
آموزش Linux Essentials

علی رشیدی

۱۳۹۵/۰۷/۱۰

ارائه شده توسط

وبلاگ علی رشیدی

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

برای آموزش‌های بیشتر به وبلاگ مراجعه کنید.

قسمت ۱۳

عبارات باقاعده

علی رشیدی

بها: نشر رایگان این آموزش

عبارات باقاعده: مقدمات

دستورات لینوکس معمولاً برای انجام عملیاتی بر روی متن استفاده میشوند. قدرتمندترین ابزار برای انجام این عملیات و اعمال الگوها بر روی متن فایل‌ها، عبارات باقاعده است. عبارات باقاعده، رشته‌هایی هستند که برای جستجو در متن استفاده میشوند. در نگاه اول، آن‌ها شبیه الگوهای جستجوی فایل به نظر میرسند، که در قسمت‌های قبلی آموزش دادیم. اما این عبارات به طرز متفاوتی کار میکنند و امکانات بیشتری دارند.

اینکه چرا باید کسی با این عبارات کار کند مشخص است. ما قصد داریم یک کاربر حرفه‌ای لینوکس شویم که میتواند یک برنامه نویس باشد که در حین کار قطعاً با پردازش متن مواجه میشود، یا یک کاربر که نیاز به جستجو و مطابقت متن دارد. همه‌ی این کارها به عبارات باقاعده مربوط است و پیشنهاد می‌شود حتماً این قسمت را فراگیرید.

عبارات باقاعده معمولاً به صورت بازگشتی، و از عباراتی ایجاد می‌شوند که خودشان هم باقاعده اند. ساده‌ترین عبارات باقاعده حروف، کاراکترها، ارقام و خیلی چیزهای دیگر هستند که به جای خودشان قرار می‌گیرند (به معنی خودشان هستند). برای مثال، a ، یک عبارت باقاعده است که با کاراکتر a جور میشود.

کلاس‌های کارکتری، که قبلاً کمی در مورد آن‌ها بحث کردیم، با روشی مشابه الگوهای جستجوی shell ایجاد میشوند. بنابراین عبارت باقاعده $[a-e]$ (که یک محدوده (range) نیز نامیده میشود) دقیقاً با یک کاراکتر از a تا e مطابقت میکند، و $a[xy]b$ با یکی از دو رشته axy یا ayb مطابقت میکند. میتوان این رشته‌ها را به طور مسلسل وار نوشت، مانند $[A-Za-z]$ که با تمامی حروف کوچک و بزرگ مطابقت میکند.

مکمل محدوده‌ها اما با روشی متفاوت از shell نوشته میشود. در عبارات باقاعده، برای مثال، محدوده abc با تمامی حروف به جز a ، b ، c مطابقت میکند. در صورتی که در shell مینوشتیم $[!abc]$. نقطه، هم ارز علامت سؤال در الگوهای جستجوی shell است، یعنی با دقیقاً یک حرف مطابقت میکند. تنها کاراکتری که با آن مطابقت نمیکند خط جدید (\n) است. پس $a.c$ با abc و a/c مطابقت دارد، اما نه با

a

c

علت این است که اغلب برنامه‌ها بر مبنای رشته‌های یک خطی کار میکنند، چرا که پردازش چند خط کار را سخت تر میکند. اگر چه گاهی اوقات کار با چند خط آسانتر است.

تفاوت دیگر عبارات باقاعده و کلاس‌های کاراکتری، این است که کلاس‌های کاراکتری با ابتدای نام فایل مقایسه میشدند، اما عبارات باقاعده با هر قسمتی از یک خط میتواند مطابقت داشته باشد. البته میتوانید اینرا محدود کنید، اگر میخواهید فقط ابتدای خط بررسی شود ابتدای عبارت $^$ و اگر میخواهید انتهای خط بررسی شود در انتهای عبارت، $$$ قرار دهید. در ضمن در صورت استفاده از $$$ نیازی به اشاره به کاراکتر خط جدید (\n) نیست. یعنی برای جستجوی خطوطی که با XYZ تمام می‌شوند میتوان نوشت $XYZ$$ و نیازی به $XYZ\n$$ نیست.

نکته: $^$ و $$$ درواقع با کاراکترهای مخفی مطابقت میکنند که ابتدا و انتهای خط را مشخص میکنند.

از $*$ نیز میتوان برای تکرار عبارت باقاعده‌ی پیش از آن استفاده کرد. یک مثال میزنیم. الگوی جستجوی $a*.txt$ را در نظر بگیرید که با تمامی فایل‌هایی که با a شروع و با $.txt$ خاتمه می‌یافتند مطابق بود. عبارت باقاعده‌ی معادل آن این‌گونه است:

$^a.*\n.txt$

a میگوید که هر رشته‌ای که با a شروع شود موردنظر ماست. همچنین $^$ مشخص میکند که جستجو از ابتدای خط شروع شود.

سپس میخواهیم قسمت ثابت متن یعنی $.txt$ را مشخص کنیم. اما نمینویسیم $.txt$. چرا که در عبارات باقاعده . معادل «هر رشته‌ای»

بود. پس آن را با گذاشتن دو \, escape میکنیم.
