

Manual Deduplication

In general, if you correctly configured exporters (ingress/egress) and decided to enable automatic deduplication by exporting from all devices in flow continuity then all flows in your Traffic Patterns should be automatically deduplicated. Read more in [Configuring NetFlow Export \(Ingress vs. Egress\)](#) and [Enabling Automatic Deduplication](#).

However, if this is not the case then it is also possible for you to adjust Traffic Pattern configuration in a way to achieve flow deduplication.



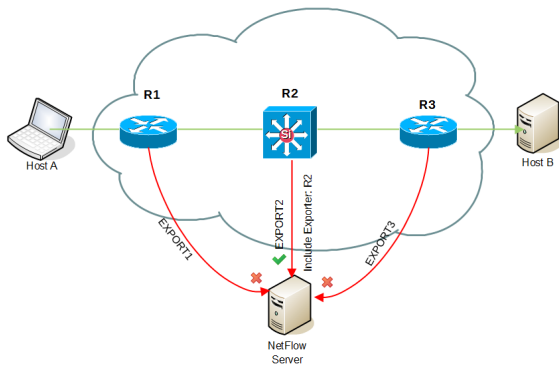
Before proceeding, pay attention to first disable automatic deduplication (at **NetFlow Settings > Configuration**).

On this page:

- [Deduplication based on the central exporter](#)
- [Deduplication based on exporters and their interfaces](#)
- [Deduplication based on next hop](#)

Deduplication based on the central exporter

If you have a central exporter (a netflow exporter through which all desired traffic is passing through) then preventing duplicated Traffic Pattern traffic is easy. You just need to add a filter to the Traffic Pattern in the Exporter section of the Traffic Pattern definition. Add the IP address of the central exporter while include option is set. This will result in Traffic Pattern matching only netflow that was exporter by the central exporter.



In our example above, flow that passes and is exported by three routers (R1, R2 and R3) will be taken into account and processed only from central router (R2) since Traffic Pattern includes its IP address in Exporter filter.

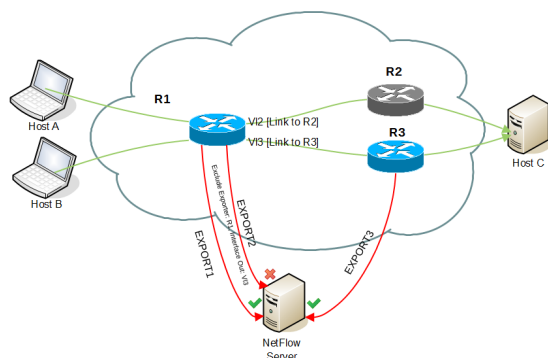


Have in mind that all other traffic (that does not pass via central exporter) will not be captured.

Learn more about [Filtering Based on Exporter and its Interfaces](#).

Deduplication based on exporters and their interfaces

If you do not have a central exporter and/or your network topology is more complex you can prevent duplicated Traffic Patterns by entering exporters and their specific interfaces from which you will either include or exclude traffic when matching traffic to a Traffic Pattern. In this way you can exclude specific interfaces on exporters that would duplicate the traffic.



In the example above, flow travelling via R1 and R2 will not be duplicated since R2 is not an exporter, however flow travelling via R1 and R3 will be duplicated. By excluding Interface Out: VI3 on Exporter R1 only export from exporter R3 will be processed.

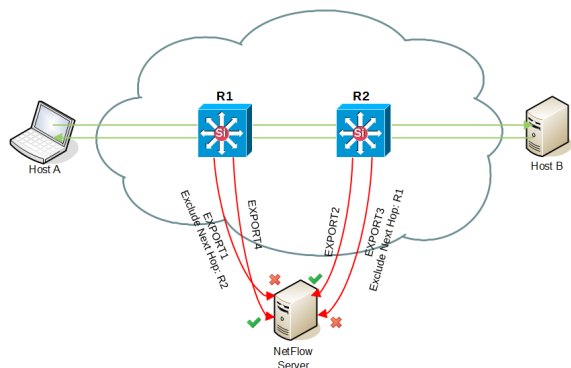


Have in mind that all other traffic (that passes via included exporters and interfaces) will be captured.

Learn more about [Filtering Based on Exporter and its Interfaces](#).

Deduplication based on next hop

In the example below, a flow travelling from Host A to Host B passes via two central routers R1 and R2. As a consequence, one flow is exported and processed to a netflow server twice (by R1 and R2). This should be overcome by adding next hop filter.



The solution is to exclude R2 as Next Hop IP address. This will simply skip all the flows passing from router R1 to R2. Flows will be then matched and processed only by router R2. The same applies for flows from Host B to Host A - excluding R1 as Next Hop IP address will skip all the flows passing from router R2 to R1. These flows will be processed only by R1.



Have in mind that all other traffic (that does not have R2 and R1 as next hop) will be captured.

Learn more about [Filtering Based on Next Hop](#).