

Linux username

Standard Linux/Unix username — POSIX rules.

```
^[a-z_][a-z0-9_-]*[$]?$
```

Token-by-token breakdown

Token	Meaning
^	start of string (or line in multiline mode)
[a-z_]	any of: lowercase letters, “_”
[a-z0-9_-]*	zero or more: any of: lowercase letters, digits, “_”, “-”
[\$]?	optional (zero or one): any of: “\$”
\$	end of string (or line in multiline mode)

Quick usage in 7 languages

JavaScript	<pre>const re = /^[a-z_][a-z0-9_-]*[\$]?\$/; re.test(value);</pre>
Python	<pre>import re re.match(r"^[a-z_][a-z0-9_-]*[\$]?\$", value)</pre>
Java	<pre>Pattern.compile("^[a-z_][a-z0-9_-]*[\$]?").matcher(value).matches();</pre>
C# / .NET	<pre>Regex.IsMatch(value, @"^[a-z_][a-z0-9_-]*[\$]?");</pre>
Go	<pre>regexp.MustCompile(`^[a-z_][a-z0-9_-]*[\$]?`).MatchString(value)</pre>
Ruby	<pre>/^[a-z_][a-z0-9_-]*[\$]?\$/ .match?(value)</pre>
PHP	<pre>preg_match('~^[a-z_][a-z0-9_-]*[\$]?~', \$value);</pre>

Common pitfalls

- **Anchored to the whole string.** This pattern uses ^ and \$, so it requires the entire input to match. To find it inside a longer text, drop the anchors and use the global (g) flag.
- **Lowercase only.** Ranges like [a-z] won't match uppercase. Add A-Z (or the i flag) if uppercase input should be accepted.
- **Validate beyond format.** Matching the format doesn't guarantee the value is real. Confirm the linux username against a source of truth (database, API, or checksum) where it matters.