
آموزش Linux Essentials

علی رشیدی

۱۳۹۵/۰۹/۲۰

ارائه شده توسط

وبلاگ علی رشیدی

<http://Ali-RNT.blog.ir>

برای آموزش‌های بیشتر به وبلاگ مراجعه کنید.

قسمت ۱۳

عبارات باقاعده: اضافیات + جستجوی متن با grep

علی رشیدی

بها: نشر رایگان این آموزش

۱۳.۱ عبارات باقاعده: اضافیات

توضیحات قسمت قبلی تقریباً با تمامی برنامه‌هایی که از عبارات باقاعده پشتیبانی میکنند مطابقت دارد. برنامه‌های مختلف به عبارات باقاعده بسط (extension) هایی اضافه میکنند تا کارایی برنامه هارا افزایش دهند. امثال آن زبان‌های اسکریپتی Perl, Tcl و پایتون هستند، که قابلیت‌هایی فراتر از عبارات باقاعده دارند.

برخی از بسط‌های متداول:

- **براکت های کلمات:** < با آغاز یک کلمه و > با پایان یک کلمه مطابقت میکند (یک کلمه با حرفی که قبل از آن کاراکتری غیر از حروف آمده آغاز، و با حرفی که بعد از آن کاراکتری غیر از حرف آمده پایان میابد).

- **دسته بندی:** توسط پرانتزها میتوانید عبارات را به قسمت‌های مجزا تقسیم کنید. مثلاً

`a(bc)*`

با یک حرف a که به دنبال آن هر چند تا bc آمده مطابقت دارد.

- **جایگزینی:** با خط عمودی | میتوانید چند عبارت دلخواه را تعیین کنید تا انتخاب شوند. مثلاً:

`motor (bike|cycle|boat)`

با `motor bike`, `motor cycle` و `motor boat` مطابقت میکند اما نه هیچ چیز دیگر.

- **عبارت اختیاری:** با علامت سؤال تعین میکنید که یک عبارت میتواند وجود نداشته باشد یا یک بار وجود داشته باشد. مثلاً:

`Ali (Rashidi)?`

با `Ali` و `Ali Rashidi` مطابقت میکند.

- **حداقل یکبار تکرار:** با استفاده از + به جای * درواقع همان عمل * را انجام میدهید، اما اینبار حداقل یکبار باید عبارت تکرار شود. مثلاً

`a(li)*`

`a(li)+`

عبارت اول با `a` و `ali` و `alili` و ... مطابقت دارد، در حالی که دومی با `a` مطابقت نمیکند.

- **تعیین مرتبه تکرار:** با استفاده از براکت میتوان تعیین کرد یک متن چند بار تکرار شود. مثلاً:

`ab{2,4}`

با `abb`، `abbb` و `abbbb` مطابقت دارد. میتوانید حداقل یا حداکثر را حذف کنید. در صورت حذف حداقل، ۰ در نظر گرفته می‌شود و در صورت حذف حداکثر، بینهایت.

- **ارجاع به قبل:** عبارتی مانند `\n` تکرار مرتبه `n`ام عبارت بلی که در پرانتز آمده را سبب میشود. (از تعریف که هیچی معلوم نیست) برای مثال:

`(ab)\1`

با `abab` مطابقت دارد، چرا که مرتبه ۱ پرانتز قبلی همان `ab` است. و عبارت زیر:

`(aba*d)x\2`

ابتدا ببینیم عبارت داخل پرانتز با چه چیزی جور میشود. در مرتبه ۱ با `abaad` و در مرتبه ۲ با `abaaad` جور میشود. یک `x` نیز به آن اضافه میشود. `\2` یعنی مرتبه دوم پرانتز و در نهایت این عبارت با این دو رشته جور میشود:

abaadxabaaad
abaaadxabaaad

اطلاعات بیشتر در راهنمای GNU grep موجود است.

- **عملگرهای غیر حریص (Non-greedy):** عملگرهای + و ؟ رفتاری حریصانه دارند، یعنی سعی میکنند با بیشترین مقدار ورودی که ممکن است جور شوند. مثلاً اعمال عبارت a^*a به رشته ورودی abacada فقط با abacada جور میشود، نه با aba یا abaca. برای اعمال محدودیت میتوان از عملگرهای غیر حریص

*?

+?

??

استفاده کرد که سعی میکنند کمترین ورودی ممکن را بگیرند. در مثال قبلی اگر از

$^a.*?a$

استفاده کنیم با رشته ی

aba

جور میشود.

توجه کنید که تمامی برنامه‌ها از این بسط‌ها پشتیبانی نمیکنند. جدول زیر را ببینید:

Tcl	Perl	emacs	vim	trad egrep	GNU egrep	GNU grep	بسط (Extension)
.4	.4	.1	.1	.	.	.	براکت های کلمات
.	.	.1	.1	.	.	.1	دسته بندی
.	.	.1	.2	.	.	.1	جایگزینی
.	.	.	.3	.	.	.1	عبارت اختیاری
.	.	.	.1	.	.	.1	حداقل یکبار تکرار
.	.	.1	.1	o	.	.1	تعیین مرتبه تکرار (اعمال محدودیت)
.	.	.	o	.	.	o	ارجاع به قبل
.	.	.	.4	o	o	o	عملگر های غیر حریص

o : پشتیبانی نشده، •: پشتیبانی شده.

1. به یک \ قبل از عملگر نیاز است، مثلاً ab^+ به جای $ab+$
2. به پرانتز نیازی ندارد، عبارات جایگزین همواره به یک عبارت باقاعده کامل اشاره دارند.
3. از $=$ به جای ؟ استفاده شود.
4. شیوه نوشتاری کاملاً متفاوتی دارند، مستندات را مطالعه کنید.

حالا وقت آن رسیده که بدانید چگونه از این عبارات باقاعده استفاده کنید.

۱۳.۲ جستجوی متن در فایل‌ها - grep

معروف‌ترین و مهم‌ترین برنامه‌ی جستجو که از عبارات باقاعده استفاده میکند، در لینوکس، grep است. این برنامه خطوط فایل را می‌گردد و هر خطی را که با عبارت جور شود به خروجی می‌برد. باقی خطوط نادیده گرفته میشوند.

دو گونه از grep وجود دارد: نسخه قدیمی و ابتدایی به نام fgrep که از عبارات باقاعده پشتیبانی نمی‌کند و به رشته‌های کاراکتری محدود میشود. اما یک مزیت مهم دارد که سرعت آن است. گونه‌ی دیگر، egrep است (Extended grep) که عبارات باقاعده را نیز پشتیبانی میکند اما کمی کندتر است.

نحوه نوشتاری grep به این صورت است که به حداقل یک عبارت باقاعده برای جستجو نیاز دارد. آرگومان دوم نام فایلی است که مورد جستجو واقع میشود. اگر تعیین نشود، ورودی استاندارد در نظر گرفته میشود. (در آینده توضیح داده خواهد شد).

هنگام استفاده از grep عملگرهای +، ؟ و | با یک ممیز شناور قبل از خود همراه می‌شوند (در egrep اینکار ضروری نیست). متأسفانه، عملگرهای غیر حریص در grep وجود ندارند.

نکته: عبارات باقاعده را در نقل و قول‌ها محدود کنید تا از گسترش آن‌ها توسط شل جلوگیری کنید. مخصوصاً اگر عبارت پیچیده و دارای عبارات مشابه الگوهای جستجو میباشد.

ابزار grep هم مانند هر ابزار دیگری میتواند گزینه (option) هایی را از طریق خط فرمان بگیرد که در جدول زیر مشاهده میکنید:

گزینه	نتیجه
-c (count)	فقط تعداد مطابقت‌ها را به خروجی می‌برد.
-i (insert)	میان حروف کوچک و بزرگ تفاوتی قائل نمیشود.
-l (list)	فقط نام فایل‌هایی که مطابقت دارند را لیست میکند.
-n (number)	تعداد خطوطی که مطابقت دارند را نیز به خروجی می‌برد.
-r (recursive)	فایل‌ها را در زیر شاخه (subdirectory) ها نیز جستجو میکند.
-v (invert)	خطوطی را به خروجی می‌برد که با عبارت مطابقت ندارند.

با استفاده از گزینه -f میتوانی فایلی شامل عبارت باقاعده را تعیین کنی. اگر فایل شامل چندین خط باشد هر خط به عنوان یک عبارت باقاعده مجزا در نظر گرفته می‌شود و این عبارات به صورت همزمان جستجو میشوند. ذخیره کردن عبارات در فایل راهکار خوبی برای جستجوی الگوهای است که مکرراً استفاده میشوند.