|  |  |
| --- | --- |
| sum by(add) (rate({job="varlogs”, instance=~"$instance"} | Hiermit wird die Aufsummierung der Benutzernamen definiert, die wir mit “Patterns” in LogQL definiert haben. “Patterns” ermöglichen die einfache Extrahierung von Informationen aus einer Zeile. Wir holen alle Log-Einträge, die sich auf den Job “varlogs” beziehen. Wir können auch nach spezifischen Endpoint filtern, indem wir das Schlüsselwort „instance“ benutzen. |
| | | „|“ funktioniert in LogQL wie eine Pipeline für die Verkettung von mehreren Suchmustern. |

|  |  |
| --- | --- |
| |= `**sshd**[`  |= `: **Failed**` | Regular Expression für die Suche nach Zeilen mit diesen Einträgen. |
| !~ `invalid user`  !~ `test` !~ `10.0.2.15` | Regular Expression für die Suche nach Zeilen **ohne** diese Einträge. Wir können beispielsweise Einträge ausschließen, die sich nicht bösartige Nutzer sind, um falsche Positive zu vermeiden |
| | pattern `<\_> for <**Benutzername**> from <**QuelleAddress**> port <\_>` [$\_\_range])) | Die Definition der Wörter “Benutzername” “ QuelleAddress” und als “Pattern” dienen dazu, einen Benutzernamen und eine Quelle IP-Adresse aus der Logdatei zu extrahieren. Die Platzhalter “<\_>” sind unbenannte Elemente, die in diesem Fall auf die Einträge “password” und Portnummber in der Zeile verweisen. |

|  |  |
| --- | --- |
| sum by(add) (rate({job="varlogs”, instance=~"$instance"} | Hiermit wird die Aufsummierung der Benutzernamen definiert, die wir mit “Patterns” in LogQL definiert haben. “Patterns” ermöglichen die einfache Extrahierung von Informationen aus einer Zeile. Wir holen alle Log-Einträge, die sich auf den Job “varlogs” beziehen. Wir können auch nach spezifischen Endpoint filtern, indem wir das Schlüsselwort „instance“ benutzen. |
| | | „|“ funktioniert in LogQL wie eine Pipeline für die Verkettung von mehreren Suchmustern. |
| |= `**sshd**[`  |= `: **Failed**` | Regular Expression für die Suche nach Zeilen mit diesen Einträgen. |
| !~ `invalid user`  !~ `test` !~ `10.0.2.15` | Regular Expression für die Suche nach Zeilen **ohne** diese Einträge. Wir können beispielsweise Einträge ausschließen, die sich nicht bösartige Nutzer sind, um falsche Positive zu vermeiden |
| | pattern `<\_> for <**Benutzername**> from <**QuelleAddress**> port <\_>` [$\_\_range])) | Die Definition der Wörter “Benutzername” “ QuelleAddress” und als “Pattern” dienen dazu, einen Benutzernamen und eine Quelle IP-Adresse aus der Logdatei zu extrahieren. Die Platzhalter “<\_>” sind unbenannte Elemente, die in diesem Fall auf die Einträge “password” und Portnummber in der Zeile verweisen. |