

# جلسه ی دوم گنو/ لینوکس

■ بخش اول :

آینده ی نرم افزار آزاد ( از سری سخنرانی های ریچارد استالمن )



■ بخش دوم :

آشنایی با سیستم ساختاری فایل لینوکس و درک مجوزها

# هیچ کسی کامل نیست !



■ نرم افزار غیرآزاد میگوید که شما درمانده و جدا شده اید. آن میگوید که شما نمیتوانید حتی بفهمید برنامه چکار میکند؛ شما لغت توسعه دهنده را بر آنها اطلاق می کنید؛ و اغلب آنها به شما نخواهند گفت که آن واقعا چه انجام میدهد. و اگر شما آنها دوست نداشته باشید، شما نمیتوانید آنها را تغییر دهید. حتی اگر توسعه دهنده بهترین تلاش خالصانه ی خودش را برای سودمند ساختن برنامه انجام داد، هیچکس کامل نیست. من میتوانستم برنامه ای بنویسم، و شما ممکن بود آنها برای آنچه که شما میخواهید نیمه کاره بیابید. شاید من آنها برای مقاصد ی قدری متفاوت نوشتم، نه یکسان با مقاصد شما. هیچکس نمیتواند همه چیز را پیشبینی کند. شاید من آنها به روشی انجام دادم که فکر میکردم بهترین است، اما شما ایده ی بهتری دارید. هیچکس نمیتواند همیشه همه چیز را درست درآورد.

# چهار آزادی اساسی در گنو



- نرم افزار آزاد بدین معناست که شما چهار آزادی اساسی را دارید :
- (۰) آزادی برای اجرای برنامه برای هر مقصودی است، در هر روشی که شما میخواهید.
- (۱) آزادی برای کمک کردن به خودتان بوسیله‌ی مطالعه‌ی کد منبع برای دیدن اینکه برنامه چه انجام میدهد و سپس تغییر دادن آن برای وفق دادن با نیازهای شما است.
- (۲) آزادی برای کمک کردن به همسایه‌ی شما بوسیله‌ی توزیع کپی هایی به دیگران است.
- (۳) آزادی برای کمک کردن ساختن اجتماعات بوسیله‌ی انتشار یک نسخه‌ی بهبود یافته تا دیگران بتوانند برنامه‌ی شما را در عوض استفاده کنند،

# این شما نیستید که به کامپیوترتان دستور می‌دهید!

- ما همه میدانیم که رایانه‌ها تصمیمات را واقعا خودشان نمیگیرند. آنها آنچه را که افراد بدانها گفته‌اند انجام میدهند. اما کدام افراد به آنها گفته‌اند که چکار کنند؟ وقتیکه شما در حال استفاده از رایانه‌تان هستید، آیا میتوانید به آن بگویید چکار کند، یا کس دیگری دارد به آن میگوید که چکار کند؟ چه کسی رایانه‌ی شما را کنترل میکند؟ این پرسش نرم افزار آزاد است. آزادی‌ها در تعریف نرم افزار آزاد، آزادی صفر، یک، دو و سه، علت اینکه چرا اینها آزادی‌هایی هستند که مهمند اینست که اینها آزادی‌های لازم برای شهروندان برای کنترل رایانه‌های خودشان است. شما آزادی صفر را برای قادر بودن به انجام دادن هرکاری با رایانه‌تان که میخواهید نیاز دارید. شما آزادی یک را نیاز دارید تا بتوانید نرم افزار را به کاری که شما میخواهید آن انجام دهد وا دارید. اگر شما آزادی یک را نداشته باشید، شما گیر کرده اید؛ شما یک اسیر نرم افزارتان هستید.

# اگر همه قادر به تغییر نرم افزار باشند پس سازگاری؟



واقعیست این است که کاربران سازگاری را دوست دارند. نتیجه اینست که بیشتر توسعه دهندگان نرم افزار آزاد خیلی سخت تلاش میکنند تا سازگار باشند. حدس بزنید چه اتفاق خواهد افتاد اگر من یک تفاوت ناسازگار در برنامه ام ایجاد کردم و کاربران آنرا دوست نداشتند. یک کاربری برنامه را تغییر خواهد داد و آنرا سازگار میسازد، و سپس بیشتر کاربران این نسخه را ترجیح خواهند داد. بنابراین نسخه‌ی او محبوب خواهد شد و مال من فراموش خواهد شد. حال، من البته که نمیخواهم آن اتفاق بیفتد. من میخواهم مردم نسخه‌ی مرا دوست داشته باشند و استفاده کنند، بنابراین من اینرا تشخیص خواهم داد و نسخه‌ام را از ابتدا سازگار خواهم ساخت زیرا من میخواهم مردم آنرا دوست داشته باشند. بنابراین در اجتماع ما، توسعه دهندگان نمیتوانند در برابر آنچه کاربران میخواهند مقاومت کنند. ما مجبور هستیم همراه باشیم یا کاربران به جایی که آنها میخواهند خواهند رفت و ما را در پشت سر باقی می‌گذارند.



# کسانی که واقعا ناسازگاری تحمیل می کنند!

■ اما اگر شما به توسعه دهندگان نرم افزار غیرآزاد نگاه کنید، آنهایی که خیلی قدرتمند هستند، آنها میتوانند ناسازگاری تحمیل کنند و آنها آنقدر قدرتمند هستند که کاربران نمیتوانند هیچ کاری انجام دهند. مایکروسافت برای این معروف است. آنها یک تغییر ناسازگار در یک پروتکل ایجاد میکنند، و سپس کاربران با آن گیر میکنند. اما آن تنها مایکروسافت نیست. WAP را ملاحظه کنید، برای نمونه. WAP محتوی نسخه های تغییر یافته ای از پروتکل های عادی اینترنت است، تغییر یافته برای ناسازگار شدن، و ایده این بود که آنها این تلفن ها را خواهند ساخت و خواهند گفت «آنها میتوانند روی اینترنت صحبت کنند»، اما از آنجاییکه آنها پروتکل های عادی اینترنت را استفاده نکردند، ناسازگاری بر کاربر تحمیل خواهد شد. آن طرح آنها بود. اما آن خوشبختانه کار نکرد. اما آن خطری است که شما وقتیکه کاربران واقعا کنترل ندارند با آن رودررو هستید: یک کسی تلاش خواهد کرد تا ناسازگاری بر کاربران تحمیل کند.



# اصلا میشه از نرم افزار آزاد پول درآورد؟!!

- تجارت نرم افزار آزاد پیش از این وجود دارد. در واقع، من یک تجارت نرم افزار آزاد را در ۱۹۸۵ شروع کردم. من در حال فروختن کپی های GNU Emacs بودم. من داشتم دنبال راهی برای پول درآوردن از طریق نرم افزار آزاد میگشتم. بنابراین من گفتم، «۱۵۰ دلار به من بپردازید، و من یک نوار با ادیتور متن GNU Emacs را برای شما پست خواهم کرد.» مردم شروع به پرداختن به من کردند، و من نوارها را به آنها پست کردم. من پول کافی برای زندگی کردن بر آن بدست آوردم. من این را متوقف کردم زیرا من بنیاد نرم افزار آزاد را شروع کردم، و برای بنیاد نرم افزار آزاد شروع کردن توزیع GNU Emacs مناسب بنظر رسید. من نخواستم با بنیاد نرم افزار آزاد رقابت کنم، بنابراین مجبور شدم یک راه متفاوت پیدا کنم. برای چندین سال، بنیاد از این راه پول کافی برای پرداختن به چندین کارمند درآورد، شامل برنامه نویسان. بنابراین عملا، اگر من آنرا خودم انجام داده بودم، احتمالا با فروختن کپی های نرم افزار آزاد براحتی کامیاب میشدم.

# اصلا میشه از نرم افزار آزاد پول درآورد؟! (ادامه)

- پس از آن، من تجارت نرم افزار آزاد دیگری را شروع کردم که تغییراتی را در قبال کمیون انجام میدادم. (م: تغییرات/اضافات و سفارشی سازی نرم افزار بر اساس درخواست کاربران و پرداخت از سوی آنها)
- در مورد مسئله ای که مردم بعضی وقتها مطرح میکنند: آیا ما میتوانیم احتمالا مقدار کافی نرم افزار توسعه دهیم و آن را آزاد کنیم؟ جواب هویدا است زیرا ما قبلا این کار را کرده ایم. مردمی که این پرسش را میکنند مانند اینست که میپرسند آیا هواپیماها واقعا میتوانند بالا بمانند؟ بخوبی!، من در یکی پرواز کردم. احتمالا همه ی شما همچنین در یکی از هواپیماها پرواز کرده اید. من فکر میکنم آنها میتوانند بالا بمانند. در نرم افزار آزاد امروز، ما صدها نفر را داریم، شاید هزاران، که برای توسعه ی نرم افزار آزاد دستمزد پرداخته میشوند. اما ما بیش از نیم میلیون توسعه دهندگان نرم افزار آزاد داوطلب داریم که دارند نیمه وقت کار میکنند و دستمزد پرداخته نمیشوند و دارند یک عالمه نرم افزار توسعه میدهند.

# بخش دوم : ساختار سیستم فایل لینوکس

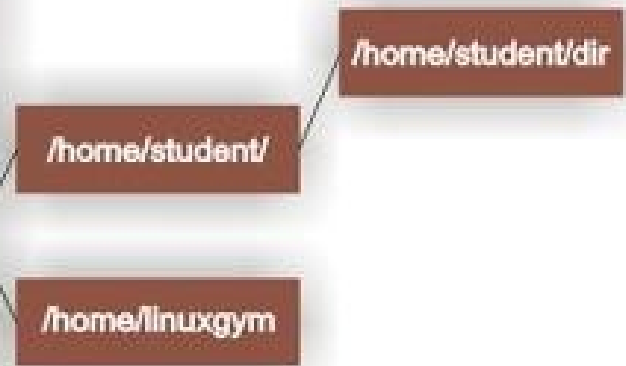
- در لینوکس همه ی اطلاعات ذخیره شده ی شما از طریق یک پوشه ی اصلی به نام root (/) قابل دسترسی است.
- موارد دیگر به صورت زیر پوشه هایی از ریشه (/) قرار می گیرند که هر کدام از این پوشه ها قسمتی از سیستم تان را لینک و یا مدیریت می کنند.
- در شکل اسلاید بعدی به طور کامل این پوشه ها را می بینید :

**ROOT DIRECTORY  
OF THE ENTIRE  
FILE SYSTEM  
HIERARCHY**

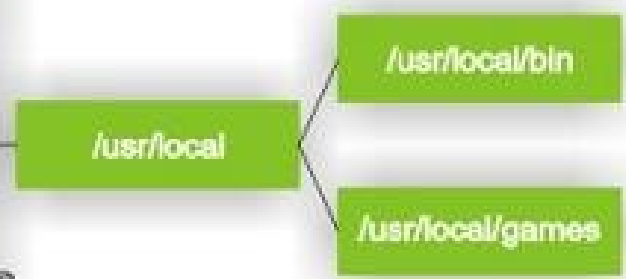
**/**

**PRIMARY HIERARCHY**

<b>/bin/</b>	<i>ESSENTIAL USER COMMAND BINARIES</i>
<b>/boot/</b>	<i>STATIC FILES OF THE BOOT LOADER</i>
<b>/dev/</b>	<i>DEVICE FILES</i>
<b>/etc/</b>	<i>HOST-SPECIFIC SYSTEM CONFIGURATION</i> <small>REQUIRED DIRECTORIES: OPT, X11, SGML, XML</small>
<b>/home/</b>	<i>USER HOME DIRECTORIES</i>
<b>/lib/</b>	<i>ESSENTIAL SHARED LIBRARIES AND KERNEL MODULES</i>
<b>/media/</b>	<i>MOUNT POINT FOR REMOVABLE MEDIA</i>
<b>/mnt/</b>	<i>MOUNT POINT FOR A TEMPORARILY MOUNTED FILESYSTEMS</i>
<b>/opt/</b>	<i>ADD-ON APPLICATION SOFTWARE PACKAGES</i>
<b>/sbin/</b>	<i>SYSTEM BINARIES</i>
<b>/srv/</b>	<i>DATA FOR SERVICES PROVIDED BY THIS SYSTEM</i>
<b>/tmp/</b>	<i>TEMPORARY FILES</i>
<b>/usr/</b>	<i>(MULTI-)USER UTILITIES AND APPLICATIONS</i> <small>SECONDARY HIERARCHY</small> <small>REQUIRED DIRECTORIES: BIN, INCLUDE, LIB, LOCAL, SBIN, SHARE</small>
<b>/var/</b>	<i>VARIABLE FILES</i>
<b>/root/</b>	<i>HOME DIRECTORY FOR THE ROOT USER</i>
<b>/proc/</b>	<i>VIRTUAL FILESYSTEM DOCUMENTING KERNEL AND PROCESS STATUS AS TEXT FILES</i>

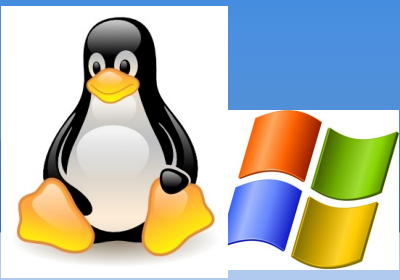


**FILESYSTEM HIERARCHY  
STANDARD ( FHS )**



# بخش دوم : ساختار سیستم فایل لینوکس

- bin : فرامین عمومی سیستم عامل لینوکس . مانند ls sort chmod
- dev حاوی نقاط دسترسی به ابزارهای سخت افزاری مانند fd cdrom hd
- etc حاوی برخی فایل های پیکربندی سیستم
- home دایرکتوری کاربران یک سیستم لینوکس
- mnt محلی برای متصل کردن ابزارها ، دیسک ها مانند سی دی و غیره
- root فولدر خانگی کاربر ریشه
- sbin دستورات مدیریتی سیستم
- tmp محل قرارگیری فایل های موقت
- usr محل قرارگیری فایل های نصب شده کاربران ، کتابخانه ها و غیره



# نکاتی برای مهاجران ویندوز!

- در لینوکس درایو نداریم! برای مثال شما برای دسترسی به فلاپی در ویندوز از درایوی به نام A استفاده می‌کنید اما در لینوکس باید از متصل کردن `mnt/floppy/` استفاده کنید (به مرور عادت خواهید کرد!)
- در ویندوز برای جدا کردن فایل‌ها و پوشه‌ها از بک‌اسلش (\) استفاده می‌کنید اما در لینوکس چه در وب و چه در آدرس دهی فایل‌ها و پوشه‌ها از اسلش (/) استفاده می‌شود. (حتی از بک‌اسلش برای نام فایل هم می‌توانید استفاده کنید)
- فایل‌ها در ویندوز همیشه دارای پسوند هستند اما در لینوکس نیازی به داشتن پسوند برای فایل‌ها نیست (خود سیستم نوع آن‌ها را تشخیص می‌دهد)
- در ویندوزهای غیر NT خبری از مجوزهای فایل نیست. ولی در لینوکس هر فایل دارای مجوزی است که به آن اضافه می‌شود.

# کاربران ، گروه‌ها و دیگران در لینوکس

- در لینوکس کلاس سه نوع موجودیت وجود دارند که می‌توانند کار انجام دهند :
- کاربر (User)
- گروه (Group)
- دیگران (Others)
- هر کاربر می‌تواند در یک گروه قرار گیرد تا در صورت اعطای مجوزها به کل کاربران موجود در آن گروه اعطا شود
- کسانی که کاربر سیستم نباشند و یا در گروه مالک نباشند ، دیگران تلقی می‌شوند که غالباً کمترین مجوزها به آنها اعطا می‌شود.

# درک مجوز فایل‌ها

- در لینوکس برای حفاظت از اطلاعات کاربران و فایل‌های سیستمی و کلا امنیت سیستم به هر فایل ۹ بیت اطلاعات اضافه می‌شود که معرف نحوه دسترسی شما و دیگران به آن فایل است.
- این بیت‌ها به صورت

`rwXrwxrwx`

مالک

نمایش داده می‌شوند.

- سه بیت اول تعیین‌کننده نحوه دسترسی مالک فایل، سه بیت دوم نحوه دسترسی گروه مالک فایل و سه بیت آخر نحوه دسترسی دیگران را مشخص می‌کند.

# درک مجوز فایل‌ها (ادامه)

- r به معنای Read یا همان دسترسی خواندن است
- w به معنای Write یا همان نوشتن است
- x به معنای Execute یا همان دسترسی اجرا روی سیستم است
- همچنین این مجوزها به صورت عددی در مبنای ۸ (اکتال) بیان می‌شوند. که به صورت زیر است: (خط تیره - به معنای نداشتن مجوز است)

0) --- no permission

1) --x execute

■ 2) -w- write

3) -wx write and execute

■ 4) r-- read

5) r-x read and execute

■ 6) rw- read and write

7) rwx read, write and execute

# برای دریافت اسلایدها و دیگر منابع

- برای دریافت این اسلاید و دیگر منابع به صفحه‌ای اختصاصی که در وبلاگم اضافه کرده‌ام مراجعه کنید:

<http://blog.itnotes.info/linux>

