



دستکاری PDF ها با PDF Toolkit

دستکاری PDF ها بوسیله PDF Toolkit

0. مقدمه

اگر پیش از این با کاغذ کارهای روزمره خود را انجام می دادید، مطمئنا بروی میز شما چسب، منگنه، پانچ، سوزن، گیره و قیچی وجود داشته؛ اگر امروز کارهای خود را به صورت دیجیتالی انجام می دهید یکی از فرمت هایی که بسیار با آن سر و کار خواهید داشت فرمت PDF یا Portable Document Format خواهد بود. فرمتی از سوی Adobe که در اکثر سیستم عامل ها و حتی تجهیزاتی مانند گوشی های موبایل و کامپیوترهای دستی مانند Palm پشتیبانی می شود. اگر از PDF استفاده می کنید pdftk برای شما حکم منگنه و چسب و پانچ و سوزن را دارد. با pdftk می توانید فایل های pdf خود را نظم ببخشید، برید، بچسبانید، فایل دیگری به آنها ضمیمه کنید و ... در این مقاله قصد دارم شما را با قابلیت های اصلی این برنامه هرچند مختصر آشنا کنم. برای کاربری پیشرفته می توانید به مستندات (مقالات، راهنماها) برنامه در سایت اصلی یا در سایت های دیگر مراجعه کنید.

1. تهیه برنامه

برنامه ی pdftk یک برنامه ی آزاد تحت مجوز GNU/GPL است [1] که به صورت رایگان قابل دریافت است. برای دریافت برنامه می توانید به سایت اصلی آن در [2] مراجعه کنید. البته بسیاری از توزیع های لینوکس مانند Gentoo، Debian و SUSE این برنامه را در مخازن یا دیسک های نصب به همراه دارند. در بخش داندلودها نیز بسته های rpm برای SUSE و Mandrake وجود دارد. اگر همکاران از لینوکس استفاده نمی کنند، می توانید این برنامه را به او نیز معرفی کنید چرا که این برنامه علاوه بر سیستم گنو/لینوکس برای سیستم عامل های Sun Solaris و Mac OS X و Microsoft Windows و FreeBSD نیز موجود است. برای دریافت فایل های نصب باینری یا دریافت سورس کد می توانید به بخش داندلود سایت مراجعه کنید. جدیدترین نسخه ی این برنامه تا زمان نوشتن این مقاله ۱.۱۲ است. برنامه ی pdf toolkit را باید در کنسول اجرا کنید، در حقیقت برنامه ای برای خط فرمان است و رابط گرافیکی ندارد و همین موضوع کار با آن را بسیار ساده تر کرده است، کفایت چند دکمه بفشارید تا PDF شما به کل تغییر کند! ساختار اصلی فرمان ها به صورت زیر است:

```
[pdftk inputfile(s) operation [options] output outputfile [password] [userpermissions]
```

2. دریافت اطلاعات درباره سند

قبل از کار با یک فایل PDF همیشه بهتر است اطلاعاتی درباره ی فایل داشته باشید، این که فایل بر اساس کدام نسخه از PDF ایجاد شده، توسط چه برنامه ای ایجاد شده و غیره. به این اطلاعات Meta-Information می گویند، ای اطلاعات در حین ایجاد توسط برنامه ی نویسنده ی PDF یا برنامه ی ایجاد کننده ی PDF در فایل PDF ذخیره می شوند. برای دیدن این اطلاعات نیاز به یک فایل متنی جدید خواهید داشت، به نمونه ی زیر توجه کنید:

```
pdftk example.pdf dump_data output info.txt $~
more info.txt $~
InfoKey: Creator
InfoValue: QuarkXPress(tm) 6.5
InfoKey: Title
InfoValue: Layout 1
InfoKey: Producer
InfoValue: QuarkXPress(tm) 6.5
InfoKey: XPressPrivate
InfoValue: %%DocumentProcessColors: Cyan Magenta Yellow Black&#10;%%DocumentCustomColors: (PANTONE 485 C)&#10;%%RGBCustomColor: .912 .137 .092 (PANTONE 485 C)&#10;%%EndComments
InfoKey: CreationDate
InfoValue: 20060306170513
PdfID0: 5e4a1ed16b804656d3e6e104af7e5ae
PdfID1: 5e4a1ed16b804656d3e6e104af7e5ae
NumberOfPages: 2
PageLabelNewIndex: 1
PageLabelStart: 1
PageLabelNumStyle: DecimalArabicNumerals
$~
```

در این نمونه اطلاعات ذخیره شده در فایل Example.pdf در فایل info.txt ذخیره خواهد شد. نیازی به ساخت info.txt از پیش نیست. این فایل را خود برنامه ایجاد خواهد کرد. با دستور cat یا more می توانید محتویات فایل info.txt را ببینید. این محتویات بسته به برنامه ی ایجاد کننده ی PDF و نیز تنظیمات کاربر می تواند متغیر باشد. برای به روز کردن اطلاعات موجود در یک فایل PDF کفایت اطلاعاتی را که نیاز به به روز رسانی دارند را در یک فایل متنی ذخیره کنید و سپس با استفاده از روش زیر آنها را به روز کنید. (فقط فیلدهایی را که می خواهید به روز کنید در فایل متنی ورودی بنویسید). در اینجا ابتدا

فایلی با محتوای نیاز به تغییر دارند ایجاد می‌کنیم:

```
InfoKey: Creator
InfoValue: TeX
InfoKey: Title
InfoValue: My Project Review
InfoKey: Producer
InfoValue: QuarkXPress(tm) 6.5
```

پس از ایجاد فایل با محتوای که قرار است تغییر کنند، ه روش زیر اطلاعات را به روز می‌کنیم:

```
pdftk Example.pdf update_info update.txt output updated.pdf $~
pdftk updated.pdf dumpdata output newinfo.txt$~
more newinfo.txt $~
InfoKey: Creator
InfoValue: TeX
InfoKey: Title
InfoValue: My Project Review
InfoKey: Producer
InfoValue: QuarkXPress(tm) 6.5
InfoKey: XPressPrivate
InfoValue: %%DocumentProcessColors: Cyan Magenta Yellow Black&#10;%%DocumentCustomColors: (PANTONE 485
C)&#10;%%RGBCustomColor: .912 .137 .092 (PANTONE 485 C)&#10;%%EndComments
InfoKey: CreationDate
InfoValue: 20060306170513
NumberOfPages: 2
PageLabelNewIndex: 1
PageLabelStart: 1
PageLabelNumStyle: DecimalArabicNumerals
$~
```

برای دیدن اطلاعات فایل‌های PDF می‌توانید از برنامه‌ی pdfinfo نیز کمک بگیرید. این برنامه نیز یک برنامه‌ی آزاد تحت مجوز GNU GPL است و در بسیاری از توزیع‌های لینوکسی مشهور مانند Debian و Gentoo یا Fedora Core به صورت پیش‌فرض نصب شده است. برای Update کردن این اطلاعات نمی‌توانید از pdfinfo استفاده کنید و باید از روش پادشده‌ی بالا یا برنامه‌های دیگر کمک بگیرید. (برای دریافت اطلاعات بیشتر درباره‌ی این برنامه به صفحات Manual آن مراجعه کنید، کافیسست در کنسول بزنید: man pdfinfo)

3. جدا کردن و چسباندن صفحات

بسیار خوب، به بخش اصلی یعنی چسباندن و جدا کردن صفحات PDF رسیدیم، با pdftk مانند یک صحاف می‌توانید کتاب الکترونیکی خود را ل ف و نشر کنید و سپس صفحاتی را که نیاز ندارید برداشته یا صفحات جدید را به آن اضافه کنید و در نهایت دوباره فایل PDF خود را سرهم کنید. بسیار خوب کار خود را با جدا کردن صفحات Pdf خود شروع می‌کنیم، برای جدا کردن صفحات فایل Example.pdf به صورت زیر عمل می‌کنیم:

```
pdftk Example.pdf burst output Page%03d.pdf $
```

بدین ترتیب فایل Example.pdf به فایل‌های Page001.pdf و Page002.pdf و ... تبدیل می‌شود. همان‌طور که حتما خودتان حدس زده‌اید %03d به برنامه می‌گوید تا شماره‌ی صفحات مندرج پس از اسامی فایل‌های جدا شده دارای سه رقم باشند. (برای نمونه 001، اگر این مقدار را به %06d تبدیل کنید ارقام بعد از اسامی مانند 000001 خواهند بود). در صورتی که می‌خواهید تمام صفحات در یک دایرکتوری به صورت مرتب جدا شده باشند می‌توانید دستور را به صورت زیر استفاده کنید:

```
~$mkdir splitted
~$ pdftk Example.pdf burst output splitted/Page%03d.pdf
~$ ls splitted/
```

حال پس از جدا کردن صفحات صفحات جدید خود را اضافه کنید و یا صفحاتی که دیگر نیازی نیست را جدا کنید، پس از این مرحله‌ی چسباندن صفحات است، مهم نیست که حتماً تمام فایل‌ها یک صفحه‌ای باشند، برای نمونه ممکن است شما بخواهید تمام فایل‌های PDF دانلود شده از مجله‌ی IranTUX را به یکدیگر بچسبانید تا یکپارچه شوند، در این صورت فایل‌های PDF شما هر کدام چند صفحه‌ای هستند، برای چسباندن صفحات به صورت زیر عمل کنید:

```
$pdftk Page001.pdf Page002.pdf Page4.pdf cat output New.pdf
```

بسیار خوب فایل New.pdf حاوی صفحات ۱ و ۲ و ۴ است، اگر بجای فایل Page001.pdf فایل pdftk.pdf و بجای فایل‌های بعدی سایر فایل‌های PDF همین شماره از مجله را قرار دهید، فایل نهایی یعنی New.pdf فایلی یکپارچه خواهد بود شامل تمام مقالات این شماره از مجله‌ی ایران‌تاکس خواهد بود.

یک سناریوی دیگر، فرض کنیم شما فردی بسیار فعال و ماهر در شغل خود هستید و رزومه‌ی کاربتان ۱۰۰ صفحه است، حال می‌خواهید صفحات ۶۰ تا ۸۵ و ۹۰ تا ۹۷ آن را برای مدیر خود بفرستید، به علاوه می‌خواهید در پایان آن نامه‌ای که در فایل mail.pdf دارید را نیز اضافه کنید، اگر بخواهید مانند بالا رفتار کنید باید تمام صفحات را جدا کنید، سپس صفحات ۶۰ و ۶۱ و ... و ۸۵ و نیز ۹۰ و ۹۱ و ۹۷ و در نهایت فایل mail.pdf را در دستور خود ذکر کنید و نتیجه را به فایل نهایی منتقل کنید، این کار زمان بسیار زیادی را هدر خواهد داد، مسلماً Sid Steward نویسنده‌ی برنامه برای این سناریو راه حلی دارد:

```
~$pdftk A=myresume.pdf B=mail.pdf cat A60-85 A90-97 B output forboss.pdf
```

نتیجه فایل forboss.pdf است که شامل صفحات ۶۰ تا ۸۵ و ۹۰ تا ۹۷ فایل رزومه و نیز تمام فایل mail.pdf است. با کمی دقت و تفکر می‌توانید روش‌های ساده‌تر و کارآمدتری برای کارهایتان بیابید.

4. پسورد و مجوزهای کاربر

یکی از ویژگی‌های فایل‌های PDF توانایی رمزگذاری است، شما می‌توانید با تعریف یک عدد رمز برای فایل PDF جلوی دسترسی افراد غیر مجاز به فایل‌تان را بگیرید. برای این کار کافیست به روش زیر عمل کنید:

```
$pdftk resume.pdf output sec_resume.pdf owner_pw 12345 user_pw 1234567890
```

فایل خروجی یعنی sec_resume.pdf یک فایل رمزگذاری شده است، باز کردن این فایل با هر برنامه‌ای نیاز به وارد کردن کلمه‌ی عبور خواهد داشت. اما دو رمز عبور برای چه؟ یکی Owner یا صاحب فایل و دیگری برای User یا کاربر فایل است. صاحب فایل می‌توانید مجوزهای دیگری که در قسمت بعد خواهیم گفت را تغییر دهد، اما کاربر فقط می‌تواند اعمالی را که به او اجازه داده شده‌اند را انجام دهد.

هر فایل PDF از سوی نویسنده یا ایجاد کننده‌ی فایل می‌تواند تعدادی مجوز دریافت کند، برای نمونه مجوز کپی محتوا یا CopyContent که به شما اجازه‌ی کپی تصاویر و متن داخل PDF را می‌دهد. مجوز چاپ یا Printing که مجوز چاپ است نیز مجوز DegradedPrinting که مجوز چاپ با کیفیت پایین است. صاحب یا Owner می‌تواند یک یا هر کدام از این مجوزها و نیز مجوزهای دیگر را تغییر دهد. به نمونه‌ی زیر توجه کنید:

```
pdftk resume.pdf output New.pdf owner_pw 12345 user_pw 1234567890 allow printing CopyContexts
```

فایل New.pdf قابل چاپ است و محتوای داخل آن را می‌توانید کپی کنید. در صورتی که می‌خواهید دو فایلی که دارای پسورد هستند را با یکدیگر ترکیب کنید می‌تواند از روش زیر بهره بگیرید:

```
$pdftk A=firs.pdf B=second.pdf input_pw A=12345 cat output firstandsecond.pdf user_pw 1234567890
```

در نمونه‌ی بالا فایل firs.pdf یک فایل رمزگذاری شده با پسورد 12345 است که با فایل second.pdf ترکیب نمودیم و در نهایت فایل firstandsecond.pdf را با رمز 1234567890 ایجاد کردیم.

5. سخن نهایی

برنامه‌های بسیاری رد سیستم عامل‌های مختلف برای کار با PDF ها وجود دارند، برنامه‌هایی مانند Acrobat Reader یا TeX (بخوانید تخ). یادگیری و استفاده از برنامه‌های غیر آزادی مانند Adobe Acrobat چندان سخت نیست اما اگر برای کارهای خود به برنامه‌ای آزاد و همه فن حریف نیاز دارید

می‌توانید TeX را آغاز کنید، این زبان بیشتر شبیه به یک زبان برنامه نویسی برای ایجاد فایل‌های PDF است و نتایج فوق‌العاده‌ای به ارمغان می‌آورد. اما اگر فرصتی برای یادگیری ندارید و یا نمی‌خواهید خود را درگیر مسائل پیچیده کنید، PDFTK یکی از ابزارهای ساده و کم‌حجم است که کارهای اصلی را برای شما انجام می‌دهد. مسلماً با این ابزار نمی‌توانید محتویات فایل‌های PDF را ایجاد و یا ویرایش کنید، کاری که با TeX به ظرافت و با قدرت بسیاری قابل انجام است.

به نقل از سایت گنو ایران

بستن این پنجره