How To Use Custom Queries In Nagios Network Analyzer 2026

Purpose

This documentation will show you how to use Nagios Network Analyzer 2026 to turn existing flow data into meaningful information. This manipulation will not destroy your data at all, so feel free to experiment, as there is no chance at all that you will break anything. You will need to have an existing source with flow data to be able to follow the examples in this documentation.

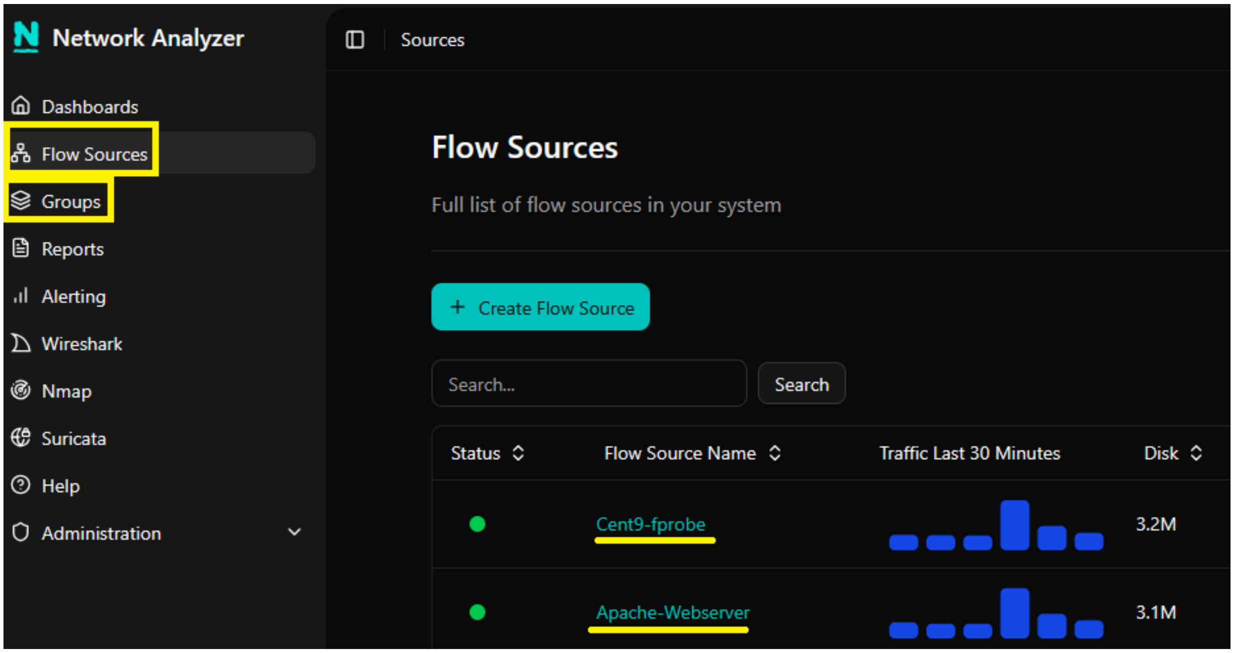
Terminology

The following terms will be used throughout this document:

- `src` - Source
- `dst` - Destination
- `src ip` - Source IP is the IP Address the traffic originated from
- `dst ip` - Destination IP is the IP Address the traffic is going to
- `src port` - Source Port is the network port the traffic is transmitted on
- `dst port` - Destination Port is the network port the traffic is received on

Performing A Query

To begin, click either **Flow Sources** or **Groups**, then click a source or group name:



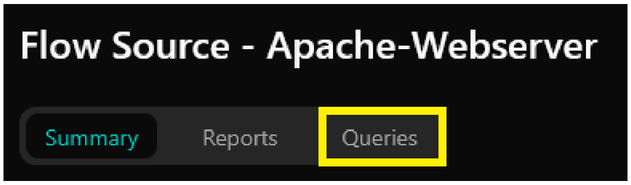
Flow Sources

Full list of flow sources in your system

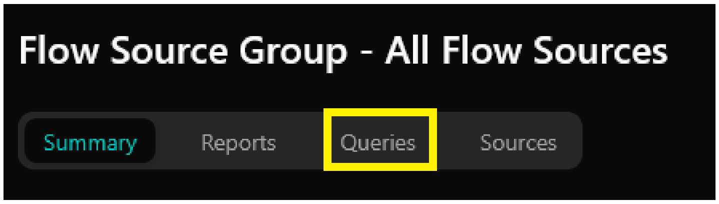
Status	Flow Source Name	Traffic Last 30 Minutes	Disk
●	Cent9-fprobe		3.2M
●	Apache-Webserver		3.1M

This will bring you to the source or source group **Summary** page. Next, click the **Queries** tab:

Source:



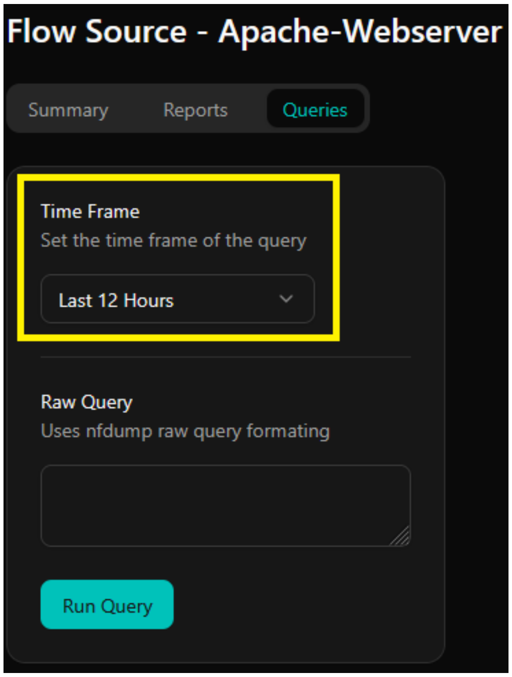
Source Group:



This is where we will be doing the majority of work and explanations in this document and will most likely be the entry point for any deep diving you do into the flow data. On this page, you'll see many fields. This section will give a description of what each one is for, and how to use it.

Time Frame

This is where you set the time frame for the query. Simply select a timeframe from the dropdown from the last 2 hours to the last month. You can also set custom elapsed time frames, to specify something like 3 hours ago until now.



Time Frame
Set the time frame of the query
Last 12 Hours

Raw Query
Uses nfdump raw query formating

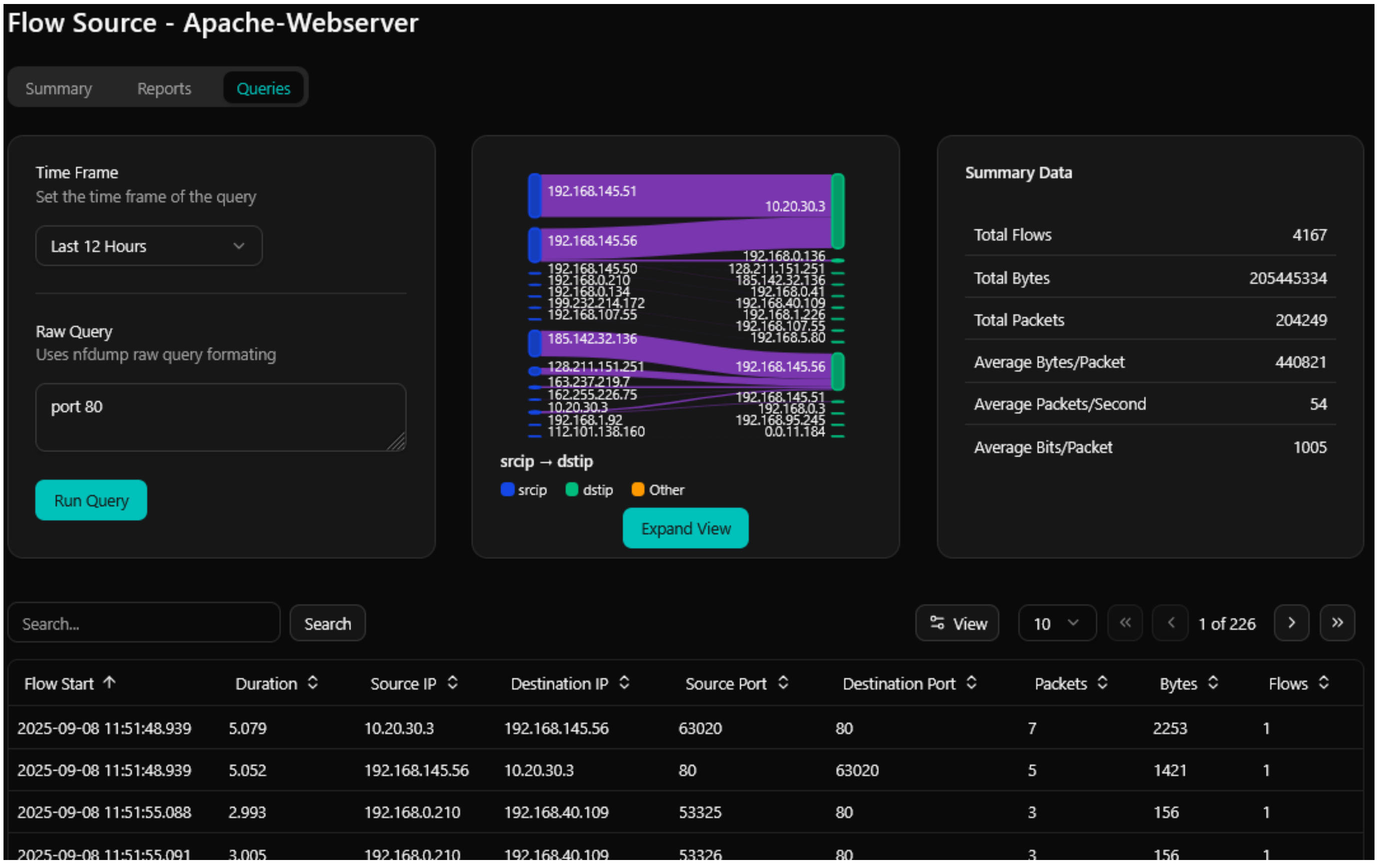
Simple Queries

The **Raw Query** field is the most powerful tool when querying data. In this field you will enter a query string to sort through the data, and if you have ever used `tcpdump` before, this section will be familiar. In this query string you can specify quite a few parameters to limit what is shown to you and chain parameters together to isolate exactly what you'd like to see. Let us assume that we would only like to see traffic on port 80. It doesn't matter if it is coming from port 80, or if it is going to port 80. In the **Raw Query** box, you would type:

```
port 80
```

...then click **Run Query**.

Here is an example of what that looks like:



Flow Source - Apache-Webserver

Time Frame
Set the time frame of the query
Last 12 Hours

Raw Query
Uses nfdump raw query formating
port 80

Summary Data

Total Flows	4167
Total Bytes	205445334
Total Packets	204249
Average Bytes/Packet	440821
Average Packets/Second	54
Average Bits/Packet	1005

Search...

10

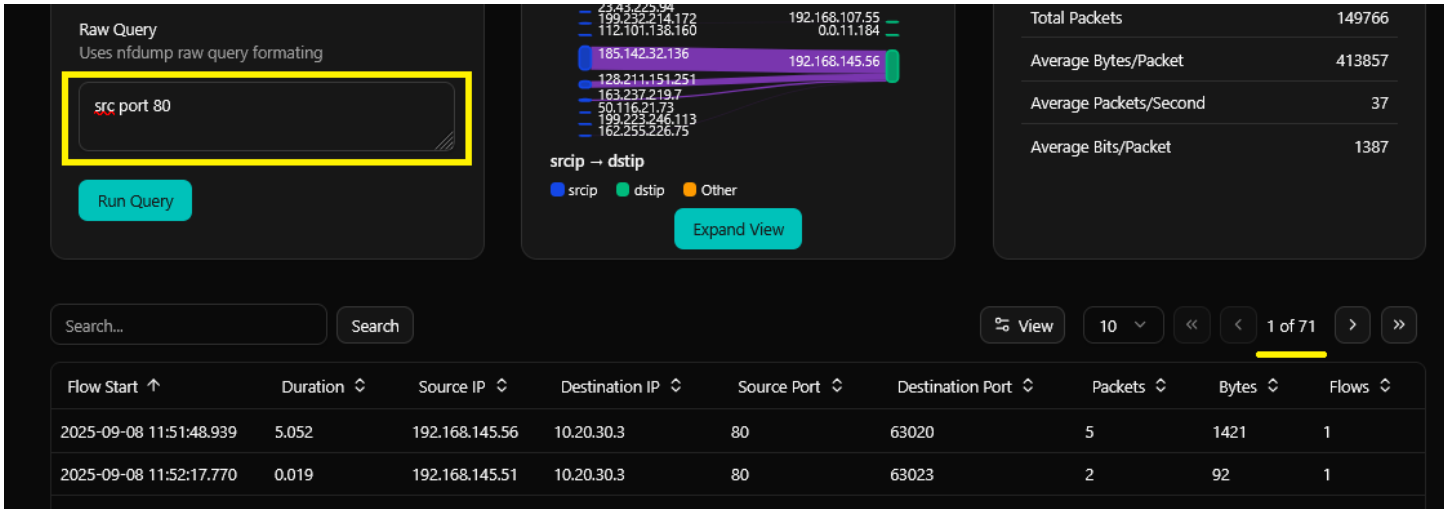
Flow Start	Duration	Source IP	Destination IP	Source Port	Destination Port	Packets	Bytes	Flows
2025-09-08 11:51:48.939	5.079	10.20.30.3	192.168.145.56	63020	80	7	2253	1
2025-09-08 11:51:48.939	5.052	192.168.145.56	10.20.30.3	80	63020	5	1421	1
2025-09-08 11:51:55.088	2.993	192.168.0.210	192.168.40.109	53325	80	3	156	1
2025-09-08 11:51:55.091	3.005	192.168.0.210	192.168.40.109	53326	80	3	156	1

This shows all of the flows to or from port 80, both `dst ip` and `src ip`.

Notice how many pages of entries this returns, 226 pages times 10 entries per page gives us 2,260 entries! If you scroll through them, you'll notice they all have port 80 as one of their ports, this is because you see traffic in both directions. If you only want to see when the source port is 80 amend the query to:

```
src port 80
```

Now the query will be limited to the source port, which drops us down to 710 entries:



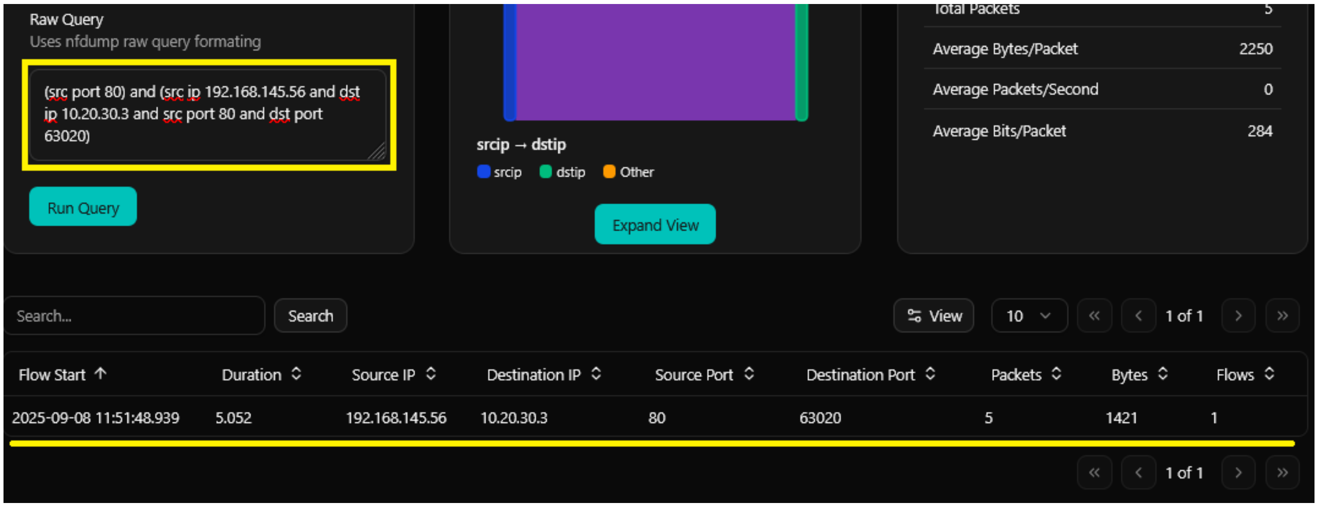
Raw Query
Uses nfdump raw query formating

src port 80

Total Packets	149766
Average Bytes/Packet	413857
Average Packets/Second	37
Average Bits/Packet	1387

Flow Start	Duration	Source IP	Destination IP	Source Port	Destination Port	Packets	Bytes	Flows
2025-09-08 11:51:48.939	5.052	192.168.145.56	10.20.30.3	80	63020	5	1421	1
2025-09-08 11:52:17.770	0.019	192.168.145.51	10.20.30.3	80	63023	2	92	1

Click any entry in the table to drill down further into the query. This will populate the **Raw Query** box with a new query based on your selection.



Raw Query
Uses nfdump raw query formating

(src port 80) and (src ip 192.168.145.56 and dst ip 10.20.30.3 and src port 80 and dst port 63020)

Total Packets	5
Average Bytes/Packet	2250
Average Packets/Second	0
Average Bits/Packet	284

Flow Start	Duration	Source IP	Destination IP	Source Port	Destination Port	Packets	Bytes	Flows
2025-09-08 11:51:48.939	5.052	192.168.145.56	10.20.30.3	80	63020	5	1421	1

Advanced Queries

We started with a very simple queries of port 80 and then src port 80. The raw query generated by clicking the table entry was far more complex, including an `and` operator and grouping items in parenthesis:

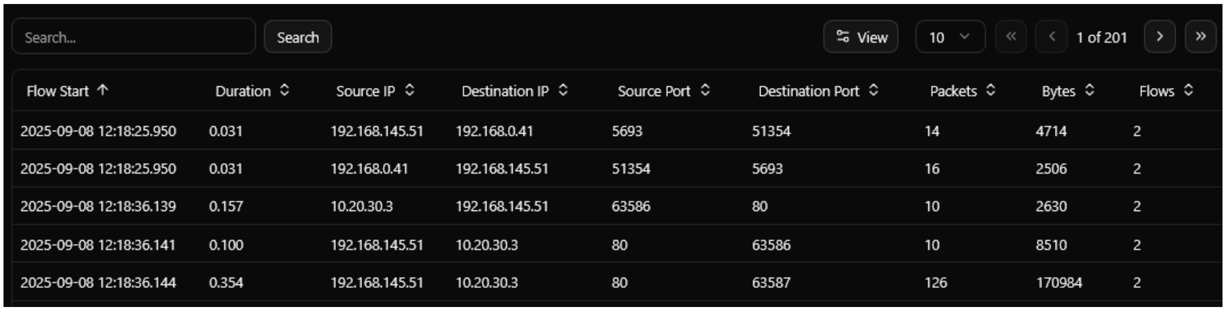
```
(src port 80) and (src ip 192.168.145.56 and dst ip 10.20.30.3 and src port
80 and dst port 63020)
```

In the following sections we'll explore additional options to help you refined your queries and make them more advanced.

IP / Network

A **Raw Query** can use an IP address or a network scope. Here is an example of using an IP address:

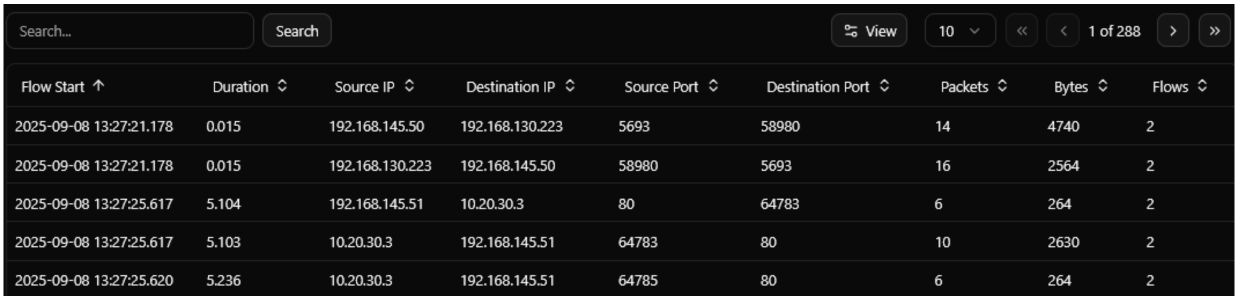
```
ip 192.168.145.51
```



Flow Start	Duration	Source IP	Destination IP	Source Port	Destination Port	Packets	Bytes	Flows
2025-09-08 12:18:25.950	0.031	192.168.145.51	192.168.0.41	5693	51354	14	4714	2
2025-09-08 12:18:25.950	0.031	192.168.0.41	192.168.145.51	51354	5693	16	2506	2
2025-09-08 12:18:36.139	0.157	10.20.30.3	192.168.145.51	63586	80	10	2630	2
2025-09-08 12:18:36.141	0.100	192.168.145.51	10.20.30.3	80	63586	10	8510	2
2025-09-08 12:18:36.144	0.354	192.168.145.51	10.20.30.3	80	63587	126	170984	2

Here is an example of using a network scope by using the slash notation:

```
net 192.168.145.0/24
```



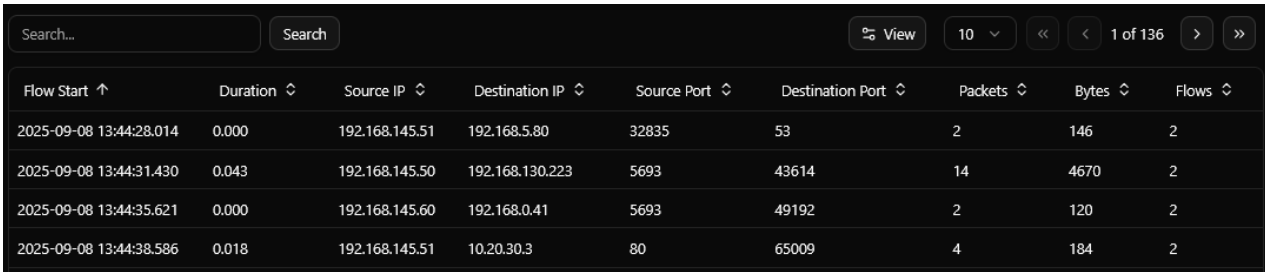
Flow Start	Duration	Source IP	Destination IP	Source Port	Destination Port	Packets	Bytes	Flows
2025-09-08 13:27:21.178	0.015	192.168.145.50	192.168.130.223	5693	58980	14	4740	2
2025-09-08 13:27:21.178	0.015	192.168.130.223	192.168.145.50	58980	5693	16	2564	2
2025-09-08 13:27:25.617	5.104	192.168.145.51	10.20.30.3	80	64783	6	264	2
2025-09-08 13:27:25.617	5.103	10.20.30.3	192.168.145.51	64783	80	10	2630	2
2025-09-08 13:27:25.620	5.236	10.20.30.3	192.168.145.51	64785	80	6	264	2

In the screenshots above, you can see that the IP address or the network scope being queried appears in either the **Source IP** or **Destination IP** columns.

Defining Source Or Destination

Queries can be prepended by using `src` or `dst` to target a specific traffic direction. Here is a net example:

```
src net 192.168.145.0/24
```

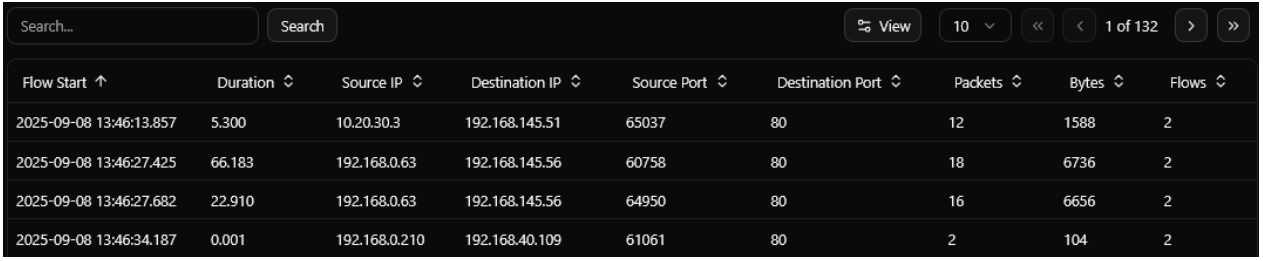


Flow Start	Duration	Source IP	Destination IP	Source Port	Destination Port	Packets	Bytes	Flows
2025-09-08 13:44:28.014	0.000	192.168.145.51	192.168.5.80	32835	53	2	146	2
2025-09-08 13:44:31.430	0.043	192.168.145.50	192.168.130.223	5693	43614	14	4670	2
2025-09-08 13:44:35.621	0.000	192.168.145.60	192.168.0.41	5693	49192	2	120	2
2025-09-08 13:44:38.586	0.018	192.168.145.51	10.20.30.3	80	65009	4	184	2

In the screenshot above, all of the `10.25.0.0/16` addresses are in the **Source IP** column.

Here is a port example:

```
dst port 80
```



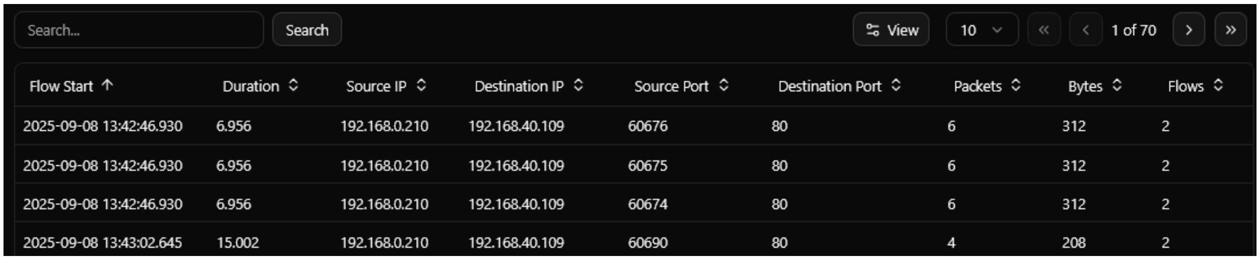
Flow Start	Duration	Source IP	Destination IP	Source Port	Destination Port	Packets	Bytes	Flows
2025-09-08 13:46:13.857	5.300	10.20.30.3	192.168.145.51	65037	80	12	1588	2
2025-09-08 13:46:27.425	66.183	192.168.0.63	192.168.145.56	60758	80	18	6736	2
2025-09-08 13:46:27.682	22.910	192.168.0.63	192.168.145.56	64950	80	16	6656	2
2025-09-08 13:46:34.187	0.001	192.168.0.210	192.168.40.109	61061	80	2	104	2

You can see in the screenshot above that `port 80` is only in the **Destination Port** column.

Logic Operator: AND

Using the `AND` operator can allow you to have more granular queries, for example:

```
src ip 192.168.0.210 and dst port 80
```

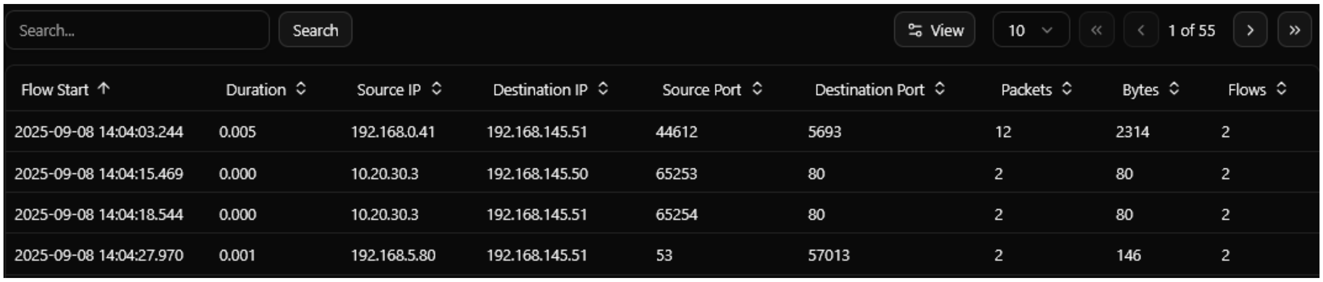


Flow Start	Duration	Source IP	Destination IP	Source Port	Destination Port	Packets	Bytes	Flows
2025-09-08 13:42:46.930	6.956	192.168.0.210	192.168.40.109	60676	80	6	312	2
2025-09-08 13:42:46.930	6.956	192.168.0.210	192.168.40.109	60675	80	6	312	2
2025-09-08 13:42:46.930	6.956	192.168.0.210	192.168.40.109	60674	80	6	312	2
2025-09-08 13:43:02.645	15.002	192.168.0.210	192.168.40.109	60690	80	4	208	2

Logic Operator: OR

Using the `or` operator can allow you to have more flexible queries, for example:

```
src ip 10.20.30.3 OR dst ip 192.168.145.51
```

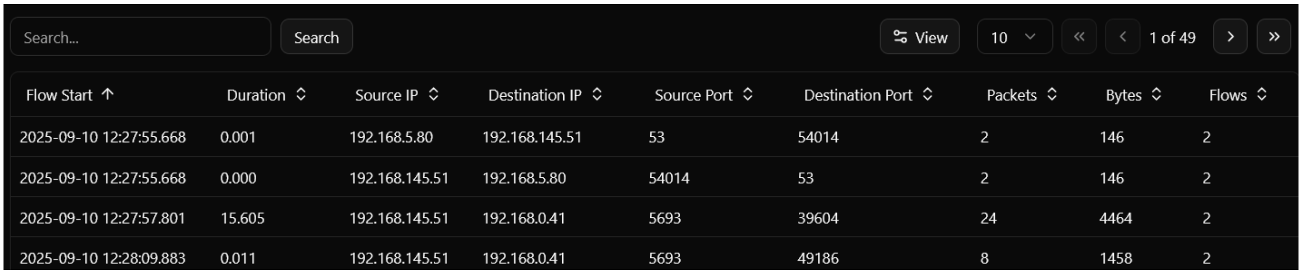


Flow Start	Duration	Source IP	Destination IP	Source Port	Destination Port	Packets	Bytes	Flows
2025-09-08 14:04:03.244	0.005	192.168.0.41	192.168.145.51	44612	5693	12	2314	2
2025-09-08 14:04:15.469	0.000	10.20.30.3	192.168.145.50	65253	80	2	80	2
2025-09-08 14:04:18.544	0.000	10.20.30.3	192.168.145.51	65254	80	2	80	2
2025-09-08 14:04:27.970	0.001	192.168.5.80	192.168.145.51	53	57013	2	146	2

Logic Operator: NOT

Using the `not` operator can allow you to have queries that exclude data, for example:

```
ip 192.168.145.51 and not dst port 5693
```

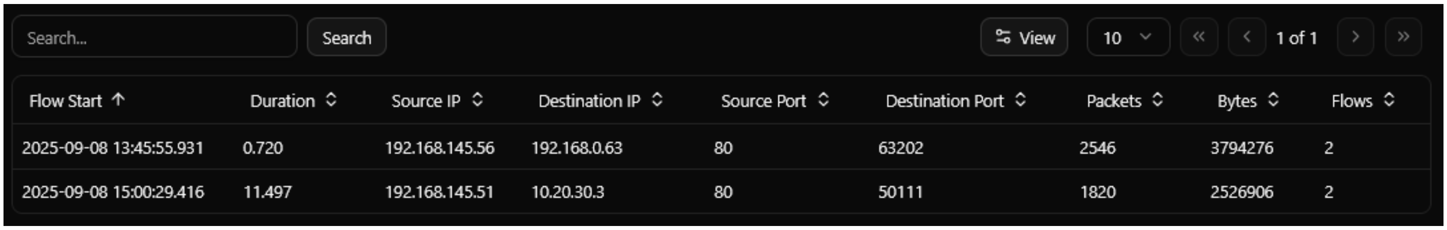


Flow Start	Duration	Source IP	Destination IP	Source Port	Destination Port	Packets	Bytes	Flows
2025-09-10 12:27:55.668	0.001	192.168.5.80	192.168.145.51	53	54014	2	146	2
2025-09-10 12:27:55.668	0.000	192.168.145.51	192.168.5.80	54014	53	2	146	2
2025-09-10 12:27:57.801	15.605	192.168.145.51	192.168.0.41	5693	39604	24	4464	2
2025-09-10 12:28:09.883	0.011	192.168.145.51	192.168.0.41	5693	49186	8	1458	2

Metrics

You can create queries on the amount of traffic that went through for each flow.

```
src port 80 and bytes > 1m
```

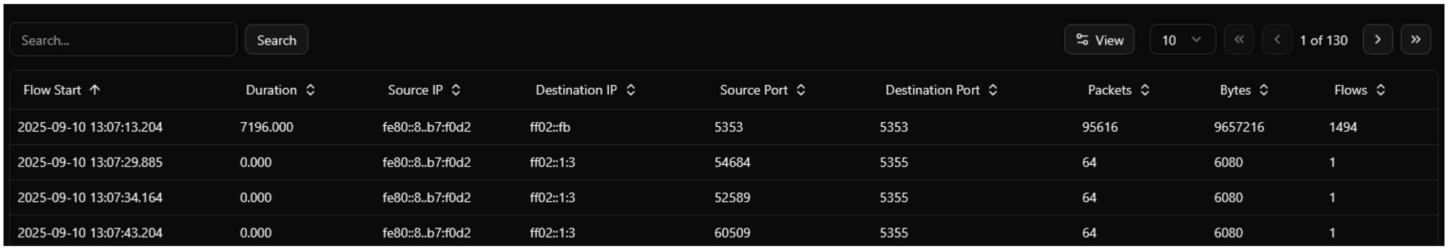


Flow Start	Duration	Source IP	Destination IP	Source Port	Destination Port	Packets	Bytes	Flows
2025-09-08 13:45:55.931	0.720	192.168.145.56	192.168.0.63	80	63202	2546	3794276	2
2025-09-08 15:00:29.416	11.497	192.168.145.51	10.20.30.3	80	50111	1820	2526906	2

IPv6

IPv6 addresses can also be queried for:

```
ip fe80::8c92:e849:a4b7:f0d2
```



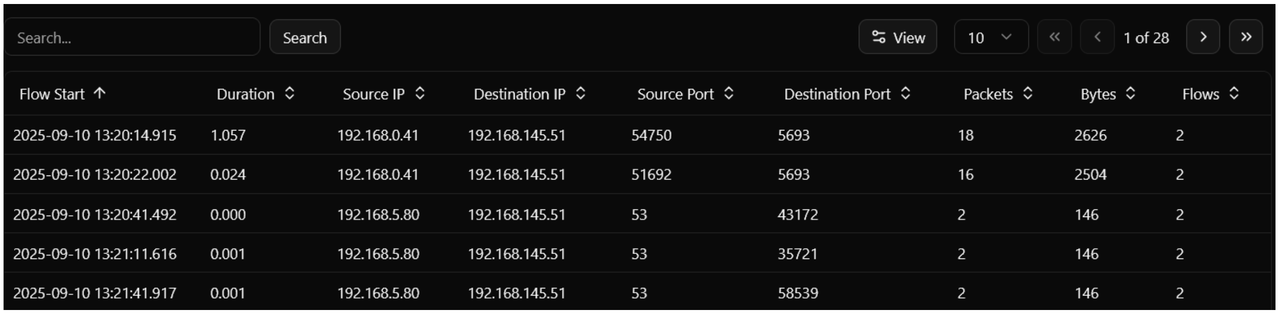
Flow Start ↑	Duration ⇕	Source IP ⇕	Destination IP ⇕	Source Port ⇕	Destination Port ⇕	Packets ⇕	Bytes ⇕	Flows ⇕
2025-09-10 13:07:13.204	7196.000	fe80::8..b7:f0d2	ff02::fb	5353	5353	95616	9657216	1494
2025-09-10 13:07:29.885	0.000	fe80::8..b7:f0d2	ff02::1:3	54684	5355	64	6080	1
2025-09-10 13:07:34.164	0.000	fe80::8..b7:f0d2	ff02::1:3	52589	5355	64	6080	1
2025-09-10 13:07:43.204	0.000	fe80::8..b7:f0d2	ff02::1:3	60509	5355	64	6080	1

Using Parenthesis To Group Expressions

You can add parenthesis to your expression to make it clear how the query will be executed, this allows for more complex queries. Here is a simple example:

```
dst ip 192.168.145.51 and (src port 53 or dst port 5693)
```

You can see that the example provided results for source port 53 OR destination port 5693 on destination IP 192.168.145.51:



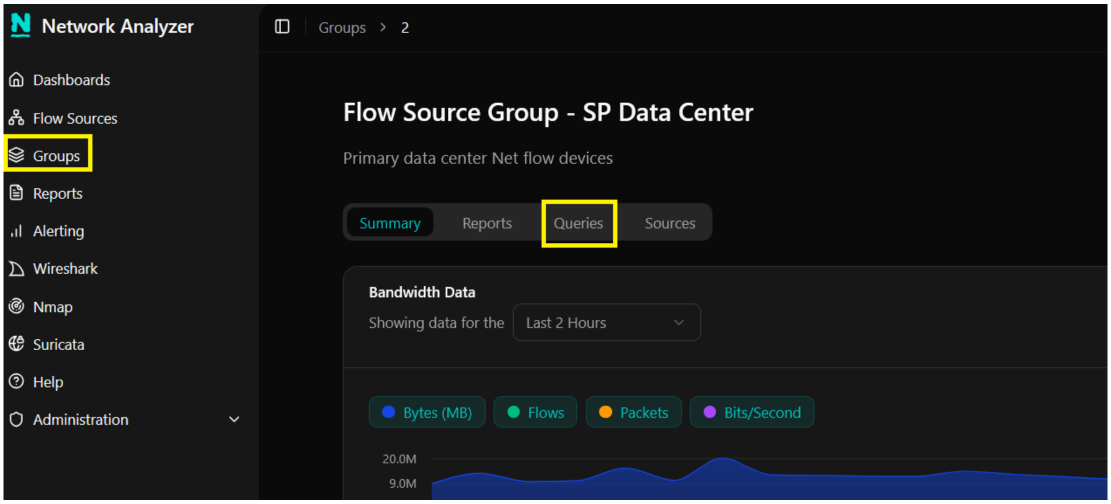
Flow Start ↑	Duration ⇕	Source IP ⇕	Destination IP ⇕	Source Port ⇕	Destination Port ⇕	Packets ⇕	Bytes ⇕	Flows ⇕
2025-09-10 13:20:14.915	1.057	192.168.0.41	192.168.145.51	54750	5693	18	2626	2
2025-09-10 13:20:22.002	0.024	192.168.0.41	192.168.145.51	51692	5693	16	2504	2
2025-09-10 13:20:41.492	0.000	192.168.5.80	192.168.145.51	53	43172	2	146	2
2025-09-10 13:21:11.616	0.001	192.168.5.80	192.168.145.51	53	35721	2	146	2
2025-09-10 13:21:41.917	0.001	192.168.5.80	192.168.145.51	53	58539	2	146	2

Here's a more complicated example that would significantly narrow down the results. The first parenthesis targets two IP addresses or an entire IPv6 subnet (using multiple `ors`). The second parenthesis allows port `80` OR `443`. Then two more conditions were defined:

```
(src ip 10.25.254.50 OR src ip 10.25.14.10 OR src net
2001:44b8:3132:25:0:0:0:0/64) AND (dst port 80 OR dst port 443) AND NOT src
ip 2001:44b8:3132:25:10:25:14:52 AND bytes > 10m
```

Source Groups

Queries can also be performed on Source Groups. In Nagios Network Analyzer, select **Groups** from the navigation bar and click on the Source Group **Name** to bring up the details page of the selected Source. Click the **Queries** tab to bring up the query options. The functionality is the same as Sources.



Further Reading

This documentation covered many of the features available in queries, however, it did not comprehensively cover all abilities of the query syntax. If you would like to read more, please refer to the following link:

https://github.com/phaag/nfdump

Finishing Up

This completes the documentation on how to use custom queries in Nagios Network Analyzer. If you have additional questions or other support-related questions, please visit us at our Nagios Support Forum, Nagios Knowledge Base, or Nagios Library:

Visit Nagios Support Forum

Visit Nagios Knowledge Base

Visit Nagios Library