Regular expression filters

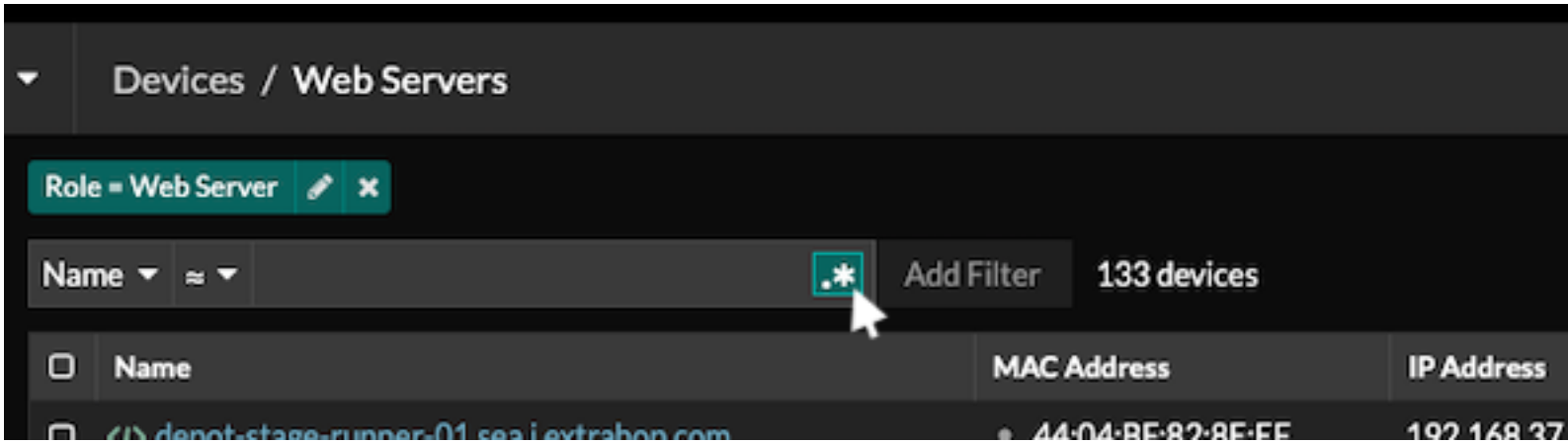
Published: 2021-01-20

Filter your search results by writing regular expression (regex) strings in certain search fields throughout the ExtraHop system. For example, you can filter for parameters in a detail metric key, such as a number within an IP address. You can also filter by excluding specific keys or a combination of keys from charts.

Regex-capable search fields have visual indicators throughout the system and accept standard syntax.

Search fields with an asterisk

Click the asterisk to enable regex strings.

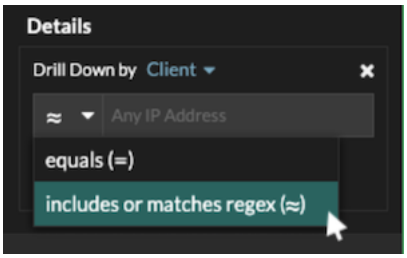


This type of field is available from the following system pages:

- Filtering a table of devices
- Creating filter criteria for a dynamic device group

Certain search fields with a trifield operator

Click the operator drop-down to select the regex option.

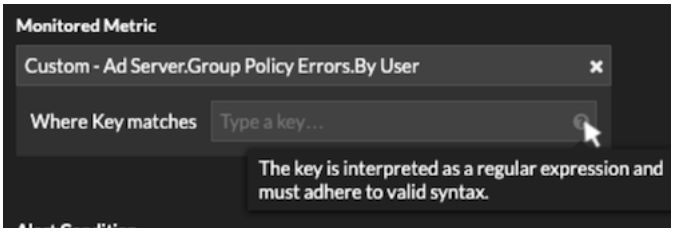


This type of field is available from the following system page:

- Editing a chart in Metric Explorer

Certain search fields with a tooltip

Hover over the tooltip in the field to see when regex is required.



This type of field is available from the following system page:

- Adding record relationships to a custom metric

The following table includes examples of standard regex syntax.

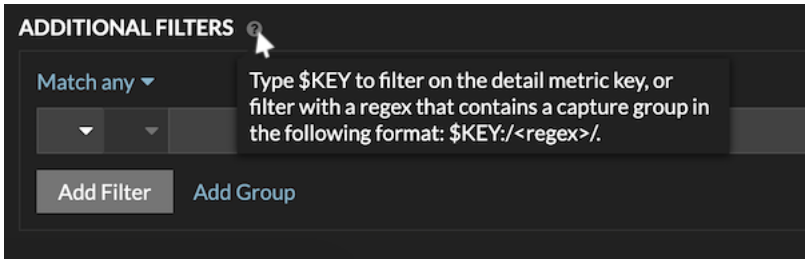
Chart Scenario	Regex filter	How it works
Compare HTTP status codes `200` to `404`.	`(200|404)`	The vertical bar symbol (	) is the OR operator. This filter matches `200`, or `404`, or both status codes.
Display any HTTP status code that contains a `4`.	`[41]`	Square brackets ([and]) designate a range of characters. The filter searches for every character inside the brackets, regardless of order. This filter matches any value that contains a `4` or a `1`. For example, this filter can return `204`, `400`, `101`, or `201` status codes.
Display all `500`-level HTTP status codes.	`^[5]`	The caret symbol (^) outside square brackets ([and]) means "starts with." This filter matches any value that begins with a `5`. For example, this filter can return `500` and `502` status codes.
Display all `400` and `500`-level HTTP status codes.	`^[45]`	Multiple values inside square brackets ([and]) are searched individually, even when preceded by the caret symbol (^). This filter does not search for values that begin with `45`, but matches all values that begin with a `4` or `5`. For example, this filter can return `400`, `403`, and `500` status codes.
Display any HTTP status codes except `200`-level status codes.	`^(?!2)`	A question mark (?) and exclamation point (!) inside parentheses specify a value to exclude. This filter matches all values except values beginning with a `2`. For example, this filter can return `400`, `500`, and `302` status codes.
Display any IP address with a `187`.	`187.`	Matches `1`, `8`, and `7` characters in the IP address. This filter will not return IP addresses that end

Chart Scenario	Regex filter	How it works
		in 187 because the trailing period designates that something must come after the values. If you want to search the period as a literal value, you must precede it with a backslash (\ ).
Review all IP addresses containing `187.18.`	`187\.18.`	Matches `187.18` and anything that follows. The first period is treated literally because it is preceded by a backslash (\ ). The second period is treated as a wildcard. For example, this filter returns results for `187.18.0.0`, `180.187.0.0`, or `187.180.0.0/16`. This filter does not return an address that ends with `187.18`, because the wildcard requires that characters follow the specified values.
Display any IP address except `187.18.197.150`.	`^(?!187\.18\.197\.150)`	Matches anything except `187.18.197.150`, where `^(?!)` specifies the value to exclude.
Exclude a list of specific IP addresses.	`^(?!187\.18\.197\.15[012])`	Matches anything except `187.18.197.150`, `187.18.197.151`, and `187.18.197.152`, where `^(?!)` specifies the value to exclude and the square brackets ([and]) specify multiple values.

Additional filters

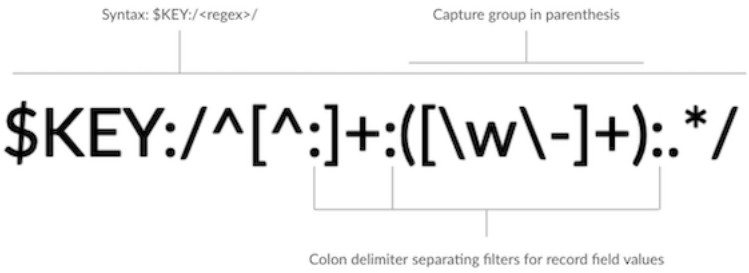
When you create a custom detail metric from the Metric Catalog, you can add advanced regex syntax to the Additional Filters search field in the Record Relationships section.

The tooltip appears after you select **Detail Metric** and is not available when **Base Metric** is selected.



The regex syntax in this field must meet the following requirements:

- If your key contains multiple values, your regex syntax must include a single capture group. A capture group is designated by parenthesis. Your capture group determines the filter value.



- If you want to return a specific value from a detail metric key that contains multiple record field values, the regex must follow this syntax:

  $KEY : /*<regex>*/

  For example, if your detail metric key is ipaddr:host:cipher and you only want to return the IP address value, you would type the following:

  $KEY : /^([^:]+):.+/
- If your key contains multiple record field values, the values are separated by a delimiter that is specified in the trigger that is generating the key. The placement of the delimiters in your regex syntax must match the delimiters in the detail key. For example, if you have a key with three values that are separated by a delimiter that is a colon, the three values for the key in your regex syntax must be separated by two colons.

  Tip: If you want to return all record field values in a detail metric key, type $KEY. For example, if your detail metric key is ipaddr:host:cipher, type $KEY in the search field to return all three of those field record values (IP address, hostname, and SSL cipher suite).