مثلاً: Ubuntu-VM * مسیر ذخیره را مشخص میکنیم (پیش‌فرض مشکلی ندارد)   **5-تنظیم هارد دیسک مجازی:**   * اندازه دیسک را وارد میکنیم(مثلاً: **20 GB یا بیشتر**) * گزینه **Store virtual disk as a single file** را انتخاب میکنیم(بهتر برای عملکرد)   **6-پایان و ساخت ماشین:**   * روی **Finish** کلیک میکنیم. * حالا ماشین ساخته شده و آماده اجراست.   **نصب اوبونتو:**   1. ماشین مجازی را **Start** میکنیم 2. نصب‌کننده اوبونتو به طور خودکار لود می‌شود. 3. زبان دلخواه را انتخاب کن و روی**Install Ubuntu** کلیک میکنیم.   2**-نتیجه دستورات :cleart,history,pwd**  Clear  پاک کردن تمام خروجی‌های قبلی در ترمینال و نمایش یک ترمینال تمیز. در واقع صفحه ترمینال رو رفرش می‌کنه، ولی خروجی‌ها هنوز در حافظه ترمینال هستن  history  نمایش لیستی از دستورات قبلی که در ترمینال وارد شده‌اند، به‌همراه شماره هر دستور.  Pwd  نمایش مسیر (path) فعلی که در آن قرار داریم :  /home/your-username/Desktop    3**) با mkdir یک دایرکتوری ایجاد کنید و با cd به دایرکتوری ساخته شده منتقل بشید :**  mkdir testdir  این دستور یک پوشه به نام testdirدر مسیر فعلی ایجاد می‌کنه.  cd testdir  با این دستور وارد دایرکتوری testdirمی‌شویم.    4**)با دستور cat > [name.txt] دو فایل text ایجاد کرده و داخل هر کدام جمله ای بنویسید با ls فایل ها را مشاهده کنید ؟**  مراحل انجام :  cat>test1.txt  بعد از زدن این دستور، ترمینال منتظر ورود متن می‌شود :  متن خود را وارد میکنیم :  This is the first file  کلید Ctrl + D رو میزنیم (یعنی پایان ورودی)  سپس میتوان با وارد کردن دستور ls فایل ها را مشاهده کرد :    A screenshot of a computer  AI-generated content may be incorrect.  A screenshot of a computer  AI-generated content may be incorrect.**دستور cat>>[name\_of\_file\_1.txt][name\_of\_file\_2.txt] را اجرا کنید ؟ فرق ان با دستور cat>[name\_of\_file\_1.txt][name\_of\_file\_2.txt] چیست ؟**  دستور اول محتوای فایل اول را به فایل دوم اضافه کرد :    مثلا ما در تصویر بالا میخواهیم محتوای فایل test1 را به محتوای فایل test2 منتقل کنیم :  A screenshot of a computer  AI-generated content may be incorrect.  دستور دوم محتوای فایل اول را پاک کرده و محتوای فایل دوم رو جای گزاری میکند برای مثال ما الان محتوای فایل اول را جایگزین محتوای فایل دوم کرده ایم :      **دستور ps aux| grep “a” را اجرا کرده و نتیجه را شرح دهید :**   1. ps aux لیست کامل فرآیندهای در حال اجرا در سیستم رو نشان میدهد (شامل همه‌ی کاربران). 2. | خروجی دستور اول رو به ورودی دستور دوم میدهد 3. grep "a" → فقط خط‌هایی رو فیلتر می کند که شامل حرف **a** باشند.   **نتیجه:** نشان دادن همه‌ی پردازش‌هایی که در خروجی ps aux، شامل حرف a باشند (مثلاً اسم پردازش یا مسیر).    **7) دستور man sudo> [name.txt] را اجرا کنید فایلtext شامل چیست ؟**  **تجزیه و توضیح:**  **man sudo**  این قسمت، **صفحه‌ی راهنمای (manual)** مربوط به دستور sudo رو باز می‌کند. در واقع اطلاعات کاملی درباره‌ی نحوه استفاده از sudo میدهد ؛ مثل:   * توضیحات کلی * نحوه‌ی استفاده (Syntax) * گزینه‌ها و سوئیچ‌ها * مثال‌ها   **ری‌دایرکت خروجی**  علامت > به این معناست که خروجی دستور قبل، **به جای نمایش در ترمینال، داخل یک فایل ذخیره میشود .**  **name.txt**  این اسم فایلیه که می‌خواهیم خروجی داخل آن ذخیره بشه. اگر وجود نداشته باشه، ساخته می‌شود؛ اگر وجود داشته باشد، محتواش پاک و جایگزین می‌شود  **دستور بالا دقیقاً چه کاری انجام می‌دهد؟**  خروجی دستور man sudo (یعنی کل توضیحات و راهنما) رو داخل فایلی به نام name.txt می‌ریزد.    8**- تعداد خطوط فایل text فایل را با دستور wc بدست اورید و با rm ان را حذف کنید ؟**    9**-با دستور version—python3 از نصب پایتون مطمئن بشید و با nano یک فایل پایتونی دلخواه بسازید و ان را در محیط ترمینال run کنید ؟**  **1. مطمئن شدن از نصب پایتون:**  داخل ترمینال دستور رو به رو را وارد میکنیم python3 --version  python3 -V : برای دیدن نسخه دقیق تر استفاده میشود .  خروجی چیزی شبیه این خواهد بود:  Python 3.10.12  اگر پایتون نصب نبود میتوان با دستورات زیر ان را در ابتدا نصب کرد .  sudo apt update  sudo apt install python3  **۲. ساخت یک فایل پایتونی با nano:**  داخل ترمینال دستور روبه را وارد میکنیم nano test.py  حال در ویرایشگر nano، این کد ساده رو مینویسیم:  This is the test  برای ذخیره فایل و خروج از nano:   * Ctrl + Oبرای ذخیره * Enter برای تایید نام فایل * Ctrl + Xبرای خروج   **۳. اجرای فایل پایتون:**  python3 test.py  خروجی:  This is the test    **دستور ls-I را اجرا کنید و کارکرد ان را شرح دهید (معنی هر کدام از ستون ها )**  **ستون 1: نوع فایل و سطح دسترسی‌ها**  drwxr-xr-x   * **کاراکتر اول (d)**: نوع فایل   + d: دایرکتوری (Directory)   + -: فایل معمولی (Regular file)   + l: لینک نمادین (Symbolic link) * **۹ کاراکتر بعدی**: مجوزهای دسترسی (permission bits)   + **r**: خواندن (read)   + **w**: نوشتن (write)   + **x**: اجرا (execute)   + تقسیم‌شده به سه بخش:     1. مالک (user/owner)     2. گروه (group)     3. سایر کاربران (others)   مثال:  rwx | r-x | r-x  owner | group | others  **ستون 2: تعداد لینک‌های سخت (Hard Links)**   * برای فایل‌ها معمولاً ۱ هست * برای دایرکتوری‌ها: تعداد لینک‌ها به آن، یعنی شامل خودش (.)، والد (..) و زیرپوشه‌ها   **ستون 3: نام مالک فایل**  rahil   * کاربری که صاحب فایل یا پوشه است   **ستون 4: نام گروه**  rahil   * گروهی که فایل به آن تعلق دارد * می‌توان چند کاربر در یک گروه داشت   **ستون 5: اندازه فایل (به بایت)**  4096   * برای فایل‌ها: حجم واقعی * برای دایرکتوری‌ها معمولاً 4096 بایت هست (ساختار داخلی)   **ستون‌های 6 تا 8: تاریخ و زمان آخرین تغییر**  Apr 7 00:42   * ماه (Apr) * روز (7) * ساعت و دقیقه (00:42)   **ستون 9: نام فایل یا پوشه**  Desktop   * نام فایل، پوشه یا لینک     **11-حالت های مختلف گرفتن و دادن مجوز را به یک فایل با دستور chmod انجام دهید و سعی کنید با nano وارد فایلی که از ان مجوز ها را گرفتید بشید ؟**  **nano testfile.txt**  کاربر فایل متنی testfile.txt را با استفاده از ویرایشگر متنی nano باز می‌کند (احتمالاً برای ایجاد یا ویرایش محتوا).  **chmod 000 testfile.txt**  این دستور تمام دسترسی‌ها را از فایل حذف می‌کند:   * هیچ‌کس (نه مالک، نه گروه، نه دیگران) اجازه‌ی خواندن، نوشتن یا اجرای فایل را ندارد.   **cat testfile.txt**  سعی داریم ک محتوای فایل را با دستور cat بخوانیم ، اما چون در مرحله قبل همه‌ی دسترسی‌ها حذف شده، با پیغام خطای زیر مواجه می‌شویم:   * cat: testfile.txt: Permission denied یعنی: اجازه‌ی دسترسی به فایل وجود ندارد   **chmod 600 testfile.txt**  با این دستور:   * **مالک ،کاربر rahil** دسترسی **خواندن و نوشتن (rw-)** را پیدا می‌کند. * **گروه و دیگر کاربران** هیچ دسترسی‌ای ندارند.   **chmod 755 testfile.txt**  این دستور دسترسی‌ها را به صورت زیر تغییر می‌دهد:   * **مالک:** خواندن، نوشتن و اجرا (rwx) * **گروه:** خواندن و اجرا (r-x) * **دیگران:** خواندن و اجرا (r-x)   معمولاً این سطح دسترسی برای فایل‌های اجرایی (scripts) استفاده می‌شود.  **chmod 444 testfile.txt**  این دستور فقط اجازه‌ی **خواندن (r--)** را برای همه (مالک، گروه، دیگران) فعال می‌کند. هیچ‌کس اجازه‌ی نوشتن یا اجرای فایل را ندارد.  A screenshot of a computer  AI-generated content may be incorrect.  **12-دستور sudo adduser[name] یوزر جدیدی ایجاد کنید و از یوزر فعلی به یوزر جدید منتقل بشید :**  **دستور sudo adduser sara:**  **این دستور برای ایجاد یک کاربر جدید به نام sara در سیستم استفاده شده است. چون از sudo استفاده شده، نیاز به دسترسی root (مدیر سیستم) دارد.**  **سیستم چندین مرحله را برای ایجاد کاربر جدید انجام داده و پیام‌هایی را نشان داده است:**   1. **info: Adding user 'sara' سیستم در حال افزودن کاربر sara است.** 2. **info: Selecting UID/GID from range 1000 to 59999 سیستم در حال انتخاب شناسه‌های عددی UID و GID برای کاربر جدید است.** 3. **info: Adding new group 'sara' (1001) یک گروه جدید به نام sara با شناسه 1001 ساخته می‌شود.** 4. **info: Adding new user 'sara' (1001) with group 'sara' (1001) کاربر sara با شناسه کاربری 1001 به همراه گروه هم‌نام ساخته می‌شود.** 5. **info: Creating home directory '/home/sara' یک پوشه شخصی برای کاربر ساخته می‌شود در مسیر /home/sara.** 6. **info: Copying files from '/etc/skel/' فایل‌های پیش‌فرض (مثل .bashrc، .profile و ...) از پوشه‌ی /etc/skel/ به پوشه‌ی خانگی کاربر کپی می‌شوند.**   **تنظیم رمز عبور:**  **سیستم از کاربر می‌خواهد رمز عبور برای کاربر جدید تعیین کند:**   * **پیغام BAD PASSWORD: The password is shorter than 8 characters نشان می‌دهد که رمز وارد شده کمتر از 8 کاراکتر بوده و از نظر امنیتی ضعیف محسوب می‌شود.** * **با این حال رمز عبور پذیرفته شده: passwd: password updated successfully.**   **وارد کردن اطلاعات کاربر:**  **بعد از ایجاد کاربر، سیستم اطلاعات اضافی درباره کاربر sara را درخواست می‌کند. مقادیر وارد شده:**  **Full Name: saadat نام کامل**   * **Room Numberشماره اتاق:1** * **Work Phone / Home Phone / Other: خالی گذاشته شده‌اند**   **در انتها، سیستم می‌پرسد:**  **Is the information correct? [Y/n]**  **یعنی: «آیا اطلاعات وارد شده درست است؟» کاربر باید Y یا n وارد کند تا تأیید یا رد کند.**  **نتیجه نهایی:**  **کاربر sara با موفقیت ایجاد شده و دارای اطلاعات شخصی اولیه و رمز عبور است. حالا این کاربر می‌تواند وارد سیستم شود و پوشه شخصی مخصوص خودش را دارد.**  A screenshot of a computer program  AI-generated content may be incorrect.  **13- یک گروه ایجاد کنید و تعدادی یوزر به آن گروه اضافه کنید ؟**  **۱. ایجاد یک گروه جدید**  برای ایجاد یک گروه جدید، از دستور groupadd استفاده میکنیم :  sudo groupadd mygroup  به جای mygroup میتوان هر نام دلخواه دیگری را وارد کرد .  **۲. ایجاد کاربران جدید (در صورت نیاز)**  اگر هنوز کاربران ساخته نشده‌اند، می‌توانیم آنها را با دستور adduser ایجاد کنیم:  sudo adduser user1  sudo adduser user2  در حین ایجاد، ازما رمز عبور و اطلاعات دیگر خواسته می‌شود. می‌توان فقط Enter بزنیم تا از مقدار پیش‌فرض استفاده شود.  **۳. اضافه کردن کاربران به گروه**  برای اضافه کردن هر کاربر به گروه، از دستور usermod استفاده میکنیم :  sudo usermod -aG mygroup user1  sudo usermod -aG mygroup user2   * گزینه -aG به معنی "اضافه کردن به گروه مکمل" است، بدون حذف گروه‌های دیگر کاربر. * mygroup نام گروه است. * user1 و user2 کاربران مورد نظر هستند.( در اینجا ما کاربر sara را اضافه کردیم )   **۴. بررسی عضویت کاربران در گروه**  برای اطمینان از اینکه کاربر عضو گروه شده، از دستور زیر استفاده میکنیم :  groups user1  **نکته: اعمال تغییرات پس از ورود مجدد**  کاربران باید **از سیستم خارج و دوباره وارد شوند (log out/in)** تا عضویت گروه جدید اعمال شود.      **14-با دستور chown مالکیت و گروه یک فایل را عوض کنید و سپس با دستور ls-l تغییر را نمایش دهید ؟**  **مرحله ۱: یک فایل میسازیم**  touch testfile .txt  **مرحله ۲: تغییر مالکیت و گروه با chown**  فرض کنیم می‌خواهیم مالک فایل را به کاربر user1 و گروه فایل را به mygroup تغییر بدهیم:  sudo chown user1:mygroup myfile.txt   * user1 → نام کاربر جدید (مالک) * mygroup → نام گروه جدید * user1:mygroup → ساختار کلی chown برای تنظیم همزمان مالک و گروه   در این جا ما از دستور زیر استفاده میکنیم :  Sudo chown 14:developers testfile.txt  **مرحله ۳: بررسی با دستور ls -l**  اکنون با دستور زیر وضعیت فایل را میبینیم :  ls -l testfile.txt  خروجی چیزی شبیه به این خواهد بود:  -rw-r--r-- 1 user1 mygroup 0 Apr 8 12:00 myfile.txt  اینجا:   * user1 → مالک فایل * mygroup → گروه فایل   خروجی آزمایش ما به نحو زیر میباشد :    **17-با توجه به ویدئو آموزشی ارسال شده هسته لینوکس را یکبار کامپایل کنید و مراحل را شرح دهید:**  **قبل از شروع باید یک سری ابزار و پکیج نصب باشد:**  **sudo apt update**  **sudo apt install build-essential libncurses-dev bison flex libssl-dev libelf-dev**  **۱. دریافت سورس‌کد هسته لینوکس:**  **می‌توان از سایت رسمی یا با استفاده از git دریافت کرد:**  **wget https://cdn.kernel.org/pub/linux/kernel/v6.x/linux-6.8.tar.xz**  **tar -xf linux-6.8.tar.xz**  **cd linux-6.8**  **یا:**  **git clone https://git.kernel.org/pub/scm/linux/kernel/git/stable/linux.git**  **cd linux**  **۲. پیکربندی هسته:**  **قبل از کامپایل باید تنظیمات دلخواه را مشخص کنیم:**  **make menuconfig**  **با استفاده از این دستور، منویی باز می‌شود که می‌توانید قابلیت‌های هسته را فعال یا غیرفعال میکنیم.**  **۳. کامپایل هسته:**  **برای ساخت هسته و ماژول‌ها:**  **make -j$(nproc)**  **استفاده از -j$(nproc) باعث می‌شود کامپایل با تعداد هسته‌های CPU سیستم انجام شود (سریع‌تر).**  **۴. نصب ماژول‌ها:**  **بعد از ساخت، باید ماژول‌های کرنل نصب شوند:**  **sudo make modules\_install**  **۵. نصب هسته:**  **اکنون می‌توان هسته را نصب کرد:**  **sudo make install**  **این دستور فایل‌های زیر را در /boot قرار می‌دهد:**   * **vmlinuz-<version>** * **System.map-<version>** * **config-<version>**   **و گراب (GRUB) را نیز به‌روز می‌کند.**  **۶. ریبوت و انتخاب هسته جدید:**  **سیستم را ریستارت میکنیم:**  **sudo reboot**  **در منوی GRUB (اگر فعال باشد) می‌توان هسته جدید را انتخاب کرد.**  **بررسی پس از بوت:**  **بعد از بالا آمدن سیستم، بررسی میکنیم که هسته جدید در حال اجراست:**  **uname -r**  **A screenshot of a computer  AI-generated content may be incorrect.**  بعد از کامپایل کردن محتوای فایل به شرح زیر میباشد:  A screenshot of a computer  AI-generated content may be incorrect.  **18-کامپایل کردن یک ماژول ، وارد کردن ان داخل هسته و load کردن ان :**  در ادامه مراحل کار شرح داده میشود :  **1-کدنویسی ماژول (Module Code)**  **ایتدا یک فایل c برای ماژول ایجاد میکنیم : hello.c**  **#include <linux/module.h>**  **#include <linux/kernel.h>**  **#include <linux/init.h>**  **MODULE\_LICENSE("GPL");**  **MODULE\_AUTHOR("Your Name");**  **MODULE\_DESCRIPTION("A simple Linux kernel module.");**  **static int \_\_init hello\_init(void) {**  **printk(KERN\_INFO "Hello, Kernel!\n");**  **return 0;**  **}**  **static void \_\_exit hello\_exit(void) {**  **printk(KERN\_INFO "Goodbye, Kernel!\n");**  **}**  **module\_init(hello\_init);**  **module\_exit(hello\_exit);**    **مرحله ۲: ایجاد فایل Makefile برای کامپایل**  **یک فایل با نام Makefile کنار فایل C ایجاد میکنیم:**  **Makefile**  **obj-m += hello.o**  **all:**  **make -C /lib/modules/$(shell uname -r)/build M=$(PWD) modules**  **clean:**  **make -C /lib/modules/$(shell uname -r)/build M=$(PWD) clean**  **مرحله ۳: کامپایل ماژول**  **در ترمینال، در همان پوشه‌ای که فایل‌ها هستند، این دستور را میزنیم:**  **make**  **خروجی باید فایلی با پسوند .ko باشد مثلاً hello.ko**    **۴: وارد کردن ماژول به هسته (Load Module)**  **sudo insmod hello.ko**    **مرحله ۵: بررسی اینکه ماژول لود شده**  **lsmod | grep hello**  **و برای دیدن پیام‌های printk:**  **Sudo dmesg | tail**  **پیام hello kernel در این مرحله مشاهده میشود :**    **مرحله ۶: خارج کردن ماژول از هسته (Unload Module)**  **sudo rmmod hello**  **و دوباره بررسی:**  **dmesg | tail sudo**  **پیام goodby kernel در این جا مشاهده میشود :**  **A screen shot of a computer  AI-generated content may be incorrect.** |