



WanGuard 6.1 User Guide

- Console
- Sensors (Packet Sensor, Flow Sensor, SNMP Sensor, Sensor Cluster)
- Filters (Packet Filter, Flow Filter, Filter Cluster)

Copyright & Trademark Notices

This edition applies to version 6.x of the licensed program WanGuard and to all subsequent releases and modifications until otherwise indicated in new editions.

Notices

References in this publication to ANDRISOFT S.R.L. products, programs or services do not imply that ANDRISOFT S.R.L. intends to make these available in all countries in which ANDRISOFT S.R.L. operates. Evaluation and verification of operation in conjunction with other products, except those expressly designated by ANDRISOFT S.R.L., are the user's responsibility. ANDRISOFT S.R.L. may have patents or pending patent applications covering subject matter in this document. Supplying this document does not give you any license to these patents. You can send license inquiries, in writing, to the ANDRISOFT S.R.L. sales department, sales@andrisoft.com.

Copyright Acknowledgment

© 2015, ANDRISOFT S.R.L. All rights reserved.

All rights reserved. This document is copyrighted and all rights are reserved by ANDRISOFT S.R.L. No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying and recording, or by any information storage and retrieval system without permission in writing from ANDRISOFT S.R.L.

The information contained in this document is subject to change without notice. If you find any problems in the documentation, please report them to us in writing. ANDRISOFT S.R.L. will not be responsible for any loss, costs or damages incurred due to the use of this documentation.

WanGuard and WanSight are SOFTWARE PRODUCTS of ANDRISOFT S.R.L. WanGuard and WanSight are trademarks of ANDRISOFT S.R.L. Other company, product or service names may be trademarks or service marks of others.

ANDRISOFT S.R.L.

Website: <https://www.andrisoft.com>

Sales: sales@andrisoft.com

Technical Support: support@andrisoft.com

© 2015, ANDRISOFT S.R.L. All rights reserved.

Table of Contents

1. IP Traffic Monitoring, DDoS Detection and Mitigation with WanGuard.....	6
WanGuard Key Features & Benefits.....	6
WanGuard Software Components.....	7
2. Choosing a Method of Traffic Monitoring and DDoS Detection.....	8
Comparison between Packet Sniffing, Flow Monitoring and SNMP Polling.....	9
3. Choosing a Method of DDoS Mitigation.....	10
DDoS Mitigation with WanGuard Filter.....	10
WanGuard Filter Deployment Scenarios.....	11
4. WanGuard Installation.....	13
System Requirements.....	13
Console Hardware Requirements.....	14
Packet Sensor Hardware Requirements.....	14
Flow Sensor Hardware Requirements.....	15
SNMP Sensor Hardware Requirements.....	15
Sensor Cluster Hardware Requirements.....	15
Packet Filter Hardware Requirements.....	16
Flow Filter Hardware Requirements.....	16
Filter Cluster Hardware Requirements.....	17
Software Installation.....	17
Opening the Console for the First Time.....	17
Licensing Procedure.....	17
Quick Configuration Steps.....	18
5. Basic Concepts of WanGuard Console.....	19
Side Region.....	19
Central Region.....	19
South Region.....	19
Upper Menus.....	19
6. Configuration » General Settings » Graphs & Storage.....	20
Sensor and Applications Graph Troubleshooting.....	22
IP/Subnet and Profiling Graph Troubleshooting.....	22
AS and Country Graph Troubleshooting.....	22
7. Configuration » General Settings » Anomaly Detection.....	23
8. Configuration » General Settings » Custom Decoders.....	24
9. Configuration » General Settings » Mitigation Options.....	25
10. Configuration » Network & Policy » Response.....	27
Conditional & Dynamic Parameter List.....	28
11. Configuration » Network & Policy » IP Zone.....	34
Anomaly Detection Settings & Threshold Templates.....	35
12. Configuration » Servers.....	37
Server Troubleshooting.....	37
13. Configuration » Components » Packet Sensor.....	38
Packet Sensor Optimization Steps for Intel 82599.....	40
Packet Sensor Optimization Steps for Myricom.....	40
Packet Sensor Troubleshooting.....	41

14.Configuration » Components » Flow Sensor.....	42
Flow Sensor Troubleshooting.....	44
15.Configuration » Components » SNMP Sensor.....	46
SNMP Sensor Troubleshooting.....	48
16.Configuration » Components » Sensor Cluster.....	49
17.Configuration » Components » BGP Connection.....	50
BGP Connection Troubleshooting.....	51
18.Configuration » Components » Packet Filter.....	53
Packet Filter Troubleshooting.....	56
19.Configuration » Components » Flow Filter.....	58
20.Configuration » Components » Filter Cluster.....	62
21.Configuration » Schedulers » Scheduled Reports.....	65
22.Configuration » Schedulers » Event Reporting.....	66
23.Configuration » General Settings » Outgoing Email.....	67
24.Configuration » General Settings » User Management.....	68
25.Configuration » General Settings » User Authentication.....	69
26.Reports » Tools.....	71
Reports » Tools » Anomalies.....	71
Active Anomalies.....	71
Anomaly Archive.....	73
Anomaly Overview.....	73
Reports » Tools » BGP Operations.....	74
Active BGP Announcements.....	74
BGP Announcement Archive.....	75
BGP Connection Events.....	75
Reports » Tools » Firewall Rules.....	76
Active Firewall Rules.....	76
Filtering Rules.....	77
Reports » Tools » Flow Collectors.....	78
Flow Records.....	78
Flow Tops.....	78
Reports » Tools » Packet Tracers.....	80
Active Packet Traces.....	80
Packet Trace Archive.....	81
27.Reports » Components.....	82
Reports » Components » Overview.....	82
Console.....	82
Servers.....	83
Sensor Clusters.....	83
Packet Sensors.....	84
Flow Sensors.....	85
SNMP Sensors.....	86
Filter Clusters, Packet Filters and Flow Filters.....	87
Reports » Components » Sensors.....	88
Sensor Dashboard.....	88
Sensor Graphs.....	88
Sensor Tops.....	89
Flow Records.....	90

Flow Tops.....	90
AS Graphs.....	91
Country Graphs.....	91
Sensor Events.....	91
Anomaly Overview.....	91
Reports » Components » Filters.....	92
Filter Dashboard.....	92
Filter Graphs.....	92
Filter Events.....	93
Filtering Rules.....	93
Filter Instances.....	93
28.Reports » Dashboards.....	94
29.Reports » IP Addresses & Groups.....	95
IP Dashboard.....	95
IP Graphs.....	95
IP Accounting.....	96
Flow Records.....	96
Flow Tops.....	97
Profile Graphs.....	97
Anomaly Overview.....	97
30.Reports » Servers.....	98
Console / Server Dashboard.....	98
Console / Server Graphs.....	98
Server Events.....	99
Console Events.....	99
Server Commands.....	99
31.Appendix 1 – IPv4 Subnet CIDR Notation.....	100
32.Appendix 2 – Configuring NetFlow Data Export.....	101
Configuring NDE on an IOS Device.....	101
Configuring NDE on a CatOS Device.....	102
Configuring NDE on a Native IOS Device.....	102
Configuring NDE on a 4000 Series Switch.....	103
Configuring NDE on a Juniper Router (non-MX).....	103
33.Appendix 3 – BGP Black Hole Guideline for WanGuard Sensor.....	105
Understanding of RTBH using WanGuard.....	105
Black-holing on upstream.....	106
IBR BGP Session with WanGuard Sensor – Cisco Router BGP Configuration.....	106
BGP Session with Two ISPs – Cisco Router BGP Configuration.....	106
Interaction with traffic diversion / WanGuard Filter.....	107
34.Appendix 4 – Network Integration Guideline for WanGuard Filter.....	108
Understanding Traffic Diversion Method.....	108
BGP Configuration Guideline.....	109
Quagga / bgpd Configuration.....	110
Cisco Router BGP Configuration.....	112
Understanding Traffic Forwarding Methods.....	114
Layer 2 Forwarding Method.....	114
Layer 3 Forwarding Method.....	115
35.Appendix 5 – Software Changelog.....	122

IP Traffic Monitoring, DDoS Detection and Mitigation with WanGuard

Andrisoft WanGuard is enterprise-grade software that delivers to NOC, IT and Security teams the functionality needed for effectively monitoring and protecting large WAN networks through a single integrated package.

Unforeseen traffic patterns affect user satisfaction and clog costly transit links. Providing reliable network services is imperative for the success of today's organizations. As the business cost of network malfunctions continues to increase, rapid identification and mitigation of threats to network performance and reliability becomes critical in order to meet expected SLAs and network availability requirements. Such threats include distributed denial-of-service attacks (spoofed SYN flood, NTP amplification attacks, generic UDP floods, etc.), propagating worms, misuse of services, and interference of best-effort traffic with critical or real-time traffic. WanGuard's network-wide surveillance of complex, multilayer, switched or routed environments together with its unique combination of features is specifically designed to meet the challenge of pin-pointing and resolving any such threats.

WanGuard Key Features & Benefits

- ✓ **FULL NETWORK VISIBILITY** – Supports all major IP traffic monitoring technologies: packet sniffing, NetFlow version 5,7 and 9; sFlow version 4 and 5; IPFIX and SNMP.
- ✓ **FAST DDOS DETECTION** – Leverages an innovative traffic anomaly detection engine that quickly detects volumetric attacks by profiling the online behavior of users and by comparing over 130 live traffic parameters against user-defined thresholds.
- ✓ **ON-PREMISE DDOS MITIGATION** – Protects networks by using BGP black hole routing; protects services by detecting and cleaning malicious traffic on packet-scrubbing servers deployed in-line or out-of-line.
- ✓ **POWERFUL REACTION TOOLS** – Executes predefined actions that automate the responses to attacks: sends notification emails, announces prefixes in BGP, generates SNMP traps, modifies ACLs, and executes scripts that have access through an easy-to-use API to over 80 internal parameters.
- ✓ **DETAILED FORENSICS** – Samples of packets and flows are captured for forensic investigation during each attack. Detailed attack reports can be emailed to you, affected customers or to the attacker's ISP.
- ✓ **ADVANCED WEB CONSOLE** – Consolidated management and reporting through a single, interactive and highly-configurable HTML5 web portal with customizable dashboards, user roles, remote authentication, etc.
- ✓ **PACKET SNIFFER** – A distributed packet sniffer that saves packet dumps from different network entry points. View packet details in a Wireshark-like web interface.
- ✓ **FLOW COLLECTOR** – A fully featured NetFlow, sFlow, and IPFIX Analyzer and Collector that saves flows in a compressed format for long term storage. Flows can easily be searched, filtered, sorted, and exported.
- ✓ **COMPLEX ANALYTICS** – Generates complex reports with aggregated data for hosts, departments, interfaces, applications, ports, protocols, countries, autonomous systems, and more.
- ✓ **REAL-TIME REPORTING** – Bandwidth graphs are animated and have a short-term accuracy of just 5 seconds.
- ✓ **HISTORICAL REPORTING** – You can view reports from the last 5 seconds to the last 10 years by selecting any

custom time period. Bandwidth histograms contain 95th-percentile values for burstable billing.

- ✓ **SCHEDULED REPORTING** – Any report can be generated automatically and emailed to interested parties at preconfigured intervals of time.
- ✓ **FAST & SCALABLE** – The software was designed to run on commodity hardware. Its components can be distributed on server clusters.
- ✓ **THE LOWEST TCO** – The most affordable on-premise DDoS detection and protection solution on the market.

All statistical data is stored in an internal SQL database that can be easily queried and referenced.

WanGuard Software Components

WanGuard Sensor provides bandwidth monitoring, traffic accounting and traffic anomaly detection. The collected information allows you to generate complex traffic reports, graphs and tops; instantly pin down the cause of network incidents; automate reactions to attacks; understand patterns in application performance and make the right capacity planning decisions.

WanGuard Filter is an optional component used for generating filtering rules that scrub off abnormal traffic in a granular manner without impacting the user experience or resulting in downtime.

WanGuard Console is a multi-tenant web graphical user interface that functions as the administrative core of the software. It offers single-point management and reporting by consolidating data received from the WanGuard Sensors and WanGuard Filters that are deployed within the network.

For brevity, WanGuard Sensor is referred to as Sensor, WanGuard Filter as Filter, and WanGuard Console as Console.

Choosing a Method of Traffic Monitoring and DDoS Detection

This chapter describes the traffic monitoring technologies supported by WanGuard Sensor.

There are four WanGuard Sensor “flavors” that differ only in the way they obtain traffic information:

- **Packet Sensor** analyzes packets. It can be used on appliances that are either deployed in-line (servers, firewalls, routers, bridges) or connected to a mirrored port or TAP.

In switched networks, only the packets for a specific device reach the device's network card. If the server running the Packet Sensor is not deployed in-line, in the main data-path, then a network TAP or a switch / router that offers a “monitoring port” or “mirroring port” must be used. In this case, the network device sends copies of data packets traveling through selected ports or VLANs to the monitoring port. Packet Sensor inspects every packet it receives and conducts packet-based traffic analysis.

- **Flow Sensor** analyzes flows. It is used for monitoring NetFlow® (jFlow, NetStream, cflowd), sFlow® and IPFIX data.

Many routers and switches can collect IP traffic statistics and periodically send them as flow records to a Flow Sensor. Because the flow protocol already performs pre-aggregation of traffic data, the flow data sent to the Flow Sensor is much smaller than the monitored traffic. This makes the Flow Sensor a good option for monitoring remote or high-traffic networks. The main downside of flow-based traffic analysis is that pre-aggregating traffic data adds a delay of at least 30 seconds to collecting real-time traffic statistics.

- **SNMP Sensor** monitors the bandwidth usage of routers and switches on a port-by-port basis.

When this technology is used, an SNMP Sensor queries the device (e.g. router, switch and server) for the traffic counters of each port with quite small data packets. These are triggering reply packets from the device. Compared to other bandwidth monitoring technologies, the SNMP option is very basic and offers no IP-specific information. SNMP creates the least CPU and network load.

- **Sensor Cluster** aggregates pre-existing Sensor traffic data into a single, unified anomaly detection and IP graphing domain.

It sums-up traffic data collected by Packet Sensors, Flow Sensors and SNMP Sensors.

For redundancy, high availability and to be able to view packet traces and flow dumps, use Flow Sensor(s) and Packet Sensor(s) simultaneously.

Comparison between Packet Sniffing, Flow Monitoring and SNMP Polling

Use the Packet Sensor when the speed of detecting attacks is critical or when there is a need for capturing raw packets for forensics. Because it inspects every packet entering the network, it needs to run on servers with powerful CPUs and fast network adapters.

Flow Sensor analyzes pre-aggregated traffic information sent by routers/switches, so it can monitor multiple 10 Gbit or 40 Gbit interfaces, even when it is running on a low-performance server.

Major Flow Sensor disadvantages:

- ✗ it exhibits reduced speed in processing real-time data because all flow exporters aggregate traffic data over time, with delays of more than 30 seconds
- ✗ enabling the flow exporting feature often results in increased CPU usage on the network device
- ✗ it needs to run on servers with lots of RAM

It is recommended to use the SNMP Sensor only for devices unable to export flows or mirror packets, or when comparing flow and SNMP derived statistics in order to ensure the flow data accuracy.

The table below lists the main differences between the Sensor types:

	Packet Sensor	Flow Sensor	SNMP Sensor
Traffic Monitoring Technology	- Sniffing packets passing an in-line appliance - Port mirroring (SPAN, Roving Analysis Port) - Network TAP	- NetFlow version 5, 7 and 9 (jFlow, NetStream, cflowd) - sFlow version 4 and 5 - IPFIX	- SNMP version 1 - SNMP version 2c - SNMP version 3
Maximum Traffic Capacity per Sensor*	10 GigE	multiples of 40 Gbps	multiples of 40 Gbps
DDoS Detection Time**	≤ 5 seconds	≥ flow export time (≥ 30 seconds) + 5 seconds	≥ 5 seconds, no details on attacked destinations
IP Graphing Accuracy	≥ 5 seconds	≥ 20 seconds	N/A
Traffic Validation Options	IP classes, MAC addresses, VLANs, BPF	IP classes, Interfaces, AS Numbers, Ingress/Egress	Interfaces
Packet Tracer	Yes	No	No
Flow Collector	No	Yes	No

* The software is not limited by the number of connections between IPs.

** WanGuard Sensor is able to detect which destination is under attack. The attackers and attack patterns are detected only by WanGuard Filter.

Choosing a Method of DDoS Mitigation

WanGuard provides a network-level protection against volumetric Denial of Service attacks by several complementary methods:

- **WanGuard Sensor** can be configured to announce to upstream provider(s) to stop routing traffic towards the attacked destinations. This is a simple, widely-used DDoS protection technique as it requires only a preexistent agreement with your BGP peer(s). The attacked destinations are effectively blocked from accessing the Internet, leaving upstream links and all other destinations not congested during attacks.
- **WanGuard Sensor** can announce an Internet Service Provider (ISP) or Managed Security Service Provider (MMSP) that offers anti-DDoS services to start scrubbing malicious packets in cloud.
- **WanGuard Filter** can scrub malicious packets on-premise by applying dynamic filtering rules on stateless software firewalls and hardware packet filters. Dedicated filtering servers can be clustered in packet scrubbing farms. This method protects critical services against attacks that do not congest the upstream links.
- **WanGuard Filter** can be configured to send notification emails to the ISP's originating non-spoofed attacks.
- **WanGuard Filter** can be configured to detect filtering rules and to apply them using ACL wrapper scripts on third-party DDoS mitigation appliances, firewalls or routers.

DDoS Mitigation with WanGuard Filter

When a Sensor detects that a destination is under attack, it executes a Response that can activate a Filter instance. Filter instances cannot run stand-alone and can only be started through Responses.

WanGuard Filter includes a sophisticated traffic analysis engine that detects **attack patterns** by inspecting packets and flows sent to the attacked destinations.

Each attack pattern is formed by malicious packets that share some common OSI Layer 3-7 data:

- When an attack is launched from a non-spoofed IP address, the attack pattern is always the IP of the attacker.
- When the attack is spoofed and comes from random IP addresses, the attack pattern can be a common source or destination TCP or UDP port, source or destination IP address, IP protocol number, packet length, packet content, TTL, ICMP type, DNS Transaction ID, originating country, etc.
- When WanGuard Filter detects multiple attack patterns, it generates only the filtering rule(s) that have the least negative impact on normal customer traffic.

Each attack pattern detected by WanGuard Filter is translated into a **filtering rule** that can be applied on the server's stateless firewall, on the network adapter's hardware packet filter, or on third-party appliances. WanGuard Filter is designed to generate filtering rules that block malicious traffic in a granular manner, without impacting the user experience or resulting in downtime.

The stateless operation of WanGuard Sensor and WanGuard Filter ensures detection and mitigation of volumetric attacks that may cripple even the most powerful stateful devices such as firewalls, Intrusion Detection Systems (IDS) or Intrusion Protection Systems (IPS). This is why the WanGuard servers should be installed near the network's entry points, before other stateful devices.

The only disadvantage of the stateless operation is that neither WanGuard Sensor nor WanGuard Filter could detect or block many non-volumetric application-layer (OSI Layer 7) attacks, unlike traditional IPSes.

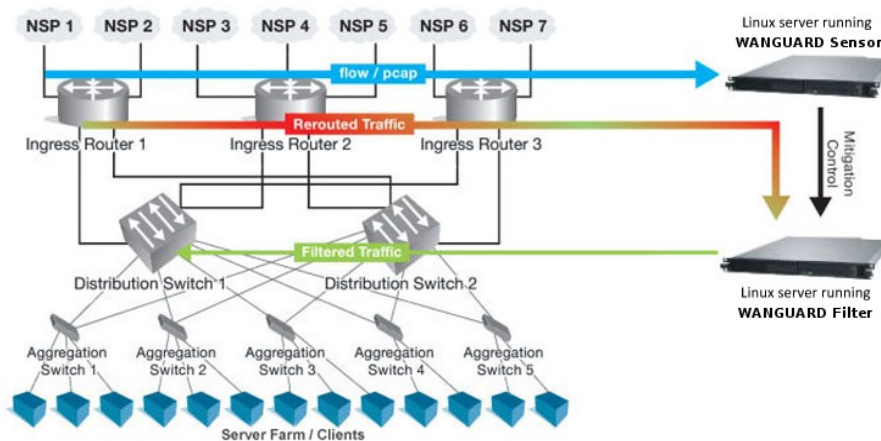
There are three WanGuard Filter “flavors” that differ only in the way they obtain traffic information:

- **Packet Filter** analyzes packets traveling through appliances (servers, firewalls, routers, bridges) deployed in-line, connected to a mirrored port, or that make use of BGP traffic diversion. It needs to run on a powerful server to be able to do packet inspection on high-speed interfaces. Each configuration option is covered on page 53.
- **Flow Filter** analyzes NetFlow® (jFlow, NetStream, cflowd), sFlow® or IPFIX flow data. It can work only in cooperation with a Flow Sensor, so it is not able to generate filtering rules as fast as a Packet Filter. Since flows contain limited traffic information, filtering rules are limited to: IP addresses, IP protocols, TCP and UDP ports, country and protocol. Each configuration option is covered on page 58.
- **Filter Cluster** aggregates traffic data collected by Packet Filter and/or Flow Filter instances. It can be used to create clusters of filtering servers. Each configuration option is covered on page 62.

WanGuard Filter Deployment Scenarios

WanGuard Filter can be deployed on servers configured for:

- **Side-filtering** – The Filter sends a BGP routing update to a border router (route reflector) that sets its server as the next hop for the suspect traffic. The cleaned traffic is routed back into the network using static or dynamic routing.



- **In-line routing** – The Filter runs on a server that resides in the main data path, configured as an OSI

Layer 3 Linux router.

- **In-line network bridging** – The Filter runs on a server that resides in the main data path, configured as an OSI Layer 2 Linux network bridge.
- **Out-of-line monitoring** – The Filter runs on a server that receives a copy of packets from a network TAP or a mirroring port. Direct filtering is not possible, but it is still able to generate filtering rules that improve the visibility of attacks and can be applied on other in-line appliances.
- **Local protection** – The Filter runs as a service on each Linux server that provides critical services. The filtering rules are applied using the local firewall.

WanGuard Installation

Installing WanGuard does not generate negative side effects on the network's performance. Full installation and configuration may take less than an hour; after that, the network will be monitored and protected immediately. No baseline data gathering is required.

WanGuard can be installed exclusively on Linux platforms. To install and configure the software you will need basic Linux operation skills and at least medium computer networking skills. Contact <support@andrisoft.com> if you encounter software installation issues or if you have questions about the system requirements listed below.

System Requirements

WanGuard 6.1 can be installed on the following 64-bit Linux distributions: Red Hat Enterprise Linux 6 or 7 (commercial), CentOS 6 or 7 (free, Red Hat-based), Debian Linux 6 “Squeeze” or 7 “Wheezy” or 8 “Jessie” (free, community-supported), Ubuntu 12 or 14 (free, Debian-based), OpenSuSE 13 (free, Novel-based). The most tested and stable distribution that can be used with WanGuard is CentOS 7.

WanGuard was designed to be completely scalable, so it can be installed either on a single server with adequate hardware resources, or on multiple servers distributed across the network. You can use Virtual Machines to try the software, but the use of dedicated servers for production is mandatory.

A few arguments against using Virtual Machines with WanGuard after the trial period:

- Having fast and uninterrupted access to the hard disk is a critical requirement.
- The resources must be provisioned in a predictable and timely manner.
- Many Virtual Machines do not have a stable clock source.

Importance of HW resources	CPU Speed (> GHz/core)	CPU Cores (> cores)	RAM Size (> GB)	HDD Size (> GB)	HDD/SSD Speed (> Mbytes/s)	Network Adapter (Vendor, Model)
Console	High	High	High	Very High	Very High	Very Low
Packet Sensor	Very High	High	Low	Low	Low	Very High
Flow Sensor	Low	Low	Very High	Medium	High	Very Low
SNMP Sensor	Very Low	Low	Very Low	Very Low	Very Low	Very Low
Sensor Cluster	Medium	Medium	Medium	Very Low	Very Low	Very Low
Packet Filter	Very High	Very High	Medium	Very Low	Very Low	Very High
Flow Filter	Low	Low	High	Very Low	Very Low	Very Low
Filter Cluster	Medium	Medium	High	Very Low	Very Low	Very High

Legend	Very High Importance	High Importance	Medium Importance	Low Importance	Very Low Importance
--------	----------------------	-----------------	-------------------	----------------	---------------------

Console Hardware Requirements

Capacity	Minimum Hardware Requirements for 20 Components
Architecture	64bit x86
CPU	2.4 GHz dual-core Xeon
RAM	4 GB
NICs	1 x Fast Ethernet for management
HDDs	2 x 7200 RPM HDD, RAID 1, 80 GB (additional disk space may be needed for IP graphs)

The Console server stores the database and centralizes all operational logs, graphs and IP accounting data.

Its performance is determined by its settings, as well as the performance of the server and the performance of the applications it relies on: MySQL or MariaDB, Apache HTTPD and PHP.

To access the web interface provided by Console, use one of the following web browsers: Google Chrome, Firefox 3.5 or later, Safari 3.0 or later, Internet Explorer 8 or later. JavaScript and cookies must be enabled. Java and Adobe Flash are not required. The contextual help provided by Console requires Adobe PDF Reader. For the best experience, use a 1280x1024 or higher resolution display.

Packet Sensor Hardware Requirements

Packet Sniffing Capacity	1 Gbit/s – 1,400,000 packets/s	10 Gbit/s – 14,000,000 packets/s
Architecture	64bit x86	64bit x86
CPU	2.0 GHz dual-core Xeon	3.2 GHz quad-core Xeon (e.g. Intel X5672)
RAM	2 GB	4 GB
NICs	1 x Gigabit Ethernet (with driver supported by PF_RING) 1 x Fast Ethernet for management	1 x 10 GbE adapter (Myricom or Intel 82599 chipset) 1 x Fast Ethernet for management
HDDs	2 x 5200 RPM HDD, RAID 1, 35 GB	2 x 5200 RPM HDD, RAID 1, 35 GB

Packet Sensor can be load-balanced on multiple CPU cores only with:

- Intel 82599 chipset network adapters, such as Intel X520, Intel X540, HP X560, Silicom PE310G4DBi9-T.
- Myricom network adapters with a Sniffer 10G license.

On any other network adapter supported by Linux, Packet Sensor runs single-threaded on a single CPU core, which may lead to packet loss when analyzing very high packet rates. Packet Sensor is compatible with the PF_RING high-speed packet capturing engine.

Through Sensor Cluster you can increase the packet sniffing capacity to 40 Gbit/s, 100 Gbit/s or more by using multiple servers that run Packet Sensors on 10 Gbit/s network adapters.

Flow Sensor Hardware Requirements

Capacity	Minimum Hardware Requirements for 5,000 flows/s
Architecture	64bit x86
CPU	2.0 GHz dual-core Xeon
RAM	8 GB
NICs	1 x Fast Ethernet for management
HDDs	2 x 7200 RPM HDD, RAID 1, 60 GB

Flow Sensor does not have a limit on the number of interfaces it can monitor or a limit of how many flows per second it can process. Each Flow Sensor can process the flows of a single flow exporter. A server with enough RAM can run tens of Flow Sensors. For Flow Sensor, the amount of the RAM is much more important than the CPU speed.

Flow Sensor stores all received flows on the local disk in a compressed binary format.

SNMP Sensor Hardware Requirements

Capacity	Minimum Hardware Requirements for 20 Devices
Architecture	64bit x86
CPU	1.6 GHz dual-core Xeon
RAM	1 GB
NICs	1 x Fast Ethernet for management
HDDs	2 x 5200 RPM HDD, RAID 1, 20 GB

SNMP Sensor does not have a limit on the number of interfaces it can monitor. Each SNMP Sensor can monitor a single device. A server can run an unlimited number of SNMP Sensors.

Sensor Cluster Hardware Requirements

The hardware requirements for Sensor Cluster are low because the analyzed traffic information is pre-aggregated by the associated Sensors (Flow Sensors, Packet Sensors or SNMP Sensors).

It is recommended to run Sensor Cluster on the Console server.

Packet Filter Hardware Requirements

Packet Sniffing Capacity	1 Gbit/s – 1,400,000 packets/s	10 Gbit/s – 14,000,000 packets/s
Architecture	64bit x86	64bit x86
CPU	2.4 GHz Xeon	3.2 GHz quad-core Xeon (e.g. Intel X5672)
RAM	2 GB	4 GB
NICs	1 x Gigabit Ethernet (with driver supported by PF_RING) 1 x Fast Ethernet for management	1 x 10 GbE adapter (Chelsio T4/T5, Silicom Intelligent Director or Intel 82599 chipset) 1 x Fast Ethernet for management
HDDs	2 x 5200 RPM HDD, RAID 1, 35 GB	2 x 5200 RPM HDD, RAD 1, 35 GB

The main task of Packet Filter is to inspect the packets sent to the attacked destinations and to generate filtering rules that isolate the malicious traffic. For packet inspection, Packet Filter uses the same capturing engines used by Packet Sensor. To load-balance Packet Filter on multiple CPU cores, use the same configuration required by Packet Sensor.

When a filtering rule is generated, Packet Filter can report it and apply it on the local software-based firewall, in-NIC hardware filter, or third-party filtering appliance.

The software firewall used by Packet Filter does not use the connection tracking mechanism specific to stateful firewalls or IPSes. This ensures a much better filtering and routing performance during spoofed attacks and SYN floods. However, the filtering and packet-forwarding capacity may not be line-rate during powerful attacks with small packets.

The hardware filters supported by Packet Filter offer line-rate packet filtering on:

- Chelsio T4/T5 network adapters. Packet Filter can program 486 LE-TCAM filter rules that block traffic towards source/destination IPv4/IPv6 addresses, source/destination TCP/UDP port, IP protocol.
- Intel 82599 chipset network adapters, such as Intel X520, Intel X540, HP X560. Packet Filter can program 4096 filter rules that block source or destination IPv4 addresses.
- Silicom Intelligent Director adapters.

To increase the packet filtering capacity to 40 Gbit/s, 100 Gbit/s or more, define a Filter Cluster that aggregates multiple Packet Filters running on different servers equipped with 10 Gbit/s network adapters. To split and distribute the traffic evenly, use a hardware load balancer or equal-cost multi-path routing.

Flow Filter Hardware Requirements

The hardware requirements for Flow Filter are very low because the traffic to be analyzed was already pre-aggregated by Flow Sensor. If the Flow Filter is used only for reporting and not for software/hardware packet filtering, run it on the same server that runs the Console.

Flow Filter can apply filtering rules just like Packet Filter. The requirements for software-based and/or hardware-based traffic filtering are listed in the Packet Filter Hardware Requirements section.

Filter Cluster Hardware Requirements

Filter Cluster is used to group and aggregate multiple Packet Filters and/or Flow Filters.

The hardware requirements for Filter Cluster are low as the analyzed traffic information is pre-aggregated by the associated Filters. If the Filter Cluster is used only for reporting and not for software/hardware packet filtering, run it on the same server that runs the Console.

Filter Cluster can apply filtering rules just like Packet Filter. The requirements for software-based and/or hardware-based traffic filtering are listed in the Packet Filter Hardware Requirements section.

Software Installation

The download link is listed in the email containing the trial license key. The latest software installation instructions are listed on the Andrisoft website.

The trial license key activates all WanGuard features for 30 days. You can install the trial license key on any number of servers. To switch to a full, registered version, apply a license key purchased from the online store.

Opening the Console for the First Time

WanGuard Console provides a web interface and centralized system through which you will control and monitor all other components. If you correctly followed the installation instructions, from now on you will only need to log into the Console to manage and monitor servers and software components.

To login into the Console, open `http://<console_hostname>/wanguard`. If the page cannot be displayed, make sure the Apache web server is running and the firewall does not block incoming traffic on port 80. You can also access the Console securely using HTTPS if the Apache web server was configured with SSL/TLS support.

Licensing Procedure

If you have not yet licensed WanGuard you will be asked to do so. Upload the *trial.key* file sent to you by email by clicking the key icon. The license key contains encrypted information about the licensed capabilities of the software. You can switch between WanGuard and WanSight by changing the license key.

Log into the Console using the default username/password combination: **admin/changeme**.

If the Console is installed on a public server, you should immediately change the default password of the “admin” account. To do so, click the **Admin** menu at the top-right corner of the browser window and select **<Change Password>**.

To understand how to navigate within Console, read the chapter beginning on page 19.

Quick Configuration Steps

- ➔ Estimate storage requirements, review decoders and IP graph settings – page 20
- ➔ Setup anomalies detection parameters and decoders – page 23
- ➔ Configure the reaction to traffic anomalies – page 27
- ➔ Add your IP address ranges and important hosts to an IP Zone – page 34
- ➔ Configure anomaly detection for prefixes, create Threshold Templates – page 35
- ➔ Add a Packet Sensor – page 38, Flow Sensor – page 42, or SNMP Sensor – page 46
- ➔ Configure BGP Connections for black hole routing or traffic diversion – page 50
- ➔ For DDoS mitigation add a Packet Filter – page 53, or a Flow Filter – page 58
- ➔ Watch the event log. Receive error notifications by email – page 66
- ➔ Generate reports and send them periodically by email – page 65
- ➔ Create your own dashboards and add widgets with useful information – page 94
- ➔ Create personalized Console accounts for your staff or customers – page 68

Basic Concepts of WanGuard Console

Please read this chapter to understand the basic premises required to properly operate the software. The next chapters cover the configuration of the software, while the last 5 chapters cover the reporting features.

To understand the operation of the Console you should be aware of the structure of the web interface:

Side Region

The Side Region is used for navigation throughout the Console. It is located at the east and/or west edge of the browser's window, according to the user's preference. If it is not visible, it has been either collapsed or hidden by an administrator. Clicking the edge of regions expands or collapses them.

The Side Region contains 2 sections – Reports and Configuration – that can be collapsed or expanded by clicking the title bars or by pressing Ctrl+R. Both sections contain multiple panels that can also collapse or expand, with such state being maintained between sessions. Panels are refreshed every 5 to 10 seconds.

The Reports section title bar contains a “Quick Search” button. Keyboard shortcut: Ctrl+S.

Central Region

Each report, dashboard or tool you select in the Side Region opens a tab (page) in the Central Region. You may switch between (sub-) tabs with a mouse or with the keyboard shortcut (Alt+) Ctrl+→ and (Alt+)Ctrl+←. You can close all tabs except for the Landing Tab (initially set to the Configuration Wizard). To change the Landing Tab, edit your user profile in Configuration » General Settings » User Management.

South Region

The South Region provides a quick way to view live data: events (system logs), animated traffic graphs, traffic anomalies, and statistics from all software components. It is located at the bottom of the browser's window. By default, it is collapsed; to expand it, click the thin line near the bottom edge or press Ctrl+E.

Upper Menus

The Upper Menus are located on the top-right part of the Console window.

The Help menu contains a link to the User Guide, a few helper tools and the About window. Dependent on context, the User Guide will open at the chapter describing the last-opened window or tab. Contextual help works with Adobe PDF Reader.

The User menu provides a Log Out option and lets you quickly change the password and few user preferences.

Configuration » General Settings » Graphs & Storage

A very important initial step in configuring WanGuard is to make sure that the server(s) the software runs on have enough resources to process and store IP graphs, flows and packet dumps. Storage-related settings can be tuned by editing Configuration » General Settings » Graphs & Storage.

In a later chapter, you will be able to configure the Sensors to generate traffic graphs, tops and accounting data for every IP that belongs to the monitored network. If you intend to use this feature, you may want to change the default IP storage settings, as changing these later will reset all existing IP graphs, tops and accounting data.

The **Sensor Top N** value (default: 20) specifies the maximum number of items stored for ordered sets of data, such as top Talkers, External IPs, ASNs, Countries, TCP/UDP ports, IP protocols, etc.

The Packet Sensors save packet dumps on the local disk in the path configured for **Packet Traces**. The Flow Sensors save flow data on the local disk in the path configured for **Flow Collectors**. When the Console is not installed on the Sensor server, export these paths towards the Console's file system using an NFS share ([KB article link](#)). If you do not, the Console will not be able to display data saved on remote servers.

All graph files are stored by the Console server, in the **Graphs Disk Path**. Graph files are optimized for storing time series data and do not grow over time. All IP graph options described below have a direct impact on the storage space required on the Console server.

The **Graph IP Sweeps** option prevents creating IP graph files for IPv4 and/or IPv6 addresses that receive traffic without sending any traffic in return. Do not set to “Off” when monitoring unidirectional links or asymmetric traffic.

The size of each IP graph file is listed on the bottom of the window in the *Disk space required for each IP graph file* field. When Sensor Clusters are not used, the maximum number of IP graph files that could be generated can be calculated with the formula: ((number of Packet Sensors) + (number of Flow Sensor interfaces)) x (number of IPs contained in subnets with IP Graphing set to “Yes” in the IP Zone).

There are 2 mutually exclusive methods for creating and updating IP graph files, so select the appropriate one for your setup:

- **Create & update IP graph files directly on disk** – This method optimizes the long-term storage of IP graph data by allowing up to 3 **Round Robin Archives**. The values within the Round Robin Archives determine the granularity of the graphs and the interval of time they are saved for. These entries specify for how long, and how accurately data should be stored. A smaller data average (5 seconds minimum) will generate a very accurate graph, but will require more disk space, while a bigger data average is less accurate and uses less disk space.

On non-SSD drives, the disk seek time may be too high to update thousands of IP graph files every few minutes. If this is the case, configure the **RRDCache daemon** to increase the I/O performance of the Console server ([KB article link](#)). If the speed of updating IP graph files is not fast enough, consider the method below.

- **Update IP graph files in RAM or SSD** – This method is not optimal for long-term storage because it allows a single Round Robin Archive per IP graph file. The files are created and updated in **Graphs RAM Path**, and moved periodically onto a larger, albeit slower disk. Select this method when the previous method configured with RRDCached is not fast enough to sustain updating thousands of very high-

granularity IP graphs.

Decoders represent internal functions that differentiate and classify the underlying protocols of each packet and flow. Each enabled decoder increases the size of IP graph, top and accounting data, and causes a small performance penalty on Packet Sensor and Packet Filter. It is recommended to enable only the decoders you are interested in.

You can define your own decoders in Configuration » General Settings » Custom Decoders. Default decoders:

Decoder	Description
TOTAL	Always enabled, matches all IP packets & flows.
TCP	Matches TCP traffic.
TCP+SYN	Matches TCP traffic with SYN flag set and ACK unset. Flow Sensor counts one packet per flow.
UDP	Matches UDP traffic.
ICMP	Matches ICMP traffic.
OTHER	Matches IP protocols that differ from TCP, UDP and ICMP.
BAD	Matches TCP or UDP port set to 0, or IP protocol set to 0.
FLOWS	Matches flow records and replaces packets/s with flows/s. Works only with Flow Sensor.
FLOW+SYN	Matches flow records with SYN flag set. Flow Sensor counts all packets per flow.
FRAGMENT	Matches fragmented IP packets. Works only with Packet Sensor.
TCP-NULL	Matches TCP traffic without TCP flags, indicative of reconnaissance sweeps.
TCP+RST	Matches TCP traffic with RST flag set.
TCP+ACK	Matches TCP traffic with SYN flag unset and ACK set.
TCP+SYNACK	Matches TCP traffic with SYN flag set and ACK flag set.
HTTP	Matches TCP traffic on source or destination port 80.
HTTPS	Matches TCP traffic on source or destination port 443.
MAIL	Matches TCP traffic on source or destination ports 25,110,143,465,585,587,993,995.
DNS	Matches UDP traffic on source or destination port 53.
SIP	Matches TCP or UDP traffic on source or destination port 5060.
IPSEC	Matches IP traffic on IP protocol 50 or 51.
WWW	Matches TCP traffic on source or destination ports 80, 443.
SSH	Matches TCP traffic on source or destination port 22.
NTP	Matches UDP traffic on source or destination port 123.
SNMP	Matches UDP traffic on source or destination ports 161, 163.
RDP	Matches TCP or UDP traffic on source or destination port 3389.
YOUTUBE	Matches IP traffic going or coming from Youtube AS 43515, 36561, or from Youtube subnets.
NETFLIX	Matches IP traffic going or coming from Netflix AS 55095, 40027, 2906, or from Netflix subnets.
HULU	Matches IP traffic going or coming from Hulu AS 23286, or from Hulu subnets.
FACEBOOK	Matches IP traffic going or coming from Facebook AS 54115, 32934, or from Facebook subnets.

Consolidation functions build consolidated values for Round Robin Archives. If you are interested in traffic spikes, check MAXIMUM. If you are interested in average values, check AVERAGE. For low traffic values, check MINIMUM.

Automate the deletion of old data and monitor the disk usage of IP graphs in Configuration » General Settings » Data Retention.

Sensor and Applications Graph Troubleshooting

- ✓ Ensure that all Sensors run correctly by verifying the event log and by viewing live statistics from Reports » Components » Overview. The troubleshooting guide for Packet Sensor is located on page 41, for Flow Sensor on page 44 and for SNMP Sensor on page 48.
- ✓ If the Applications graph is empty but other Sensor graphs are not, and the Sensor is running for more than 5 minutes, open Configuration » General Settings » Graphs & Storage, click <Save> and select <Yes> when being asked to delete existing graph files.

IP/Subnet and Profiling Graph Troubleshooting

- ✓ Ensure that all Sensors run correctly by verifying the event log and by viewing live statistics displayed in Reports » Components » Overview. The troubleshooting guide for Packet Sensor is located on page 41, for Flow Sensor on page 44 and for SNMP Sensor on page 48.
- ✓ Generating IP and profiling graph data has the biggest impact on the load of the Console server. Enable each feature (IP graphing, IP accounting, IP profiling) sequentially for each subnet, after making sure the Console server can handle it. The storage requirements for each subnet are listed in the IP Zone, and the current disk usage in Configuration » General Settings » Data Retention.
- ✓ The internal program used for saving IP graph data is /opt/andrisoft/bin/genrrds_ip. If it is overloading the Console server or the event log contains warnings such as “Updating IP graph data takes longer than 5 minutes”, use RRDCacheD, RAM/SSD updating method, faster disk drivers, enable IP graphing for fewer subnets, or deploy a Sensor Cluster configured to aggregate IP graph data.
- ✓ The internal program used for generating IP or subnet graphs is /opt/andrisoft/bin/gengraph_ip. The program is launched by Console users for each requested IP or subnet graph. If the Console server gets too loaded by gengraph_ip, execute “killall gengraph_ip” and configure RRDCacheD. When launched, the program does not stop until the graph is generated. The program can be slow when users request subnet graphs for subnets not specifically defined in the IP Zone. It is not possible to throttle the number of graphs requested by users.

AS and Country Graph Troubleshooting

- ✓ Ensure that all Sensors run correctly by verifying the event log and by viewing live statistics from Reports » Components » Overview. The troubleshooting guide for Packet Sensor is located on page 41, for Flow Sensor on page 44 and for SNMP Sensor on page 48.
- ✓ To enable AS and Country graphs, set the Top Generator parameter to either “Extended” for Flow Sensor, or “Full” for Packet Sensor.
- ✓ SNMP Sensor is not able to generate AS graphs or Country graphs.

Configuration » General Settings » Anomaly Detection

The global settings of the anomaly detection engine can be configured in Configuration » General Settings » Anomalies. Traffic anomaly detection has to be enabled individually for each subnet listed inside IP Zones (details on page 34).

The **Deduplication** feature must be enabled to avoid detection of multiple anomalies for the same attack, when the attack is matched by multiple decoders that are included within each other (without this feature, if you define a 500k pps threshold for the TOTAL decoder, a 400k pps threshold for the TCP decoder and a 30k pps threshold for the TCP+SYN decoder, and a 600k pps TCP+SYN attack is being received, Sensor will detect 3 anomalies, one for each decoder. With this feature on, Sensor will report a single anomaly for the TCP+SYN decoder). Select the first option to disable this feature. Select the second option to enable it. Select the third option to also ignore anomalies for bits/s thresholds when similar anomalies exist for packets/s thresholds.

The **Expiration Time** option lets you select the number of minutes of inactivity before anomalies expire. The default value is 5 minutes.

The **Expiration Type** option can be used to increase linearly or exponentially the number of minutes of inactivity before recurring anomalies expire.

WanGuard Sensor detects traffic anomalies using two different and nonexclusive methods:

- **Threshold Anomalies** – Detected for user-defined threshold values. Thresholds can be defined inside IP Zones for the decoders enabled in the **Threshold Anomaly Decoders** list. Decoders are explained in detail in the previous chapter. Enable only the decoders for which you will define thresholds.

Thresholds can include either absolute values (e.g. IP receives 100k UDP packets/s) or percentage values (e.g. IP receives 30% UDP packets/s). To prevent **Percentage Thresholds** from being triggered for small amounts of traffic, configure minimum packets/s and bits/s values. Percentage values are calculated based on the rates of the monitored interface, for the same decoder. E.g. For an interface that receives 100k UDP packets/s, a 30% UDP packets/s threshold defined for an IP, will trigger when the IP receives over 30k UDP packets/s.

- **Profile Anomalies** – Detected through a behavioral recognition approach. WanGuard Sensor detects any activity that deviates from the “normal” traffic received by the protected subnets. After enabling in IP Zone profile anomaly detection for a subnet/host, the Console builds a behavioral traffic graph for a 24 hour period. You can see the graph in Reports » IP Addresses » [Any Subnet/IP] » Profile Graphs.

Profile anomaly detection is recommended only for hosts and subnets that have a predictable traffic pattern. Larger subnets are usually more predictable. To prevent false positives, adjust the deviation percent and minimum packet and bit rates.

The **Deviation %** represents the maximum allowed deviation from the “normal” traffic before triggering a profile anomaly. The default value of 100 allows traffic up to twice (100% normal + 100% deviation) the “normal” value.

The values from the **Advanced Profiling Parameters** panel should not be modified by users.

Configuration » General Settings » Custom Decoders

Decoders represent internal functions that differentiate and classify the underlying protocols of each packet and flow. The predefined decoders are listed in the “Graphs & Storage” chapter on page 20. If you do not wish to define custom decoders, you may safely skip this chapter.

To manage user-defined decoders go to Configuration » General Settings » Custom Decoders. Each custom decoder contains the following information:

- **Decoder Name** – A short name to help you identify the decoder. The field is mandatory.
- **Graph Color** – The color used in graphs for the decoder. The default color is a random one, which can be changed by entering a different HTML color code or by clicking the drop-down menu.
- **Decoder Description** – An optional short description of the decoder.
- **Packet Filtering Expression** – Enter a BPF filtering expression for packets if you intend to use a Packet Sensor and/or Packet Filter. Click the light bulb icon on the right to open a window that will show you the correct syntax. Examples:
 - To match TCP packets with the SYN flag set, enter *tcp[tcpflags] & tcp-syn!=0*
 - To match UDP packets with the destination port under 1024, enter *udp and dst portrange 1-1023*
- **Flow Filtering Expression** – Enter a filtering expression for flows if you intend to use a Flow Sensor and/or Flow Filter. Click the light bulb icon on the right to open a window that will show you the correct syntax. Examples:
 - To match TCP flows having only the SYN flag set, enter *flags S and not flags AFRPU*
 - To match flows with the MPLS label0 set to 2, enter *mpls label0=2*
- **Included Decoders** – Required if you plan to use the decoder for IP graphing. Select the decoders that include the matched traffic, or TOTAL if not sure.
- **Conflicting Decoders** – Required if you plan to use the decoder for IP graphing. Select the decoders that might match same traffic, but not always. Useful only for stacking multiple decoders inside IP graphs.
- **Most-specific Filter Engine** – If you intend to use a Filter for anomalies detected using the decoder, select the most-specific Filter engine that could analyze the traffic. Otherwise, select *disabled*.
- **Netfilter Expression** – Enter Netfilter/iptables argument(s) that match the same traffic matched by the decoder in order to prevent irrelevant packets from passing the software firewall.

Configuration » General Settings » Mitigation Options

To configure and fine-tune some advanced features of WanGuard Filter, go to Configuration » General Settings » Mitigation Options.

All configuration options listed below are related only to the decoder selected next to the **Mitigation Options for Decoder** field located on the top bar.

- **TCP SYN Proxy** – When enabled, Filter activates the SYN Proxy mechanism immediately after its activation. This mechanism shields servers inside the trusted network from SYN flood attacks using a SYN proxy implementation to verify the WAN clients before forwarding their connection requests to the protected server.

When the filtering server applies a SYN Proxy to a TCP connection, it responds to the initial SYN packet with a manufactured SYN/ACK reply waiting for the ACK in response before forwarding the connection request to the server. Devices attacking with SYN flood packets do not respond to the SYN/ACK reply. The firewall identifies them by their lack of this type of response and blocks their spoofed connection attempts. SYN Proxy forces the firewall to manufacture a SYN/ACK response without knowing how the server will respond to the TCP options normally provided on SYN/ACK packets

- **Invalid TCP Flags** – When enabled, Filter blocks all invalid TCP flags immediately after its activation. The filtering rules are applied by the software firewall, for traffic forwarded to/from the attacked destination.
- **Invalid DNS Packets** – When enabled, Filter blocks all invalid DNS traffic (illegal combination of source port and destination port) immediately after its activation. The filtering rules are applied by the software firewall, for traffic forwarded to/from the attacked destination.
- **Private/Reserved IPs** – When enabled, Filter blocks immediately after its activation all private or reserved IPv4 or IPv6 subnets. The filtering rules are applied by the software firewall, for traffic forwarded to/from the attacked destination.
- **IP Blacklist/Reputation** – When enabled, Filter blocks all blacklisted IPs immediately after its activation. The filtering rules are applied by the software firewall, for traffic forwarded to/from the attacked destination.

The <IP Blacklist Options> button allows you to use predefined or to define your own sources that list IPs with bad reputation. This option should be used only for a relatively small number of blacklisted IPs, as it may affect the firewall performance and the routing/forwarding process.

- **Packet Rate-limiting** – You can use this parameter to limit rate of packets / time unit to a predefined value, or to a percentage of the anomaly threshold if the value ends with the character “%”.
- **Packet Rate-limit Hash** – You can apply the packet rate-limiting globally, to a single object (*Src. IP*, *Src. Port*, *Dst. IP* or *Dst. Port*) or to any combination of objects. If the rate-limiting should be connection-oriented, select all objects. To rate-limit the packet rate of each source IP, select the *Src. IP* object.
- **Byte Rate-limiting** – You can use this parameter to limit rate of bytes / time unit to a predefined value, or to a percentage of the anomaly threshold when the value ends with the character “%”.
- **Byte Rate-limit Hash** – You can apply the byte rate-limiting globally, to a single object (*Src. IP*, *Src. Port*,

Dst. IP or *Dst. Port*) or to any combination of objects. If the rate-limiting should be connection-oriented, select all objects. To rate-limit the byte rate of each source IP, select the *Src. IP* object.

The **Filtering Rules Settings** table lets you view and change the automatic filtering rule policy:

- **Enabled** – Check to allow Filter to detect the filtering rule automatically.
- **Filtering Rule** – Describes the filtering rule.
- **Priority** – By double-clicking the cell, you can change the order in which Filter applies filtering rules. The default settings prioritize filtering rules that match the most specific malicious traffic: source IP, source TCP port, source UDP port. You can disable filtering rules such as destination IP/port to prevent service interruption at the risk of allowing malicious traffic to pass through during randomized attacks.
- **Severity** – By double-clicking the cell, you can change the minimum severity of the filtering rule. A severity of 1 enables the filtering rule only when the matched traffic is above the anomaly threshold. To enable the filtering rule only when it matches traffic twice the anomaly threshold, set to 2, etc.
- **OSI Layer** – Shows the OSI layer where the detection is performed.
- **Compatibility** – Displays whether the filtering rule can be detected and applied by Packet Filter and/or Flow Filter.

Configuration » Network & Policy » Response

Responses provide a unique and powerful way to automate the reaction to traffic anomalies detected by Sensors and to filtering rules detected by Filters. To add a Response, go to Configuration » Network & Policy » <+> » Response. If you do not plan to use this feature, you may skip this chapter.

When invoked by a Sensor or Filter, the Response runs the contained **actions**. These are modules that provide means to execute various commands, send notifications, write logs, etc. There are 2 types of actions:

- **Anomaly Actions** – Executed by Sensor for each traffic anomaly while the anomaly is active (actions inside the “While an anomaly is active” panel) or when it expires (actions inside the “When an anomaly expires” panel).
- **Filtering Rule Actions** – Executed by Filter for each filtering rule while the rule is active (actions inside the “While a filtering rule is active” panel) or when it expires (actions inside the “When a filtering rule expires” panel). Filtering rules expose malicious packets that share common OSI layer 3-7 fields (attacker IPs, TCP/UDP ports, length, protocols, TTL, content, etc.).

To add an action, click the <+> button on the title bar of a panel from the left. To modify, delete or rename an action, select the action name. To enable or disable an action, click the square button next to action name.

Each action panel contains specific fields. The following fields are common:

- **Action Name** – The name or short description of the action.
- **Action Priority** – Selects the order of execution relative to the other actions that are defined in the same panel. Lower numerical values correspond to increased priority.
- **Periodicity** – The actions in the “While...” panels can be executed once for each anomaly or filtering rule, or periodically. The interval of execution is 5 seconds for Packet Sensor and Sensor Cluster, and 5-60 seconds for Flow Sensor, depending on the Graphs Accuracy parameter.
- **Execution** – Actions can be executed either automatically with no user intervention, or manually by an operator or administrator that clicks the appropriate icon from Reports » Tools » Anomalies » Active Anomalies » Actions.
- **Record Action** – The name of the action can be visible inside anomaly reports.
- **Preconditions** – Preconditions are rules that must be passed before the action is executed. Each precondition is formed from a **Conditional Parameter**, a comparison function and a user-defined value. Conditional parameters are dynamic, internal parameters that are updated every 5 seconds by Sensors and Filters. When the list of preconditions is empty, the action is always executed.

Dynamic Parameters are parameters defined within curly brackets (e.g. { and }) that can be included in the body of most actions. Every conditional parameter corresponds to a dynamic parameter.

By using the custom script action together with dynamic parameters, you can create custom reactions to anomalies and filtering rules. Custom scripts are executed by the Sensor that detected the anomaly, on the Sensor

server. Or, by the Filter instance that created the filtering rule, on the Filter server. When using a custom script, make sure it can be accessed by the “andrisoft” account (e.g. by saving it in /tmp or /opt/andrisoft/bin) and it can be executed by a non-root account.

The <List Prefixes> button allows you to see what IP classes are configured to use the selected Response.

Conditional & Dynamic Parameter List

#	CONDITIONAL PARAMETER	TYPE	DYNAMIC PARAMETER	DESCRIPTION
GENERAL PARAMETERS				
1	IP Address	String	{ip}	The IP address or block from your network originating or being the target of the anomaly.
2		String	{ip_dns}	The reverse DNS of the anomaly IP. This is {ip} if the DNS lookup is not returning a DNS PTR record.
3	CIDR	Number	{cidr}	The IP mask of the IP address or IP block.
4	Prefix	String	{prefix}	The IP/CIDR from your network originating or being the target of the anomaly.
5	IP Group	String	{ip_group}	The IP Group of the Prefix.
6	Sensor Name	String	{sensor}	The Sensor that detected the anomaly.
7	Sensor Group	String	{sensor_group}	The Device Group of the Sensor.
8	Sensor IP	String	{sensor_ip}	The IP of the server running the Sensor.
9	Sensor Type [<i>Packet Sensor, Flow Sensor, SNMP Sensor, Sensor Cluster</i>]	String	{sensor_type}	Can be Packet Sensor, Flow Sensor, SNMP Sensor or Sensor Cluster.
10	Sensor ID	Number	{sensor_id}	The unique ID of the Sensor.
11	Flow Exporter IP	String	{router_ip}	The IP of the flow exporter, for anomalies detected by Flow Sensor.
12	IP Zone Name	String	{ipzone}	The IP Zone used by the Sensor.
13	Response Name	String	{response}	The Response activated by the anomaly.
14		String	{response_actions}	The list of actions executed by the Response.
15	Template Name	String	{template}	The Threshold Template defining the triggering rule, if any.
16	Expiration Delay (seconds)	String	{expiration}	The number of seconds between the last time the anomaly is detected and the time the anomaly is expired.

17	Captured Packets	Number	{captured_pkts}	The number of packets captured by the Response.
18	BGP Log Size (bytes)	Number	{bgplog_bytes}	The size of the BGP announcement log. It is non-zero if a BGP routing update was triggered for the anomaly.
19	Unique Dynamic Parameters	String	{exclusive}	Contains dynamic parameter(s) that must be unique in all active anomalies. It is usually used for avoiding duplicating actions across multiple attacks. Example: when set to "{ip} {decoder}", the action is executed only if there is no other active anomaly towards the same IP and decoder.
20	Classification [<i>Unclassified, False Positive, Possible Attack, Trivial Attack, Verified Attack, Crippling Attack</i>]	String	{classification}	Console users can manually classify anomalies in Reports » Tools » Anomalies.
21		String	{software_version}	WanGuard software version.
ANOMALY PARAMETERS				
1	Anomaly Description	String	{anomaly}	A description of the anomaly.
2	Anomaly ID	Number	{anomaly_id}	The unique identification number of the anomaly.
3	Anomaly Comment	String	{comment}	The comment added for the anomaly by Console users.
4	Direction [<i>incoming, outgoing</i>]	String	{direction}	The direction of the rule that triggered the anomaly. Can be "incoming" or "outgoing".
5		String	{direction_to_from}	It is "to" for incoming anomalies and "from" for outgoing anomalies.
6		String	{direction_receives_sends}	It is "receives" for incoming anomalies and "sends" for outgoing anomalies.
7	Domain [<i>IP, subnet</i>]	String	{domain}	Domain is "IP" when CIDR = 32 for IPv4 or 128 for IPv6; "subnet" in all other cases.
8	Anomaly Class [<i>threshold, profile</i>]	String	{class}	Class is "threshold" for threshold-based anomalies and "profile" for profiling-based anomalies.
9	Threshold Type [<i>absolute, percentage</i>]	String	{threshold_type}	Threshold-based anomalies can be defined as "absolute" values or as a "percentage" of the total traffic received by Sensor.
10	Anomaly Decoder (Protocol) [TOTAL,...]	String	{decoder}	The traffic decoder (protocol) for the detected anomaly.

11	Comparison [<i>over</i> , <i>under</i>]	String	{operation}	The value is “over” for thresholds exceeding expectations or “under” for thresholds below expectations.
12		String	{comparison}	The value is “>” for thresholds exceeding expectations or “<” for thresholds below expectations.
13	Unit [<i>pkts/s</i> , <i>bits/s</i>]	String	{unit}	Unit is “pkts/s” for packets per second thresholds or “bits/s” for bits per second thresholds.
14	Threshold Value	Number*	{rule_value}	The value configured as threshold.
15	Computed Threshold	Number*	{computed_threshold}	The value of the threshold, dynamically adjusted for profiling-based and percentage-based anomalies.
16	Peak Value	Number*	{value}	The highest value of the abnormal traffic. Represents pkts/s or bits/s, depending on the anomaly unit.
17	Latest Value	Number*	{latest_value}	The latest value of the abnormal traffic. Represents pkts/s or bits/s, depending on the anomaly unit.
18	Sum Value	Number*	{sum_value}	For pkts/s thresholds represents the number of packets of the abnormal traffic. For bits/s thresholds represents the number of bits of the abnormal traffic.
19	Peak Rule Severity	Number	{severity}	The ratio between the peak abnormal traffic rate and the threshold value.
20	Latest Rule Severity	Number	{latest_severity}	The ratio between the latest abnormal traffic rate and the threshold value.
21	Peak Link Severity	Number	{link_severity}	The ratio between the peak abnormal traffic rate and the interface's traffic rate.
22	Latest Link Severity	Number	{latest_link_severity}	The ratio between the latest abnormal traffic rate and the interface's traffic rate.
23		String	{anomaly_log_10}	The first 10 packets or flows of the abnormal traffic.
24		String	{anomaly_log_50}	The first 50 packets or flows of the abnormal traffic.
25		String	{anomaly_log_100}	The first 100 packets or flows of the abnormal traffic.
26		String	{anomaly_log_500}	The first 500 packets or flows of the abnormal traffic.
27		String	{anomaly_log_1000}	The first 1000 packets or flows of the abnormal traffic.
OVERALL TRAFFIC PARAMETERS				

1	Peak TOTAL Pkts/s	Number*	{total_pps}	The peak packets/s throughput of the IP or subnet, for the TOTAL decoder.
2	Peak TOTAL Bits/s	Number*	{total_bps}	The peak bits/s throughput of the IP or subnet, for the TOTAL decoder.
3	Latest TOTAL Pkts/s	Number*	{latest_total_pps}	The latest packets/s throughput of the IP or subnet, for the TOTAL decoder.
4	Latest TOTAL Bits/s	Number*	{latest_total_bps}	The latest bits/s throughput of the IP or subnet, for the TOTAL decoder.
5	TOTAL Packets	Number*	{sum_total_pkts}	The sum of packets of the IP or subnet, recorded during the anomaly, for the TOTAL decoder.
6	TOTAL Bits	Number*	{sum_total_bits}	The sum of bits of the IP or subnet, recorded during the anomaly, for the TOTAL decoder.
TIME-RELATED PARAMETERS				
1	From (unixtime)	Number	{from_unixtime}	The start time of the anomaly in unixtime format.
2	Until (unixtime)	Number	{until_unixtime}	The expiration time of the anomaly in unixtime format.
3	From (ISO 8601)	String	{from},{from_year},{from_month},{from_day},{from_dow},{from_hour},{from_minute}	The start time of the anomaly in iso8601 format.
4	Until (ISO 8601)	String	{until},{until_year},{until_month},{until_day},{until_dow},{until_hour},{until_minute}	The expiration time of the anomaly in iso8601 format.
5	Duration (seconds)	Number	{duration}	The duration of the anomaly, in seconds.
6		String	{duration_clock}	Text string describing the duration of the anomaly. E.g. <5sec, 4m 3s, 5h 4h 3s
7		String	{duration_clock_full}	Text string describing the duration of the anomaly. E.g. <5 seconds, 5 minutes
8	Internal Ticks	Number	{tick}	Internal tick parameter. Packet Sensor increments the value every 5 seconds while the anomaly is being detected.
FILTER PARAMETERS				
1	Filter Name	String	{filter}	The Filter that detected the filtering rule.
2	Filter ID	Number	{filter_id}	The internal ID of the Filter that detected the filtering rule.
3	Filter Type [<i>Packet Filter, Flow Filter, Filter Cluster</i>]	String	{filter_type}	The type of Filter.
4	Filter Group	String	{filter_group}	The Device Group of the Filter.
5	Number of Filters	Number	{filters}	The number of Filter instances activated for the anomaly.

6	Filters Pkts/s	Number*	{filters_pps}	The latest packets/second throughput recorded by active Filter(s) in the abnormal traffic.
7	Filters Bits/s	Number*	{filters_bps}	The latest bits/second throughput recorded by active Filter(s) in the abnormal traffic.
8	Filters Max Pkts/s	Number*	{filters_max_pps}	The maximum packets/second throughput recorded by active Filter(s) in the abnormal traffic.
9	Filters Max Bits/s	Number*	{filters_max_bps}	The maximum bits/second throughput recorded by active Filter(s) in the abnormal traffic.
10	Filtered Packets	Number*	{filters_filtered_packets}	The number of packets filtered by active Filter(s).
11	Filtered Bits	Number*	{filters_filtered_bits}	The number of bits filtered by active Filter(s).
12	Filters CPU Usage	Number	{filters_max_cpu_usage}	The maximum CPU% used by active Filter(s) instance(s).
FILTERING RULE PARAMETERS				
1	Filtering Rule #	Number	{filtering_rule_id}	A unique ID of the filtering rule.
2	Filtering Rule Type [<i>Source IP, Src Port TCP, Src Port UDP, Packet Length, Country, IP Protocol...</i>]	String	{filtering_rule_type}	What kind of filtering rule. All possible values are listed under Configuration » General Settings » Mitigation Options.
3	Filtering Rule Value	String	{filtering_rule_value}	What should be filtered.
		String	{filtering_rule_ip_dns}	If the filtering rule is for an IP, the dynamic parameter provides the reverse DNS of the IP.
4	Filtering Rule ISP	String	{filtering_rule_ip_isp}	If the filtering rule is for an IP, the dynamic parameter provides corresponding organization / ISP / Autonomous System.
5	Filtering Rule Country	String	{filtering_rule_ip_country}	If the filtering rule is for an IP, it contains the country the IP comes from.
6	Filtering Rule Pkts/s	Number*	{filtering_rule_pps}	The latest pkt/s throughput of the traffic matched by the filtering rule.
7	Filtering Rule Bits/s	Number*	{filtering_rule_bps}	The latest bits/s throughput of the traffic matched by the filtering rule.
8	Filtering Rule Peak Pkts/s	Number*	{filtering_rule_max_pps}	The maximum pkt/s throughput of the traffic matched by the filtering rule.
9	Filtering Rule Peak Bits/s	Number*	{filtering_rule_max_bps}	The maximum bits/s throughput of the traffic matched by the filtering rule.
10	Filtering Rule Unit/s	Number*	{filtering_rule_unit}	It is {filtering_rule_pps} for packets/s thresholds and {filtering_rule_bps} for bits/s thresholds.
11	Filtering Rule Peak Unit/s	Number*	{filtering_rule_max_unit}	It is {filtering_rule_max_pps} for packets/s thresholds and {filtering_rule_max_bps} for bits/s thresholds.

12	Filtering Rule Severity	Number	{filtering_rule_severity}	Represents the ratio between the traffic matched by the filtering rule and the threshold value.
13	Filtering Rule Packets	Number*	{filtering_rule_packets}	The number of packets matched by the filtering rule.
14	Filtering Rule Bits	Number*	{filtering_rule_bits}	The number of bits matched by the filtering rule.
15	Filtering Rule Time Interval (seconds)	Number	{filtering_rule_difftime}	The duration of the filtering rule.
16	Filtering Rule Whitelist	Number	{filtering_rule_whitelisted}	If the filtering rule is whitelisted, the value is 1. Otherwise, it is 0.
17	Filtering Rule Traffic Sample Size (bytes)	Number*	{filtering_rule_log_size}	The size of the packet trace that captures traffic matched by the filtering rule. Needs a Capture Traffic action.
18		String	{attacker_isp}	If the filtering rule is for an IP, this dynamic parameter provides the email address of the attacker's ISP.
19		String	{filtering_rule_log_10}	The first 10 packets of the traffic matched by the filtering rule. Needs a Capture Traffic action.
20		String	{filtering_rule_log_50}	The first 50 packets of the traffic matched by the filtering rule. Needs a Capture Traffic action.
21		String	{filtering_rule_log_100}	The first 100 packets of the traffic matched by the filtering rule. Needs a Capture Traffic action.
22		String	{filtering_rule_log_500}	The first 500 packets of the traffic matched by the filtering rule. Needs a Capture Traffic action.
23		String	{filtering_rule_log_1000}	The first 1000 packets of the traffic matched by the filtering rule. Needs a Capture Traffic action.

* All numbers are integers. Numerical values can be returned in multiples of 1,000 by appending `_kilo` to the Dynamic Parameter. The same goes for 1,000,000 by appending `_mega` and 1,000,000,000 by appending `_giga`. To get the biggest multiplier (k, M, G) for the value, append `_prefix`. To return the decoder before the biggest multiplier (k, M, G) value, append `_decoder_prefix`.

Configuration » Network & Policy » IP Zone

IP Zones are hierarchical, tree-like structures from which Sensor(s) learn the monitored network's boundaries and extract per-subnet settings.

You must add all your IP blocks the IP Zone(s) listed in Configuration » Network & Policy. You can add prefixes (IP blocks/subnets/ranges) using the Console web interface, or from the CLI by executing the command “php /opt/andrisoft/api/cli_api.php” on the Console server.

To define a new IP Zone, go to Configuration » Network & Policy » <+> » IP Zone. You need more than one IP Zone only when you need different per-subnet settings for different Sensors. If this is the case, it may be easier to open an existing IP Zone that already contains your IP address ranges, and duplicate it by pressing the < **Duplicate** > button. A new IP Zone will be created with the same name and the word “(copy)” attached, containing the same prefixes and IP groups as the original.

The IP Zone Configuration window is divided by two vertical sections. The buttons that manage prefixes are located in the upper part of the left-hand section. When a new prefix is added, the tree below automatically updates itself. The right-hand section contains panels with user-provided settings for the selected prefix.

To enter IP addresses or IP blocks, you must use the CIDR notation. To enter individual hosts in IP Zones you must use the /32 CIDR mask for IPv4, or /128 for IPv6. For more information about the CIDR notation, see Appendix 1 from page 100.

Every IP Zone contains at least the 0.0.0.0/0 network. Since the CIDR mask is /0, this “supernet” contains all IP addresses available for IPv4 and IPv6. For an easier configuration, every new prefix that you define will inherit by default the properties of the closest (having the biggest CIDR) IP class that includes it.

The **IP Settings** panel in the right-hand section provides the following options:

- **IP Group** – This editable combo box should contain a short description of the selected prefix, or the name of the customer that uses it. Setting the same IP group for multiple prefixes will allow you to generate aggregated traffic reports.
- **IP Graphing** – Set to “Yes” to allow the Sensor to generate graph data for every IP contained in the selected prefix. The **Graph IP Sweeps** option from Configuration » General Settings » Graphs & Storage can be used to prevent generating graph data for IPs that only receive traffic without sending any traffic in return.
- **IP Accounting** – Set to “Yes” to allow the Sensor to generate daily accounting data for each IP contained in the selected prefix.

The **Storage Requirements** column indicates the disk space needed to store the data generated by a single Packet Sensor or Flow Sensor interface. Enabling IP graphing and IP accounting for very large prefixes (e.g. 0.0.0.0/0) might generate (useless) data that can potentially overload the Console server.

The **Comments** panel allows you to write a comment for the selected prefix. It is not visible elsewhere.

Anomaly Detection Settings & Threshold Templates

Define traffic threshold rules by adding them to the **Thresholds** panel from the IP Zone Configuration window. To ease the addition of identical thresholds for multiple prefixes, add them to a Threshold Template instead, by clicking Configuration » Network & Policy » <+> » Threshold Template.

A threshold rule contains the following metrics:

- **Domain** – Sensors can detect anomalies to/from an IP contained in the subnet, or to / from the subnet as a whole.
- **Direction** – The direction of the traffic can be “receives” for inbound traffic that enters the prefix, or “sends” for outbound traffic that leaves the prefix.
- **Comparison** – Select “over” for volumetric anomalies (e.g. DrDoS, DDoS) or “under” to detect missing traffic towards a monitored subnet.
- **Value** – Enter the threshold value as an absolute number or as a percentage of the total traffic received by Sensor, for the selected decoder. Absolute values can be multiples of 1000 with K (kilo) appended, multiples of 1 million with M (mega) appended, or multiples of 1 billion with G (giga) appended.
- **Decoder** – Select one of the decoders enabled in Configuration » Anomaly Detection (see page 23).
- **Unit** – DDoS attacks usually reach a very high number of packets per second, so select “pkts/s” to detect them. For bandwidth-related anomalies, select “bits/s”.
- **Response** – Select a previously defined Response, or select “None” to have no reaction to anomalies.
- **Parent** – Select “Yes” if the threshold should be inherited by more specific prefixes. You can cancel inherited thresholds by selecting “Unlimited” in the Value field.
- **Inheritance** – Displays the parent prefix if the rule was inherited from a less-specific prefix.

Adding a threshold rule on 0.0.0.0/0 that reads, “Any IP receives over 5% TCP+SYN pkts/s” will catch port scans and all significant SYN attacks towards any IP address belonging to your network. A threshold rule on 0.0.0.0/0 that reads, “Subnet sends under 1 TOTAL bits/s” will execute the Response when the link goes down.

Best practices for setting up traffic thresholds for IPs:

- ✓ TCP+SYN thresholds should be set to low values, around 500-1000 packets/s. TCP uses packets with the SYN flag set only for establishing new TCP connections, and few services (e.g. very high volume websites) can handle more than 1000 new connections every second. SYN packets are frequently used for flooding.
- ✓ TCP bits/s thresholds should be set to your maximum bandwidth level per IP. TCP packets carry on average around 500 bytes of data. Setting a threshold of 15k TCP packets/s should be enough for medium-sized networks.
- ✓ ICMP thresholds should be set to very low levels, 50-100 packets/s. ICMP is frequently used for flooding.
- ✓ UDP traffic has high packets/s but low bits/s, so you can set low thresholds for bits/s. Setting UDP packets/s thresholds of around 10k/s per destination should not generate false positives while catching all significant UDP floods. UDP is also used frequently for flooding.

- ✓ The OTHER decoder matches all non-TCP, non-UDP and non-ICMP traffic. You can set thresholds for OTHER if you have non-standard applications in your network. More than 90% of Internet traffic is either TCP or UDP.
- ✓ Enable additional decoders, such as HTTP, MAIL, NTP, etc., to be able to set thresholds for specific services and servers.
- ✓ Configure illegal IP address ranges that should never be seen in normal traffic (unallocated IP addresses or parts of your internal IP address range that is unoccupied). Then, add small thresholds to these, to catch malicious activities such as scans and worms.

Adding similar threshold rules for the same prefix is not allowed, even if the rules have different values or Responses. To execute different actions for different threshold values, define only the smallest threshold value, and then use preconditions inside the Response. For example, if you want to activate Filter for UDP attacks stronger than 100 Mbps but also to null-route by BGP when those attacks reach 1 Gbps, add only the “Any IP receives over 100M UDP bits/s” rule. Then, inside the Response add 2 actions: one that activates Filter with no Precondition, and another that executes a BGP announcement with the Precondition “Peak Value” “over” “1G”.

The **Profile Anomalies** panel contains the Profiling Data parameter, which enables or disables the detection of traffic anomalies by profiling traffic behavior:

- *Inherit* – The value is inherited from the parent prefix.
- *No* – Do not generate profiling data for the selected prefix.
- *For Subnet* – Generate profiling data for all traffic received by the prefix as a whole.
- *For IPs* – Use carefully as it will generate profiling data for every IP contained in the prefix. Enabling this option is not recommended for large subnets because it can overwhelm the I/O of server, and potentially generate false positives since the traffic of single IPs is not always predictable.
- *For All* – Activate both *For Subnet* and *For IPs*.

Configuration » Servers

Any server running Sensor(s) or Filter(s) must be listed under Configuration » Servers. The Console server is automatically added on installation.

To add a new server, click the <+> button from the title bar of the Configuration » Servers panel. To configure an existing server, go to Configuration » Servers and click its name.

The Server Configuration window contains the following fields:

- **Server Name** – A short name to help you identify the server.
- **Graph Color** – The color used in graphs for this server. The default color is a random one, which can be changed by entering a different HTML color code or by clicking the drop-down menu.
- **Reports Visibility** – Whether the Reports » Servers panel should contain icons of the components the server runs.
- **Device Group** – Optional description used within Console to group servers by location, role, etc.
- **Server ID** – Read-only unique identifier of the server, used when exporting NFS shares.
- **IP Address** – The IP address of the server as defined during the WANsupervisor service installation. Can be public or private.
- **Linux Distro** – The Linux distribution installed on the server.
- **Hardware Key** – Read-only string used for licensing purposes. The hardware key field is updated by the WANsupervisor service on installation or when the hardware, IP or hostname changes.
- **Monitored Network Interfaces (optional)** – The WANsupervisor service can monitor packets/s, bits/s, errors and dropped frames for each interface that exists on the server. The data is available in Reports » Servers » [Server] » Server Graphs » Data Units = Server Interfaces. These stats are provided by the OS.
- **Comments** – Comments about the server can be saved here. These comments are not visible elsewhere.

Server Troubleshooting

- ✓ In order for the server to be operational, make sure it always runs the WANsupervisor service and that its clock is synchronized with NTP. You can verify the operational status of each server in Reports » Components » Overview » Servers.
- ✓ The WANsupervisor service breaks when the MySQL service running on the Console server is restarted or unavailable even for a short amount of time (e.g. networking issue). In this case, either restart WANsupervisor manually, or use automated tools such as systemd, monitd or similar.
- ✓ You can discover performance-related issues by monitoring Reports » Server » [Server] » Server Graphs and Reports » Server » [Server] » Server Events.

Configuration » Components » Packet Sensor

In switched networks, only the packets for a specific device reach the device's network card. If the server running the **Packet Sensor** is not deployed in-line (in the main data-path), a network TAP, or a switch/router that offers a “monitoring port” or “mirroring port” must be used. In this case, the network device sends copies of data packets traveling through selected ports or VLANs to the monitoring port. Packet Sensor inspects every packet it receives and conducts packet-based traffic analysis. The advantages and disadvantages of packet-based traffic monitoring are listed on page 8.

For instructions on how to configure switches or routers for port mirroring, consult their documentation.

To add a Packet Sensor, click the <+> button from the title bar of the Configuration » Components panel. To modify an existing Packet Sensor, go to Configuration » Components and click its name.

The Packet Sensor Configuration window contains the following fields:

- **Sensor Name** – A short name to help you identify the Packet Sensor.
- **Graph Color** – The color used in graphs for the Packet Sensor. The default color is a random one, which can be changed by entering a different HTML color code or by clicking the drop-down menu.
- **Reports Visibility** – Whether the Packet Sensor should be listed inside the Reports » Components panel.
- **Device Group** – Optional description used within Console to group components by location, role, etc.
- **Sensor Server** – The server that runs the Packet Sensor. The configuration of servers is described on page 37.
- **Sniffing Interface** – The network interface listened by the Packet Sensor. If the server that runs the Packet Sensor is deployed in-line, then this field must contain the network interface that receives the traffic entering your network.
- **Capture Engine** – Select the best packet capturing engine for your setup:
 - *Embedded LibPcap* – Select to use the built-in LibPcap 1.6.2 library.
 - *System LibPcap* – Select to use the LibPcap library installed by the Linux distribution.
 - *Myricom Sniffer10G* – Select when using a Myricom network adapter with a Sniffer 10G license. Click the button on the right for driver-specific settings.
 - *PF_RING, RX+TX* – Select to use the PF_RING 6.2 library for RX and TX traffic.
 - *PF_RING, RX* – Select to use the PF_RING 6.2 library for RX traffic.
 - *PF_RING, TX* – Select to use the PF_RING 6.2 library for TX traffic.
- **CPU Affinity** – You can force the Packet Sensor to run exclusively on a given set of CPU cores.
- **Link Speed IN / OUT** – Enter the speed (bandwidth, capacity) of the monitored link. The values are used for percentage-based reports and percentage-based bits/s thresholds.
- **Sensor License** – The license used by the Packet Sensor. WanGuard provides all features; WanSight does not provide traffic anomaly detection and reaction.

- **Top Generator** – Allows generation of traffic tops:
 - *Basic* – Enables tops for Internal IPs, IP protocols, versions and TCP/UDP ports. It is the recommended value because it adds a very small performance penalty.
 - *Extended* – Enables all tops from *Basic* as well as tops for External IPs (IPs not included in the IP Zone). It adds performance penalty of over 20%, especially during spoofed attacks.
 - *Full* – Enables all tops from *Extended* as well as tops and graphs for autonomous systems. It adds a performance penalty of over 20%, especially during spoofed attacks.
- **IP Zone** – Packet Sensor needs an IP Zone from which to learn about your network's boundaries and to extract per-subnet settings. IP Zones are described in the “IP Zone” chapter on page 34.
- **IP Validation** – This option can be used to distinguish the direction of the packets or to ignore certain IPs:
 - *Off* – Packet Sensor analyzes all traffic and uses MAC Validation to distinguish the direction of traffic.
 - *On* – Packet Sensor analyzes the traffic that has the source and/or the destination IP in the selected IP Zone.
 - *Strict* – Packet Sensor analyzes the traffic that has either the source or the destination IP in the selected IP Zone.
 - *Exclusive* – Packet Sensor analyzes the traffic that has the destination IP in the selected IP zone, but not the source IP.
- **MAC Validation/Address** – This option can be used to distinguish the direction of the packets or to ignore unwanted OSI Layer 2 traffic:
 - *None* – Packet Sensor analyzes all traffic and uses IP Validation to distinguish the direction of traffic.
 - *Upstream MAC* – MAC validation is active and the MAC Address belongs to the upstream router.
 - *Downstream MAC* – MAC validation is active and the MAC Address belongs to the downstream router.

The MAC Address must be written using the Linux convention – six groups of two hexadecimal values separated by colons (:).
- **BPF Expression** – You can filter the type of traffic the Packet Sensor receives using a tcpdump-style syntax.
- **Comments** – Comments about the Packet Sensor can be saved here. They are not visible elsewhere.

To start the Packet Sensor, click the gray square button next to its name in Configuration » Components. Ensure that the Packet Sensor starts properly by watching the event log (details on page 66).

If the Packet Sensor starts without errors, but you can't see any data collected by it in Reports » Components » Overview, follow the troubleshooting guide from page 41.

Packet Sensor Optimization Steps for Intel 82599

To distribute the packet-processing tasks of the Packet Sensor over multiple CPU cores when using an adapter with the Intel 82599 chipset (Intel X520, Intel X540, HP X560, etc.):

- ✓ Follow the documentation and optimization guides provided by the network adapter vendor.
- ✓ Install PF_RING 6.2 and switch to the ZC or PF_RING-aware ixgbe driver.
- ✓ See the number of RSS queues allocated by the ixgbe driver by executing `dmesg`, or by listing `/var/log/messages` or `/var/log/syslog`. By default, the number of RSS queues is equal to the number of CPU cores when hyper-threading is off, or double the number of CPU cores when hyper-threading is on. You can set the number of RSS queues manually, by loading `ixgbe.ko` with the `RSS=<number>` option.
- ✓ Define multiple Packet Sensors, each listening to `ethX@queue_id` or `ethX@queue_range`. All Packet Sensors defined to listen to a single interface use a single Sensor license.
- ✓ Add the Packet Sensors to a Sensor Cluster to have a unified reporting and anomaly detection domain.

On a quad-core CPU with multithreading, the ixgbe driver will allocate 8 RSS queues. In this case, if you define a Packet Sensor for `ethX@0-3` and another one for `ethX@4-7`, the packet-processing task will be distributed over 2 CPU cores. PF_RING exposes up to 32 RSS queues.

Packet Sensor Optimization Steps for Myricom

To distribute the packet-processing tasks of the Packet Sensor over multiple CPU cores with a Myricom adapter:

- ✓ Follow the documentation provided by Myricom in order to install Sniffer10G v2 or v3 (recommended).
- ✓ Start the driver with `/opt/snf/sbin/myri_start_stop start`
- ✓ Check that the driver is loaded successfully with `lsmod | grep myri_snf`. Check for errors in `syslog`.
- ✓ Define multiple Packet Sensors, one for each CPU core if needed.
- ✓ For each Packet Sensor, set the Capture Engine parameter to “Myricom Sniffer10G”, and click the <Capture Engine Options> button on the right. Set the **Packet Sensor Rings** parameter to the number of Packet Sensors listening to the interface. Sniffer10G v3 users must set two unique **App IDs** for Packet Sensors and Packet Tracers listening to the same interface to ensure that the traffic is directed to both applications.
- ✓ Stop all Packet Sensors before changing the **Capture Engine** parameter.
- ✓ Add the Packet Sensors to a Sensor Cluster to have a unified reporting and anomaly detection domain.

Packet Sensor Troubleshooting

- ✓ Look for warnings or errors produced by the Packet Sensor in the event log (details on page 66).
- ✓ Ensure that you have correctly configured the Packet Sensor. Each configuration field is described in detail in this chapter.
- ✓ Make sure that the sniffing interface is up:

```
ip link show <interface_usually_eth1_or_p1p2>
```
- ✓ Ensure that you have correctly configured the switch/TAP to send packets to the server on the configured interface.
- ✓ Verify whether the server is really receiving packets through the configured interface:

```
tcpdump -i <interface_usually_eth1_or_p1p2> -n -c 100
```
- ✓ When **IP Validation** is not disabled, make sure that the selected IP Zone contains all your subnets.
- ✓ If the CPU usage of the Packet Sensor is too high, set the **Top Generator** parameter to “Basic”, install PF_RING (no ZC/DNA/LibZero required!), or use a network adapter that allows distributing Packet Sensors over multiple CPU cores.
- ✓ To troubleshoot Sensor graph or IP graph issues, follow the Graphs Troubleshooting guide from page 22.
- ✓ For PF_RING installation issues, contact ntop.org. To increase the maximum number of PF_RING programs from 64 to 256, increase the MAX_NUM_RING_SOCKETS defined in kernel/linux/pf_ring.h and recompile the pf_ring kernel module.
- ✓ Make sure you are running the latest version of the software.

Configuration » Components » Flow Sensor

Many routers and switches can collect IP traffic statistics and periodically export them as flow records to a **Flow Sensor**. Because the flow protocol already performs pre-aggregation of traffic data, the flow data sent to Flow Sensor is much smaller than the monitored traffic. This makes the Flow Sensor a good option for monitoring remote or high-traffic networks. The advantages and disadvantages of flow-based monitoring are listed on page 8.

For detailed instructions on how to enable NetFlow, sFlow or IPFIX on your network device, consult its documentation. Appendix 2 from page 101 contains some examples on how to configure NetFlow on few Cisco IOS, CatOS and Juniper devices.

To add a Flow Sensor, click the <+> button from the title bar of the Configuration » Components panel. To modify an existing Flow Sensor, go to Configuration » Components and click its name.

The Flow Sensor Configuration window contains the following fields:

- **Sensor Name** – A short name to help you identify the Flow Sensor.
- **Device Group** – Optional description used within Console to group components by location, role, etc.
- **Reports Visibility** – Whether the Flow Sensor should be listed inside the Reports » Components panel.
- **Sensor Server** – The server that runs the Flow Sensor. The configuration of servers is described on page 37.
- **Listener IP:Port** – The IP address of the network interface that receives flows and the destination port.
- **Repeater IP:Port** – An embedded packet repeater can send all incoming flows to another host or collector. Optional.
- **Flow Collector** – When enabled, all flow data is stored in a space-efficient binary format. Flow records can be queried in Reports » Tools » Flow Collectors.
- **Sensor License** – The license used by the Flow Sensor. WanGuard provides all features; WanSight does not provide traffic anomaly detection and reaction.
- **Flow Protocol** – The flow protocol used by the flow exporter: NetFlow, IPFIX or sFlow.
- **Flow Exporter IP** – The IP address of the flow exporter (router, switch, probe). Usually it is the loopback address of the router. For sFlow exporters, enter the IP that sends flow packets, not the Agent IP.
- **Sampling (1/N)** – Must contain the sampling rate configured on the flow exporter, or 1 when no sampling rate is configured. For NetFlow v9 and sFlow the value entered here is ignored because the sampling rate is automatically adjusted by the flow protocol. To force a particular sampling value, enter a negative value.
- **Flow Timeout (s)** – For flow exporters that maintain the start time of flows, such as Juniper MX, set the same flow-active/inactive-timeout value as the one defined in the flow exporter's configuration. The value must be entered in seconds (s).
- **Time Settings** – The time offset between the time zone (TZ) of the Flow Sensor server and the flow exporter. Running NTP on both devices to keep their clocks synchronized is a critical requirement for

Flow Sensor.

- **IP Zone** – Flow Sensor needs an IP Zone from which to learn the monitored network's boundaries and to extract per-subnet settings. For more information about IP Zones consult the “IP Zone” chapter on page 34.
- **Graphs Accuracy** – Low values increase the accuracy of Sensor graphs, at the expense of increasing the RAM usage. Setting this to under 20 seconds is not recommended.
- **IP Validation** – This option can be used to distinguish the direction of traffic or to ignore certain flows:
 - *Off* – Flow Sensor analyzes all flows and the traffic direction is established by interface.
 - *On* – Flow Sensor analyzes the flows that have the source and/or the destination IP in the selected IP Zone.
 - *Strict* – Flow Sensor analyzes only the flows that have either the source or the destination IP in the IP Zone.
 - *Exclusive* – Flow Sensor analyzes only the flows that have the destination IP in the IP zone, but not the source IP.
- **AS Validation** – Flows from BGP-enabled routers usually contain the source and destination AS (Autonomous System) number. In most configurations, if the AS number is set to 0, then the IP address belongs to your AS.

If enabled, only flows having the AS number set to “0” (your AS) are processed. This is rarely-used option used for establishing traffic direction. AS validation has three options:

- *Off* – Disables AS validation.
- *On* – Only flows that have the source ASN and/or the destination ASN set to 0 are analyzed.
- *Strict* – Only flows that have either the source ASN or the destination ASN set to 0 are analyzed.
- **SNMP Settings** – Configure the SNMP settings of the flow exporter to allow Console to extract interface information. When the SNMP settings are not configured, you must enter the SNMP index, speed, etc. manually for each interface.
- **Monitored Network Interfaces** – The list of interfaces that should be monitored. To avoid producing duplicate flow entries, add only upstream interfaces. Settings per interface:
 - *SNMP Index* – The interfaces are identifiable in flows only by their SNMP indexes. Enter the index manually, or configure the SNMP settings.
 - *Interface Name* – A short description used to identify the monitored interface. Descriptions longer than 10 characters clutter some reports.
 - *Graph Color* – The color used in graphs for the interface. The default color is a random one, which can be changed by entering a different HTML color code or by clicking the drop-down menu.
 - *Traffic Direction* – The direction of traffic entering the interface, relative to your network:
 - “Auto” – Set to establish the direction of traffic by IP and/or AS Validation.
 - “Upstream” – Set for upstream interfaces, e.g. peering interfaces, interfaces connected to the Internet.
 - “Downstream” – Set for downstream interfaces, e.g. customer interfaces, interfaces connected to your network.

- “Null” – Traffic to Null interfaces is discarded by the router and should be ignored.
- *Top Generator* – Allows generating traffic tops:
 - “Basic” – Enables tops for Internal IPs, IP protocols, versions and TCP/UDP ports. It adds a very small performance penalty.
 - “Extended” (recommended) – Enables all tops from “Basic” as well as tops and graphs for autonomous systems and countries, but increases the CPU usage by a few percentage points. When the router does not export AS information (e.g. non-BGP router) Flow Sensor uses an internal GeoIP database to get ASNs. Live stats for autonomous systems and countries are not very accurate.
 - “Full” – Enables all tops from “Extended” as well as tops for external IPs (IPs not included in the IP Zone), but increases the RAM usage several times over, especially during spoofed attacks. Live stats for autonomous systems and countries are very accurate. Set the value to “Extended”, unless you know what you are doing.
- *Link Speed In & Link Speed Out* – Enter the speed (bandwidth, capacity) of the interface. The values are used for percentage-based reports and percentage-based bits/s thresholds.
- **Comments** – Comments about the Flow Sensor can be saved here. These comments are not visible elsewhere.

To start the Flow Sensor, click the gray square button next to its name in Configuration » Components. Ensure that the Flow Sensor starts properly by watching the event log (details on page 66).

If the Flow Sensor starts without errors, but you can't see any data collected by it in Reports » Components » Overview after more than 5 minutes, follow the troubleshooting guide below.

Flow Sensor Troubleshooting

- ✓ Look for warnings or errors produced by the Flow Sensor in the event log (details on page 66).
- ✓ Check if you have correctly configured the Flow Sensor. Each configuration field is described in detail in the previous section.
- ✓ Ensure that the server is receiving flow packets on the configured **Listener IP:Port**:


```
tcpdump -i <interface_eth0_or_plp1_etc> -n -c 100 host <flow_exporter_ip> and udp and port <destination_port>
```
- ✓ Make sure that the local firewall permits the Flow Sensor to receive flow packets:


```
iptables -L -n -v && iptables -t raw -L -n -v
```
- ✓ Ensure that the clocks of both devices are synchronized with NTP. When the devices do not reside in the same time zone, adjust the **Time Settings** parameter from the Flow Sensor configuration accordingly.
- ✓ The Flow Sensor may crash during spoofed attacks for not having enough RAM, when a monitored interface has the *Top Generator* parameter set to “Full”. It is highly recommended to set the **Top Generator** parameter to “Extended”, not to “Full”.
- ✓ If you define interfaces with the **Traffic Direction** parameter set to “Auto”, make sure that the IP Zone you have selected for the Flow Sensor contains all your IP blocks.

- ✓ In order to provide fast and up-to-date traffic statistics, the Flow Sensor accepts only flows generated in the last 5 minutes. All flows aged with delay exceeding 5 minutes (300 seconds) are automatically discarded with the event log warning *"Wrong flow timeout settings! Received flow <starting/ending> <X> seconds ago"*.

When the warnings refer to the starting time, make sure that the clocks are synchronized, the flow exporter is properly configured, and the time-zone and the **Flow Timeout** parameter are properly set.

When the warnings refer to the ending time, make sure that the clocks are synchronized, the time-zone is properly set and the flow exporter is properly configured.

You can double-check whether the time of the Flow Sensor and the start/end time of flows differ by more than 300 seconds. In Reports » Tools » Flow Collectors » Flow Records, select the Flow Sensor, set Output to Debug and generate a listing for the last 5 minutes:

- The Date_flow_received column indicates the time when the Flow Sensor received the flow packet
- The Date_first_seen column indicates the time when the flow started
- The Date_last_seen column indicates the time when the flow ended

Flow Sensor does not misinterpret the start/end time of flows. A few flow exporters are known to have bugs or limitations regarding flow aging. In this case, contact your vendor to make sure that the flow exporter is correctly configured and it is able to expire flows in under 5 minutes.

- ✓ Ensure that you have correctly configured the flow exporter to send flows to the server for each of the monitored interfaces. To list all interfaces that send flows, go to Reports » Tools » Flow Collectors » Flow Tops, select the Flow Sensor, set Output to Debug, set Top Type to Any Interface and generate the top for the last 10 minutes. The In/Out_If column shows the SNMP index of every interface that exports flows, whether or not it was configured as a monitored interface in the Flow Sensor configuration.
- ✓ If you see statistics for only one traffic direction (inbound or outbound), go to Reports » Tools » Flow Collectors » Flow Records, and generate a listing for the last 10 minutes. If all your IPs are listed in a single column, check the flow exporter's configuration and feature list. Not all devices can export flows in both directions (e.g. Brocade equipments generate only inbound sFlow) or with the same SNMP interface index.
- ✓ The traffic readings of the Flow Sensor may differ from the SNMP Sensor or from other SNMP-based monitoring tools. Flow Sensor counts In/Out traffic as traffic entering/exiting the IP Zone (when **IP Validation** is enabled), unlike SNMP tools that count In/Out traffic as traffic entering/exiting the interface. You can double-check the traffic readings of a Flow Sensor by configuring a SNMP Sensor that monitors the same flow exporter (page 46).
- ✓ If the Flow Sensor does not show the correct statistics after upgrading the router's firmware, the SNMP index of the interfaces may have changed. In this case, enter the new SNMP index for each monitored interface.
- ✓ To troubleshoot Sensor graph or IP graph issues, follow the Graphs Troubleshooting guide from page 22.
- ✓ Make sure you are running the latest version of the software.

Configuration » Components » SNMP Sensor

SNMP Sensor monitors the bandwidth usage of routers and switches on a port-by-port basis. SNMP Sensor queries devices (e.g. routers, switches and servers) for the traffic counters of each port with small data packets. These are triggering reply packets from the device. The advantages and disadvantages of monitoring traffic by SNMP are listed on page 8.

For detailed instructions on how to enable SNMP on your network device, consult its documentation.

To add a SNMP Sensor click the <+> button from the title bar of the Configuration » Components panel. To modify an existing SNMP Sensor, go to Configuration » Components and click its name.

The SNMP Sensor Configuration window contains the following fields:

- **Sensor Name** – A short name to help you identify the SNMP Sensor.
- **Device Group** – Optional description used within Console to group components by location, role, etc.
- **Reports Visibility** – Whether the SNMP Sensor should be listed inside the Reports » Components panel.
- **Sensor Server** – The server that runs the SNMP Sensor. It is recommended to run all SNMP Sensors on the Console server. The configuration of servers is described on page 37.
- **Polling Interval** – Polling is the process of sending the SNMP request periodically to the device to retrieve information. A low polling interval (of say 1 minute) gives you granular reports but may place an increased load on your server if you poll large amount of interfaces.
- **Sensor License** – The license used by the SNMP Sensor. WanGuard provides all features (although severely limited by the SNMP technology); WanSight does not provide traffic anomaly detection and reaction.
- **IP Zone** – When a WanGuard license is being used, the SNMP Sensor is able to check thresholds listed in the selected IP Zone with the following restrictions (SNMP does not provide any information about IPs or protocols):
 - Subnet must be “0.0.0.0/0”.
 - Domain must be “subnet”.
 - Value must be absolute, not percentage.
 - Decoder must be “TOTAL”.
- **Device IP:port** – Enter the IP address and SNMP port of the networking device. The standard SNMP port is 161.
- **Timeout (ms)** – The timeout value should be at least a little more than double the time it takes for a packet to travel the longest route between devices on your network. The default value is 1000 milliseconds (1 second).
- **Retries** – This value represents the number of times the SNMP Sensor will retry a failed SNMP request defined as any SNMP request that does not receive a response within the Timeout (ms) defined above. The default value is 2.

- **Discovery** – Activates or deactivates interface discovery:
 - *Monitor all interfaces* – Select to automatically add all interfaces to the SNMP Sensor. The interface names are based on the **Interface Name** setting available when pressing the <OIDs and Tester> button.
 - *Monitor defined interfaces* – Select to monitor only interfaces listed in the SNMP Sensor configuration.
- **Authentication Protocol** – Select the SNMP protocol used for authentication:
 - *SNMP v1* – The oldest version. Easy to set up – only requires a plaintext community. The biggest downsides are that it does not support 64 bit counters, only 32 bit counters, and that it has little security.
 - *SNMP v2c* – Version 2c is identical to version 1, except it adds support for 64 bit counters. This is very important when monitoring gigabit interfaces. Even a 1Gbps interface can wrap a 32 bit counter in 34 seconds, which means that a 32 bit counter being polled at one minute intervals is useless. Select this option instead of v1 in most cases.
 - *SNMP v3* – Adds security to the 64 bit counters. SNMP version 3 adds both encryption and authentication, which can be used together or separately. Setup is much more complex than just defining a community string.
- **Community String** – SNMP v1 and v2c credentials serve as a type of password that is authenticated by confirming a match between the string provided here and the SNMP Community String stored as a MIB object on an SNMP-enabled, managed device.
- **Security Level & Name** – SNMP v3-only. SNMP Sensor supports the following set of security levels as defined in the USM MIB (RFC 2574):
 - *noAuthnoPriv* – Communication without authentication and privacy.
 - *authNoPriv* – Communication with authentication and without privacy.
 - *authPriv* – Communication with authentication and privacy.
- **Authentication Protocol & Passphrase** – SNMP v3-only. The protocols used for Authentication are *MD5* and *SHA* (Secure Hash Algorithm).
- **Privacy Protocol & Passphrase** – SNMP v3-only. An indication of whether messages sent on behalf of this user can be protected from disclosure, and if so, the type of privacy protocol which is used. This takes the value *DES* (CBC-DES Symmetric Encryption) or *AES* (Advanced Encryption Standard).
- **Monitored Network Interfaces** – The interfaces that should be monitored. To avoid mirrored graphs, add only upstream interfaces. Settings per interface:
 - *SNMP Index* – The interfaces are identifiable by their unique indexes.
 - *Interface Name* – A short description used to identify the monitored interface. Descriptions longer than 10 characters clutter some reports. By default, the auto-filled interface name is retrieved from the ifAlias OID. To change the OID used for the interface name click the <OIDs and Tester> button.
 - *Graph Color* – The color used in graphs for the interface. The default color is a random one, which can be changed by entering a different HTML color code or by clicking the drop-down menu.
 - *Traffic Direction* – The direction of the traffic entering the interface, from the user's perspective:
 - “Unset” – Traffic entering the interface is considered “downstream”; traffic exiting the interface

is considered “upstream”.

- “Upstream” – Set for upstream interfaces, e.g. peering interfaces, interfaces connected to the Internet.
- “Downstream” – Set for downstream interfaces, e.g. customer interfaces, interfaces connected to your network.
- “Null” – Traffic to Null interfaces is ignored.
- *Link Speed In & Link Speed Out* – Enter the speed (bandwidth, capacity) of the interface. The values are used for percentage-based reports and percentage-based bits/s thresholds.
- **Comments** – Comments about the SNMP Sensor can be saved here. These comments are not visible elsewhere.

To start the SNMP Sensor, click the gray square button next to its name in Configuration » Components. Ensure that the SNMP Sensor starts properly by watching the event log (details on page 66).

If the SNMP Sensor starts without errors, but you can't see any data collected by it in Reports » Components » Overview after more than 5 minutes, follow the troubleshooting guide below.

SNMP Sensor Troubleshooting

- ✓ Look for warnings or errors produced by the SNMP Sensor in the event log (details on page 66).
- ✓ Ensure that you have correctly configured the SNMP Sensor. Each configuration field is described in detail in this chapter.
- ✓ Verify if the Console can reach the device by clicking the **<OIDs and Tests>** button from the SNMP Sensor Configuration window, then press **<Query Device>**.
- ✓ Permit the server to contact the SNMP device, by configuring its ACL.
- ✓ If Sensor graphs are very spiky, increase the Polling Interval value.
- ✓ Make sure you are running the latest version of the software.

Configuration » Components » Sensor Cluster

Sensor Cluster aggregates traffic data provided by Packet Sensors and Flow Sensors into a single anomaly detection domain and/or IP graphing domain.

To add a Sensor Cluster, click the <+> button found on the title bar of the Configuration » Components panel. To configure an existing Sensor Cluster, go to Configuration » Components, and click its name.

The Sensor Cluster Configuration window contains the following fields:

- **Sensor Name** – A short name to help you identify the Sensor Cluster.
- **Graph Color** – The color used in graphs for the Sensor Cluster. The default color is a random one, which can be changed by entering a different HTML color code or by clicking the drop-down menu.
- **Reports Visibility** – Whether the Sensor Cluster should be listed inside the Reports » Components panel.
- **Device Group** – Optional description used within Console to group components by location, role, etc.
- **Sensor Server** – The server that runs the Sensor Cluster. It is recommended to run Sensor Clusters on the Console server. The configuration of servers is described on page 37.
- **Link Speed IN / OUT** – The summed-up speeds (bandwidth, capacity) of the aggregated interfaces. The values are used for percentage-based reports and percentage-based bits/s thresholds.
- **Associated Sensors** – Select which Packet Sensors and Flow Sensor interfaces must be aggregated by the Sensor Cluster.
- **IP Zone** – Sensor Cluster extracts from the selected IP Zone per-subnet settings about thresholds and/or IP graphing. For more information about IP Zones consult the “IP Zone” chapter on page 34.
- **Anomaly Detection** – Select “Aggregated” to enable anomaly detection in the summed up traffic data by the Sensor Cluster, and disable anomaly detection by associated Sensors. Select “Not Aggregated” to enable anomaly detection by each associated Sensor and to disable anomaly detection by the Sensor Cluster.
- **IP Graphing** – Select “Aggregated” to enable IP graphing by the Sensor Cluster for the summed up traffic data, and disable IP graphing by the associated Sensors. Select “Not Aggregated” to enable IP graphing by each associated Sensor and to disable IP graphing by the Sensor Cluster.
- **Comments** – Comments about the Sensor Cluster can be saved here. These comments are not visible elsewhere.

To start the Sensor Cluster, click the gray square button next to its name in Configuration » Components.

Ensure that the Sensor Cluster starts properly by watching the event log (details on page 66) and by monitoring Reports » Components » Overview.

Configuration » Components » BGP Connection

Console users can view, send and withdraw BGP announcements (routing updates) from Reports » Tools » BGP Operations. The records are stored in Reports » Tools » BGP Operations » BGP Announcement Archive.

WanGuard Sensor and WanGuard Filter can use Responses (details on page 27) to send and withdraw BGP announcements automatically, in the following cases:

- To protect your network by announcing DDoSed destinations to the upstream provider(s) using a special BGP community. Your side will no longer route the attacked addresses making them effectively null-routed by your BGP peers. This network protection technique is called black hole routing, null-routing or RTBH (Remote Triggered Black Hole).
- To re-route attacked destinations through servers running WanGuard Filter, block attackers' packets and re-inject cleaned traffic back into the network. This network protection technique is called traffic scrubbing, clean pipe, side filtering or sink hole routing.

If you do not need any of those features, you can safely skip this chapter.

Before adding a BGP Connection, install and configure the BGPd daemon provided by the quagga package. The BGPd configuration steps are listed on Appendix 3 – page 105 and Appendix 4 – page 108.

To add a BGP Connection, click the <+> button from the title bar of the Configuration » Components panel. To modify an existing BGP Connection, go to Configuration » Components and click its name.

The BGP Connection Configuration window contains the following fields:

- **BGP Connection Name** – A short name or description for the BGP Connection.
- **BGP Connection Role** – Set the correct role to see the number of BGP announcements in Reports » Tools in red for “Black Hole”, and in blue for “Traffic Diversion”.
- **BGPd Server** – The server running the BGPd daemon. If the server is not the Console server, make sure that the BGPd daemon is accessible from the Console server. The configuration of servers is described on page 37.
- **AS Number** – The same AS number with the one from the BGPd configuration.
- **Login Password** – Password needed to connect to the BGPd daemon.
- **Enable Password** – Configuration mode password of the BGPd daemon.
- **Route Map** – The route-map parameter that should be appended to each announcement. This is not mandatory, but widely used.
- **AS View** – If multiple AS views are defined in the BGPd configuration, you must enter the AS view you want to use for this configuration. This is not mandatory.
- **BGPd - bgpd.conf** – The content of the bgpd.conf file downloaded through the WANsupervisor service. The file uses a format very similar to Cisco IOS configuration format. Quagga documentation covers all configuration options.

- **Reject External IPs** – When this option is selected, only the announcements for IPs/subnets defined inside an IP Zone (excluding 0.0.0.0/0) will be sent.
- **Reject IPv4 under /** – Restricts sending prefixes that have the IPv4 CIDR mask less than the configured value. For example, a value of 32 rejects all prefixes that are not hosts and prevents manual or automatic announcements of subnets. To disable this feature enter the value 0.
- **Reject IPv6 under /** – Restricts sending prefixes that have the IPv6 CIDR mask less than the configured value. For example, a value of 128 rejects all prefixes that are not hosts and prevents manual or automatic announcements of subnets. To disable this feature enter the value 0.
- **Restrict IPv4 over /** – Set to the maximum IPv4 CIDR mask accepted by your cloud-based DDoS mitigation providers. For example, if only /24 prefixes are accepted by your BGP peers, and you want to announce a whole C class for a single attacked IP, set to 24. To disable this feature enter the value 32.
- **Restrict IPv6 over /** – Set to the maximum IPv6 CIDR mask accepted by your cloud-based DDoS mitigation providers. To disable this feature enter the value 128.
- **Quagga Zebra Local Black Hole** – Check if you need the local black hole feature provided by the zebra daemon. This is a rarely-used feature, that may be useful only for in-line servers.
- **Quagga Zebra Login & Enable Passwords** – The passwords needed to connect to the zebra daemon.
- **Comments** – Comments about the BGP Connection can be saved here. These comments are not visible elsewhere.

Enable the BGP Connection by clicking the gray square button next to its name in Configuration » Components.

You can manually send a test BGP announcement with an unused/test IP address from Reports » Tools » BGP Operations Tools » <**Black Hole**> or <**Divert Traffic**>. If you encounter errors, follow the troubleshooting guide below:

BGP Connection Troubleshooting

- ✓ Ensure that you have correctly configured the BGP Connection. Each configuration field is described in detail in this chapter.
- ✓ Look for warnings or errors produced by the BGP Connection in Reports » Tools » BGP Operations » BGP Connection Events (details on page 66).
- ✓ Telnet connection errors on port tcp/2605 in the event log indicate that the BGPd daemon is not accessible through telnet from the Console server. By default, Debian systems bound bgpd to 127.0.0.1, which is why the string “-A 127.0.0.1” must be deleted from /etc/quagga/debian.conf.
- ✓ Telnet connection errors on port tcp/2601 indicate that the BGP Connection was configured with the Quagga Zebra Local Back Hole feature, but the zebra daemon is not configured or accessible from the Console server.
- ✓ Telnet errors about pattern time-outs indicate mismatches between a parameter defined in the BGP Connection (password, AS number, route-map, AS view) and the similar parameter from bgpd.conf.
- ✓ You can clear BGP prefix errors from Reports » Tools » BGP Operations » Active BGP Announcements » <Remove All>, or by clicking the stop button for each announcement.

- ✓ Errors about “orphaned” announcements indicate that the BGP announcement is active while the anomaly has ended. The BGP announcements are withdrawn by the Sensor that detected the anomaly, immediately after the anomaly ends.

Orphaned announcements can have multiple reasons:

- The anomaly was ended forcefully by clicking the <Expire Anomalies> button
 - The Sensor that detected the anomaly was deleted or had errors
 - The WANsupervisor was not running at the time of the withdrawal
 - Networking errors between the Console server and the server running bgpd. If this is the case, you should see the exact telnet error in the event log.
- ✓ Make sure you are running the latest version of the software.

Configuration » Components » Packet Filter

WanGuard Filter is described in the “Choosing a Method of DDoS Mitigation” chapter on page 10.

If you do not plan to use Packet Filter(s) you can safely skip this chapter.

To add a Packet Filter, click the <+> button found in the title bar of the panel Configuration » Components. To configure an existing Packet Filter, go to Configuration » Components and click its name.

The Packet Filter Configuration window contains the following fields:

- **Filter Name** – A short name that will help you identify the Packet Filter.
- **Graph Color** – The color used in graphs for the Packet Filter. The default color is a random one, which can be changed by entering a different HTML color code or by clicking the drop-down menu.
- **Device Group** – Optional description used within Console to group components by location, role, etc.
- **Reports Visibility** – Whether the Packet Filter should be listed inside the Reports » Components panel.
- **Filter Server** – The server that runs the Packet Filter. The configuration of servers is described on page 37.
- **Server Topology** – Select the network topology of the server running the Packet Filter:

- *Inline Filtering* – Packet Filter runs on a server that resides in the main data path, configured as an OSI Layer 3 router or as a network bridge.

To enable routing on the filtering server follow the steps required by your Linux distribution. At least the following command needs to be executed:

```
sysctl -w net.ipv4.ip_forward=1;
sysctl -w net.ipv4.conf.all.forwarding=1;
sysctl -w net.ipv4.conf.default.rp_filter=0;
sysctl -w net.ipv4.conf.all.rp_filter=0
```

To run Packet Filter in this mode, set the interface connected to the peering/border router as Inbound Interface. To inject the packets back into the network, set a core router as the default gateway, reachable through the Outbound Interface, either directly (recommended) or through a GRE/IP in IP tunnel.

To configure the filtering server as a network bridge, follow the steps required by your Linux distribution. To run Packet Filter in this mode, set the Inbound Interface to the bridged interface, usually br0.

- *Inline monitoring* – Packet Filter runs on a server that resides in the main data path, configured as an OSI Layer 3 router or as a network bridge. Direct filtering is disabled, but Packet Filter can still generate filtering rules that improve the visibility of attacks, which can be applied on other in-line appliances or firewalls. To run Packet Filter in this mode, set the parameters like in the Inline Filtering mode.
- *Out-of-line Filtering* – To run Packet Filter in this mode, set the Traffic Diversion parameter to a BGP

Connection configured to reroute traffic. Other parameters must be set as in the Inline Filtering mode.

- *Out-of-line Monitoring* – Packet Filter runs on a server that receives a copy of packets from a network TAP or a mirroring port. Direct filtering is not possible, but Packet Filter is still able to generate filtering rules that improve the visibility of attacks, which can be applied on other in-line appliances or firewalls. To run the Packet Filter in this mode, set the Inbound Interface to be the same as the Sniffing Interface configured in the Packet Sensor.
- **Capture Engine** – Select the packet capturing engine used by the Packet Filter:
 - *Embedded LibPcap* – Select to use the built-in LibPcap 1.6.2 library.
 - *System LibPcap* – Select to use the LibPcap library installed by the Linux distribution.
 - *Myricom Sniffer10G* – Select when using a Myricom network adapter with a Sniffer 10G license. Click the button on the right for driver-specific settings.
 - *PF_RING, RX+TX* – Select to use the PF_RING 6.2 library for RX and TX traffic.
 - *PF_RING, RX* – Select to use the PF_RING 6.2 library for RX traffic.
 - *PF_RING, TX* – Select to use the PF_RING 6.2 library for TX traffic.
- **CPU Affinity** – You can force the Packet Filter to run exclusively on a given set of CPU cores.
- **Sniffing Interface** – This switch configures the interface listened by the Packet Filter:
 - *Inbound* – Packet Filter analyzes the traffic received by the Inbound Interface. The CPU usage is high because it continuously inspects the malicious packets, even if they are not being forwarded.
 - *Outbound* – Packet Filter analyzes only the traffic passing through the Outbound Interface. Choosing this option makes Packet Filter consume less CPU because the malicious packets that are dropped do not reach the Outbound Interface. Packet Filter receives traffic statistics from the local firewall.
- **Traffic Diversion** – Provides a selection of BGP Connections that may be used for traffic diversion. If the server is deployed in-line, or if you do not plan to use traffic diversion, leave the BGP Connection field set to “None”.

When a BGP Connection is selected, the Packet Filter sends a BGP routing update that will make the server next hop for the attacked IP address. When the attack ends, the Packet Filter automatically withdraws the BGP announcement and the traffic towards the IP address will be routed normally. Make sure that the Sensor that detected the attack is still able to capture traffic rerouted to the Packet Filter. For more information about BGP Connections, consult the “BGP Connection” chapter on page 50.

- **Inbound Interface** – Enter the interface that receives traffic to your network. For VLAN-specific monitoring, use the *vconfig* command to define VLAN interfaces.
- **Outbound Interface** – The cleaned traffic is sent to a downstream router through the outbound interface, which should hold the route to the default gateway. For GRE / IP over IP tunneling, configure virtual network interfaces with the *ip* command, part of the *iproute2* package.
- **Software Firewall** – Select the software firewall policy applied when the Packet Filter generates a filtering rule. Packet Filter can do software-based packet filtering and packet rate limiting using the Netfilter framework provided by the Linux kernel. The software-based packet filter is very flexible and since Packet Filter does not use the connection tracking mechanism specific to stateful firewalls, it is very fast as well.

- *No software packet filtering* – Packet Filter detects and reports filtering rules. The Linux firewall API is not used. You can implement other filtering commands using custom scripts executed by Responses.
- *Filtering rules drop matched traffic. Valid traffic is accepted* – Packet Filter detects, reports and applies filtering rules using the software firewall. If the filtering rule is not whitelisted, the traffic matched by it is blocked. The remaining traffic is passed.
- *Filtering rules drop matched traffic. Valid traffic is rate-limited* – Packet Filter detects, reports and applies filtering rules and rate-limits the remaining traffic. If the filtering rule is not whitelisted, the traffic matched by it is blocked. The traffic that exceeds the packets/second threshold value is not passed.
- *Filtering rules rate-limit matched traffic. Valid traffic is accepted* – Packet Filter detects and reports filtering rules and rate-limits matched traffic to the threshold value.
- *Apply the default Netfilter chain policy* – Packet Filter detects and reports filtering rules, and applies the default Netfilter chain policy. The Netfilter framework is still being used, but all rules have the “RETURN” target. This option is usually used for testing purposes.

Click the options button on the right to be able to configure the following Software Firewall parameters:

- **Netfilter Chain** – set to *FORWARD* if the server forwards traffic or *INPUT* if it does not.
- **Netfilter Table** – the *raw* option requires both Inbound and Outbound interfaces to be set. It provides a better packet filtering performance compared to the *filter* option.
- **Hardware Firewall** – If you have a NIC that provides hardware filters, select the appropriate option. Since hardware filters do not consume CPU, use this option to complement the Software Firewall.
 - *No hardware packet filtering* – Hardware filters are not applied.
 - *Intel x520 or x540 10 Gigabit adapter (blocks IPv4 sources)* – Packet Filter programs the Intel chipset to drop IPv4 addresses from filtering rules that contain source IPs. Up to 4086 hardware filters possible.
 - *Intel x520 or x540 10 Gigabit adapter (blocks IPv4 destinations)* – Packet Filter programs the Intel chipset to drop IPv4 addresses from filtering rules that contain destination IPs. Up to 4086 hardware filters possible.
 - *Silicom Director 10 Gigabit adapter with PF_RING HW filters* – Packet Filter uses the PF_RING framework to apply the following hardware-based filtering rules on Silicom Director adapters: source/destination IPv4, source/destination TCP/UDP port, IP protocol.
 - *Chelsio T4 or T5 10/40 Gigabit adapter with LE-TCAM filters* – Packet Filter uses the Chelsio API to apply up to 487 filtering rules that contain any combination of source/destination IPv4/IPv6 addresses, source/destination UDP/TCP port and IP protocol.
- **Sampling (1/x)** – The default value is 1. Must be equal to the number of filtering servers activated for the same anomaly, when the Packet Filter is used in a clustered architecture where each filtering server receives traffic from a round-robin packet scheduler.
- **Rules Timeout** – When set to 0, filtering rules remain active for as long as the anomaly is active. Enter a non-zero value for the filtering rules to expire only after the entered amount of seconds.
- **Whitelist** – A Filter Whitelist is a collection of user-created rules that prevent the filtering of critical traffic. To add similar rules for multiple Filters, use Whitelist Templates (Configuration » Network &

Policy » <+> » Whitelist Template).

Packet Filter might block attack patterns that you do not want to be blocked. Destination ports and destination IP addresses are blocked only in worst-case scenarios when no other attack pattern is found. In some cases, it is better to let potential malicious traffic enter the network than to filter critical traffic. For example, if your DNS server is attacked by spoofed addresses on port 53 UDP, Packet Filter might block port 53 UDP traffic towards your DNS server, making it partially unreachable from the Internet. In this case, configure a proper whitelist rule (Rule Type: *Dst Port UDP*, Operator: *equal*, Rule Value: 53) and review Configuration » General Settings » Mitigation Options.

To add a new rule to the whitelist, enter the following information:

- **Description** – Add a description for the whitelist rule.
- **Prefix** – Enter a subnet that must include the anomaly IP address, for the whitelist rule to be valid. Enter 0.0.0.0/0 for a generic whitelist rule.
- **Decoder** – Select the decoder of the anomaly, or select *All* for a generic whitelist rule.
- **Rule Type** – Possible values: *Source IP*, *Src Port TCP*, *Dst Port TCP*, *Src Port UDP*, *Dst Port UDP*, *Packet Length*, *IP TimeToLive*, *IP Protocol*.
- **Operator** – Operators for strings and numbers: *equal*, *non-equal*. Operators for numbers: *less than*, *greater than*.
- **Rule Value** – A user-defined value that should match.
- **FW Policy** – When **FW Policy** is *Permit* and **Operator** is *equal*, the Flow Filter explicitly allows the matched traffic to pass through the Software Firewall. Otherwise, a more generic filtering rule might match the white-listed filtering rule.

When a filtering rule cannot be applied because it conflicts with a whitelist rule, a white flag icon will appear next to it in Console reports.

- **Comments** – Comments about the Packet Filter can be saved here. These comments are not visible elsewhere.

Enable the Packet Filter from Configuration » Components by clicking the gray square button next to its name.

An instance of the Packet Filter will be launched when the “Activate a Filter...” Response action is triggered by a traffic anomaly.

Packet Filter Troubleshooting

- ✓ To view the counters for each Netfilter or Chelsio filtering rule go to Reports » Tools » Firewall Rules.
- ✓ To view the filtering rules applied by the Netfilter framework (the Software Firewall option) from the CLI:

```
iptables -L -n -v && iptables -L -n -v -t raw
```

To delete all chains:

```
for chain in `iptables -L -t raw | grep wanguard | awk '{ print $2 }'`; do
iptables -X $chain; done
```


- ✓ To view the filtering rules applied by the Intel 80599 chipset:
`ethtool --show-ntuple <filtering_interface>`
or, for kernels >3.1:
`ethtool --show-nfc <filtering_interface>`
- ✓ To ensure that filtering rules can be applied on the Intel 80599 chipset, load the ixgbe driver with the parameter `FdirPballoc=3`.
- ✓ To view filtering rules applied by the Chelsio T4/T5 chipset:
`cxgbtool <filtering_interface> filter show`
- ✓ If the CPU usage of the Packet Filter instance is too high, install PF_RING (no ZC/DNA/LibZero needed), or use a network adapter that allows distributing Packet Filters over multiple CPU cores.
- ✓ For PF_RING installation issues, contact ntop.org. To increase the maximum number of PF_RING programs from 64 to 256, increase the `MAX_NUM_RING_SOCKETS` defined in `kernel/linux/pf_ring.h` and recompile the `pf_ring` kernel module.
- ✓ Make sure you are running the latest version of the software.

Configuration » Components » Flow Filter

WanGuard Filter is described in the “Choosing a Method of DDoS Mitigation” chapter on page 10.

If you do not plan to use Flow Filter(s) you can safely skip this chapter.

To add a Flow Filter, click the <+> button found in the title bar of the panel Configuration » Components. To configure an existing Flow Filter, go to Configuration » Components and click its name.

The Flow Filter Configuration window contains the following fields:

- **Filter Name** – A short name that will help you identify the Flow Filter.
- **Graph Color** – The color used in graphs for the Flow Filter. The default color is a random one, which can be changed by entering a different HTML color code or by clicking the drop-down menu.
- **Device Group** – Optional description used within Console to group components by location, role, etc.
- **Reports Visibility** – Whether the Flow Filter should be listed inside the Reports » Components panel.
- **Filter Server** – The server that runs the Flow Filter. The configuration of servers is described on page 37.
- **Server Topology** – Select the network topology of the server running the Flow Filter:
 - *Inline Filtering* – Flow Filter runs on a server that resides in the main data path, configured as an OSI Layer 3 router or as a network bridge.

To enable routing on the filtering server follow the steps required by your Linux distribution. At least the following command needs to be executed:

```
sysctl -w net.ipv4.ip_forward=1;
sysctl -w net.ipv4.conf.all.forwarding=1;
sysctl -w net.ipv4.conf.default.rp_filter=0;
sysctl -w net.ipv4.conf.all.rp_filter=0
```

To run Flow Filter in this mode, set the interface connected to the peering/border router as Inbound Interface. To inject the packets back into the network, set a core router as the default gateway, reachable through the Outbound Interface, either directly (recommended) or through a GRE/IP in IP tunnel.

To configure the filtering server as a network bridge, follow the steps required by your Linux distribution. To run Flow Filter in this mode, set the Inbound Interface to the bridged interface, usually br0.

- *Inline monitoring* – Flow Filter runs on a server that resides in the main data path, configured as an OSI Layer 3 router or as a network bridge. Direct filtering is disabled, but Flow Filter can still generate filtering rules that improve the visibility of attacks, which can be applied on other in-line appliances or firewalls. To run Flow Filter in this mode, set the parameters like in the Inline Filtering mode.
- *Out-of-line Filtering* – To run Flow Filter in this mode, set the Traffic Diversion parameter to a BGP Connection configured to reroute traffic. Other parameters must be set as in the Inline Filtering mode.

mode.

- *Out-of-line Monitoring* – Flow Filter runs on a server that receives a copy of packets from a network TAP or a mirroring port. Direct filtering is not possible, but Flow Filter is still able to generate filtering rules that improve the visibility of attacks, which can be applied on other in-line appliances or firewalls. To run the Flow Filter in this mode, set the Inbound Interface to be the same as the Sniffing Interface configured in the Packet Sensor.
- **Filtering Interface** – Select where to apply filtering rules:
 - *None* – Flow Filter does not apply filtering rules directly.
 - *Inbound* – Flow Filter applies filtering rules on the inbound Interface.
 - *Outbound* – Flow Filter applies filtering rules on the outbound interface.
- **Traffic Diversion** – Provides a selection of BGP Connections that may be used for traffic diversion. If the server is deployed in-line, or if you do not plan to use traffic diversion, leave the BGP Connection field set to “None”.

When a BGP Connection is selected, the Flow Filter sends a BGP routing update that will make the server next hop for the attacked IP address. When the attack ends, the Flow Filter automatically withdraws the BGP announcement and the traffic towards the IP address will be routed normally. Make sure that the Sensor that detected the attack is still able to capture traffic rerouted to the Flow Filter.

- **Inbound Interface** – Enter the interface that receives traffic to your network. For VLAN-specific monitoring, use the *vconfig* command to define VLAN interfaces.
- **Outbound Interface** – The cleaned traffic is sent to a downstream router through the outbound interface, which should hold the route to the default gateway. For GRE / IP over IP tunneling, configure virtual network interfaces with the *ip* command, part of the *iproute2* package.
- **Software Firewall** – Select the software firewall policy applied when the Flow Filter generates a filtering rule.

Flow Filter can do software-based packet filtering and packet rate limiting using the Netfilter framework provided by the Linux kernel. The software-based packet filter is very flexible and since Flow Filter does not use the connection tracking mechanism specific to stateful firewalls, it is very fast as well.

- *No software packet filtering* – Flow Filter detects and reports filtering rules. The Linux firewall API is not used. You can implement other filtering commands using custom scripts executed by Responses.
- *Filtering rules drop matched traffic. Valid traffic is accepted* – Flow Filter detects, reports and applies filtering rules using the software firewall. If the filtering rule is not whitelisted, the traffic matched by it is blocked. The remaining traffic is passed.
- *Filtering rules drop matched traffic. Valid traffic is rate-limited* – Flow Filter detects, reports and applies filtering rules and rate-limits the remaining traffic. If the filtering rule is not whitelisted, the traffic matched by it is blocked. The traffic that exceeds the packets/second threshold value is not passed.
- *Filtering rules rate-limit matched traffic. Valid traffic is accepted* – Flow Filter detects and reports filtering rules and rate-limits matched traffic to the threshold value.
- *Apply the default Netfilter chain policy* – Flow Filter detects and reports filtering rules, and applies the default Netfilter chain policy. The Netfilter framework is still being used, but all rules have the “RETURN” target. This option is usually used for testing purposes.

Click the options button on the right to be able to configure the following Software Firewall parameters:

- **Netfilter Chain** – set to *FORWARD* if the server forwards traffic or *INPUT* if it does not.
- **Netfilter Table** – the *raw* option requires both Inbound and Outbound interfaces to be set. It provides a better packet filtering performance compared to the *filter* option.
- **Hardware Firewall** – If you have a NIC that provides hardware filters, select the appropriate option. Since hardware filters do not consume CPU, use this option to complement the Software Firewall.
 - *No hardware packet filtering* – Hardware filters are not applied.
 - *Intel x520 or x540 10 Gigabit adapter (blocks IPv4 sources)* – Flow Filter programs the Intel chipset to drop IPv4 addresses from filtering rules that contain source IPs. Up to 4086 hardware filters possible.
 - *Intel x520 or x540 10 Gigabit adapter (blocks IPv4 destinations)* – Flow Filter programs the Intel chipset to drop IPv4 addresses from filtering rules that contain destination IPs. Up to 4086 hardware filters possible.
 - *Silicom Director 10 Gigabit adapter with PF_RING HW filters* – Flow Filter uses the PF_RING framework to apply the following hardware-based filtering rules on Silicom Director adapters: source/destination IPv4, source/destination TCP/UDP port, IP protocol.
 - *Chelsio T4 or T5 10/40 Gigabit adapter with LE-TCAM filters* – Flow Filter uses the Chelsio API to apply up to 487 filtering rules that contain any combination of source/destination IPv4/IPv6 addresses, source/destination UDP/TCP port and IP protocol.
- **Sampling (1/x)** – The default value is 1. Must be equal to the number of filtering servers activated for the same anomaly, when the Flow Filter is used in a clustered architecture where each filtering server receives traffic from a round-robin packet scheduler.
- **Rules Timeout** – When set to 0, filtering rules remain active for as long as the anomaly is active. Enter a non-zero value for the filtering rules to expire only after the entered amount of seconds.
- **Whitelist** – A Filter Whitelist is a collection of user-created rules that prevent the filtering of critical traffic. To add similar rules for multiple Filters, use Whitelist Templates (Configuration » Network & Policy » <+> » Whitelist Template).

Flow Filter might block attack patterns that you do not want to be blocked. Destination ports and destination IP addresses are blocked only in worst-case scenarios when no other attack pattern is found. In some cases, it is better to let potential malicious traffic enter the network than to filter critical traffic. For example, if your DNS server is attacked by spoofed addresses on port 53 UDP, Flow Filter might block port 53 UDP traffic towards your DNS server, making it partially unreachable from the Internet. In this case, configure a proper whitelist rule (Rule Type: *Dst Port UDP*, Operator: *equal*, Rule Value: 53) and review Configuration » General Settings » Mitigation Options.

To add a new rule to the whitelist, enter the following information:

- **Description** – Add a description for the whitelist rule.
- **Prefix** – Enter a subnet that must include the anomaly IP address, for the whitelist rule to be valid. Enter 0.0.0.0/0 for a generic whitelist rule.
- **Decoder** – Select the decoder of the anomaly, or select *All* for a generic whitelist rule.
- **Rule Type** – Possible values: *Source IP*, *Src Port TCP*, *Dst Port TCP*, *Src Port UDP*, *Dst Port UDP*, *Packet Length*, *IP TimeToLive*, *IP Protocol*.

- **Operator** – Operators for strings and numbers: *equal*, *non-equal*. Operators for numbers: *less than*, *greater than*.
- **Rule Value** – A user-defined value that should match.
- **FW Policy** – When **FW Policy** is *Permit* and **Operator** is *equal*, the Flow Filter explicitly allows the matched traffic to pass through the Software Firewall. Otherwise, a more generic filtering rule might match the white-listed filtering rule.

When a filtering rule cannot be applied because it conflicts with a whitelist rule, a white flag icon will appear next to it in Console reports.

- **Comments** – Comments about the Flow Filter can be saved here. These comments are not visible elsewhere.

Enable the Flow Filter from Configuration » Components by clicking the gray square button next to its name.

An instance of the Flow Filter will be launched when the “Activate a Filter...” Response action is triggered by a traffic anomaly.

Configuration » Components » Filter Cluster

WanGuard Filter is described in the “Choosing a Method of DDoS Mitigation” chapter on page 10.

If you do not plan to use Filter Cluster(s) you can safely skip this chapter.

To add a Filter Cluster, click the <+> button found in the title bar of the panel Configuration » Components. To configure an existing Filter Cluster, go to Configuration » Components and click its name.

The Filter Cluster Configuration window contains the following fields:

- **Filter Name** – A short name that will help you identify the Filter Cluster.
- **Graph Color** – The color used in graphs for the Filter Cluster. The default color is a random one, which can be changed by entering a different HTML color code or by clicking the drop-down menu.
- **Device Group** – Optional description used within Console to group components by location, role, etc.
- **Reports Visibility** – Whether the Filter Cluster should be listed inside the Reports » Components panel.
- **Filter Server** – The server that runs the Filter Cluster. The configuration of servers is described on page 37.
- **Apply Rules By** – Select where to apply filtering rules detected by the Filter Cluster:
 - *Associated Filters* – The filtering rules are applied on each server running an associated Filter.
 - *Filter Cluster* – The filtering rules are applied only on the server running the Filter Cluster.
- **Filtering Interface** – Select the interface where to apply filtering rules:
 - *None* – Flow Filter does not apply filtering rules directly.
 - *Inbound* – Filter Cluster applies filtering rules on the inbound Interface.
 - *Outbound* – Filter Cluster applies filtering rules on the outbound interface.
- **Traffic Diversion** – Provides a selection of BGP Connections that may be used for traffic diversion. If the server is deployed in-line, or if you do not plan to use traffic diversion, leave the BGP Connection field set to “None”.

When a BGP Connection is selected, the Filter Cluster sends a BGP routing update that will make the server next hop for the attacked IP address. When the attack ends, the Filter Cluster automatically withdraws the BGP announcement and the traffic towards the IP address will be routed normally. Make sure that the Sensor that detected the attack is still able to capture traffic rerouted to the Filter Cluster.

- **Inbound Interface** – Enter the interface that receives traffic to your network. For VLAN-specific monitoring, use the *vconfig* command to define VLAN interfaces.
- **Outbound Interface** – The cleaned traffic is sent to a downstream router through the outbound interface, which should hold the route to the default gateway. For GRE / IP over IP tunneling, configure virtual network interfaces with the *ip* command, part of the *iproute2* package.

- **Associated Filters** – Select the Filters that should be aggregated by the Filter Cluster. The associated Filters are launched by the Filter Cluster instance and need not be launched individually by Response.
- **Software Firewall** – Select the software firewall policy applied when the Filter Cluster generates a filtering rule.

Filter Cluster can do software-based packet filtering and packet rate limiting using the Netfilter framework provided by the Linux kernel. The software-based packet filter is very flexible and since Filter Cluster does not use the connection tracking mechanism specific to stateful firewalls, it is very fast as well.

- *No software packet filtering* – Filter Cluster detects and reports filtering rules. The Linux firewall API is not used. You can implement other filtering commands using custom scripts executed by Responses.
- *Filtering rules drop matched traffic. Valid traffic is accepted* – Filter Cluster detects, reports and applies filtering rules using the software firewall. If the filtering rule is not whitelisted, the traffic matched by it is blocked. The remaining traffic is passed.
- *Filtering rules drop matched traffic. Valid traffic is rate-limited* – Filter Cluster detects, reports and applies filtering rules and rate-limits the remaining traffic. If the filtering rule is not whitelisted, the traffic matched by it is blocked. The traffic that exceeds the packets/second threshold value is not passed.
- *Filtering rules rate-limit matched traffic. Valid traffic is accepted* – Filter Cluster detects and reports filtering rules and rate-limits matched traffic to the threshold value.
- *Apply the default Netfilter chain policy* – Filter Cluster detects and reports filtering rules, and applies the default Netfilter chain policy. The Netfilter framework is still being used, but all rules have the “RETURN” target. This option is usually used for testing purposes.

Click the options button on the right to be able to configure the following Software Firewall parameters:

- **Netfilter Chain** – set to *FORWARD* if the server forwards traffic or *INPUT* if it does not.
- **Netfilter Table** – the *raw* option requires both Inbound and Outbound interfaces to be set. It provides a better packet filtering performance compared to the *filter* option.
- **Hardware Firewall** – If you have a NIC that provides hardware filters, select the appropriate option. Since hardware filters do not consume CPU, use this option to complement the Software Firewall.
 - *No hardware packet filtering* – Hardware filters are not applied.
 - *Intel x520 or x540 10 Gigabit adapter (blocks IPv4 sources)* – Filter Cluster programs the Intel chipset to drop IPv4 addresses from filtering rules that contain source IPs. Up to 4086 hardware filters possible.
 - *Intel x520 or x540 10 Gigabit adapter (blocks IPv4 destinations)* – Filter Cluster programs the Intel chipset to drop IPv4 addresses from filtering rules that contain destination IPs. Up to 4086 hardware filters possible.
 - *Silicom Director 10 Gigabit adapter with PF_RING HW filters* – Filter Cluster uses the PF_RING framework to apply the following hardware-based filtering rules on Silicom Director adapters: source/destination IPv4, source/destination TCP/UDP port, IP protocol.
 - *Chelsio T4 or T5 10/40 Gigabit adapter with LE-TCAM filters* – Filter Cluster uses the Chelsio API to apply up to 487 filtering rules that contain any combination of source/destination IPv4/IPv6

addresses, source/destination UDP/TCP port and IP protocol.

- **Sampling (1/x)** – The default value is 1. Must be equal to the number of filtering servers activated for the same anomaly, when the Filter Cluster is used in a clustered architecture where each filtering server receives traffic from a round-robin packet scheduler.
- **Rules Timeout** – When set to 0, filtering rules remain active for as long as the anomaly is active. Enter a non-zero value for the filtering rules to expire only after the entered amount of seconds.
- **Whitelist** – A Filter Whitelist is a collection of user-created rules that prevent the filtering of critical traffic. To add similar rules for multiple Filters, use Whitelist Templates (Configuration » Network & Policy » <+> » Whitelist Template).

Filter Cluster might block attack patterns that you do not want to be blocked. Destination ports and destination IP addresses are blocked only in worst-case scenarios when no other attack pattern is found. In some cases, it is better to let potential malicious traffic enter the network than to filter critical traffic. For example, if your DNS server is attacked by spoofed addresses on port 53 UDP, Filter Cluster might block port 53 UDP traffic towards your DNS server, making it partially unreachable from the Internet. In this case, configure a proper whitelist rule (Rule Type: *Dst Port UDP*, Operator: *equal*, Rule Value: *53*) and review Configuration » General Settings » Mitigation Options.

To add a new rule to the whitelist, enter the following information:

- **Description** – Add a description for the whitelist rule.
- **Prefix** – Enter a subnet that must include the anomaly IP address, for the whitelist rule to be valid. Enter 0.0.0.0/0 for a generic whitelist rule.
- **Decoder** – Select the decoder of the anomaly, or select *All* for a generic whitelist rule.
- **Rule Type** – Possible values: *Source IP*, *Src Port TCP*, *Dst Port TCP*, *Src Port UDP*, *Dst Port UDP*, *Packet Length*, *IP TimeToLive*, *IP Protocol*.
- **Operator** – Operators for strings and numbers: *equal*, *non-equal*. Operators for numbers: *less than*, *greater than*.
- **Rule Value** – A user-defined value that should match.
- **FW Policy** – When **FW Policy** is *Permit* and **Operator** is *equal*, the Flow Filter explicitly allows the matched traffic to pass through the Software Firewall. Otherwise, a more generic filtering rule might match the white-listed filtering rule.

When a filtering rule cannot be applied because it conflicts with a whitelist rule, a white flag icon will appear next to it in Console reports.

- **Comments** – Comments about the Filter Cluster can be saved here. These comments are not visible elsewhere.

Enable the Filter Cluster from Configuration » Components by clicking the gray square button next to its name.

An instance of the Filter Cluster will be launched when the “Activate a Filter...” Response action is triggered by a traffic anomaly. The Filter Cluster instance will automatically launch the associated Filters, so there is no need to add them to Response.

Configuration » Schedulers » Scheduled Reports

One of the greatest strengths of the Console is the ease in which it can generate complex Reports. Most reports created by clicking items from the Reports Region can be printed, exported as PDFs or sent by email.

If you want to receive periodic reports by email without having to log into Console, go to Configuration » Schedulers and click the <+> button from the title bar of the panel.

You can include more than one email address in the **Email To** field by separating addresses with a comma.

The emails are sent periodically according to the settings in the **Scheduler** tab.

To see how the email would look like without waiting for the preconfigured time, enter your email address, and then click the <**Save & Execute Now**> button. You should receive the email containing the report within a few seconds. If you do not, verify the settings from Configuration » General Settings » Outgoing Email.

All emails are formatted as HTML messages and include MIME attachments.

Configuration » Schedulers » Event Reporting

Events are short text messages that describe the change of an operational status. They are generated by WanGuard WanSight components and logged by Console.

You can list events in Reports » Components » [Component Name] » [Component Type] Event sub-tab. To search, sort or filter event messages, click the small down arrow that appears when hovering over the Event column header. To see additional details about an event click the <+> button from the first column.

To see a recent list of **Latest Events**, click the small bottom edge of the window to raise the South Region, or press Ctrl+E. On one side the Latest Events tab displays the latest 60 events, while on the other side it displays a list of components that can generate events and the number of events generated in the last 24 hours by each component. The number's color indicates the maximum severity of the events: red means that there are ERRORS, blue is for INFO events, etc.

The event's **severity** indicates its importance:

- **MELTDOWN** – Meltdown events are generated in very serious situations, such as hardware failures.
- **CRITICAL** – Critical events are generated when significant software errors occur, such as a memory exhaustion situation.
- **ERROR** – Error events are usually caused by misconfigurations, communication errors between components, or bugs. Sensors auto-recover from errors by restarting themselves.
- **WARNING** – Warning events are generated when authentication errors occur, on I/O bottlenecks or when there are time synchronization issues.
- **INFO** – Informational events are generated when configurations are changed or when users log into Console.
- **DEBUG** – Debug events are generated to help troubleshooting coding errors.

As an administrator, you should keep events with high severities under surveillance! Configure Console to send important events periodically by email, Syslog or SNMP in Configuration » Schedulers » Event Reporting.

Configuration » General Settings » Outgoing Email

Console sends notification emails using the settings from Configuration » General Settings » Outgoing Email.

Email configuration options:

- **From Email** – The email address you would like to appear as the sender.
- **From Name** – The name as you would like it to appear on messages.
- **Mailer** – Console supports several mailing systems:
 - *PHP Mail* – Use the PHP mail() function. To use it, you may have to configure a Mail Transfer Agent (postfix, qmail, sendmail) on the Console server.
 - *SMTP* – Use the integrated SMTP support to send emails directly, without using a local Mail Transfer Agent.
 - *Sendmail* – Send mails using the sendmail command. To use it, you may have to configure a Mail Transfer Agent (postfix, qmail, sendmail) on the Console server.
- **SMTP Security** – Security options:
 - *None* – No encryption.
 - *SSL* – Enable SSL encryption.
 - *TLS* – Enable TLS encryption.
- **SMTP Host** – Specify the SMTP server(s). You can include backup SMTP server(s) separated by the “;” character.
- **SMTP Port** – TCP port to connect to, usually 25 (insecure) or 587 (secure, uses SSL/TLS).
- **SMTP Login/Password** – Credentials used for SMTP authentication. When the fields are empty, no authentication is performed.
- **Email Tester** – Send a test email to verify the settings.

Configuration » General Settings » User Management

To add, modify or delete Console user accounts click Configuration » General Settings » User Management.

Each Console user must be assigned to one role (access level):

- **Administrator** – Has full privileges. Can manage other user accounts. Is the only role allowed to access Configuration » General Settings » License Manager.
- **Operator** – Can change any configuration but is not allowed to modify user accounts.
- **Guest** – Has read-only access to Console, without access to any configuration. Can have a granular, permission-based access to specific reports, dashboards, Sensors, IP groups, tools, etc.

To add a Console account, press <**Add User**> and then select the desired role. You can modify an account by double-clicking it, or by selecting it and by pressing the <**Modify User**> button.

The **Active** checkbox enables or disables the selected account.

There are two **Authentication** options:

- **Local Password** – The user is authenticated with the password entered in the **Password** field. All passwords are stored encrypted.
- **Remote Authentication** – The user is authenticated by remote LDAP or RADIUS servers configured in Configuration » General Settings » User Management (details on page 69).

The **Full Name**, **Company**, **Position**, **Email**, **Phone** and **Comments** fields are optional. These details are not used anywhere else.

Landing Tab shows the tab that will open immediately after logging in. The list is dynamic and expands as you add Sensors, dashboards, IP groups etc. Set the Landing Tab to a relevant dashboard or report.

Minimum Severity shows the minimum severity level of events displayed in Console.

Reports Region lets you switch the position of the Reports Region (described on page 19) to east or west.

Configuration Region lets you switch the position of the Configuration Region (described on page 19) to east or west.

Console Theme lets you change the Console look after re-login. The most popular themes are the corporate “Gray” and the futuristic/industrial “Azenis”.

Configuration » General Settings » User Authentication

To configure remote authentication mechanisms and login window settings click Configuration » General Settings » User Authentication.

Persistent Sessions enable cookie-based authentication for Console users that select the *Remember* option in the login screen. Subsequent sessions will skip the login screen for the next 30 days or until the user logs off.

Authentication Mode enables or disables the authentication of Console users that are not defined in Configuration » General Settings » User Management, but defined in LDAP or Radius.

Console permits the use of external Radius and LDAP servers for end user authentication.

LDAP server settings:

- **Priority** – You can set the order in which Console connects to multiple authentication services. The authentication process stops after the first successful authentication.
- **Template User** – Remotely authenticated users without a Console account have the privileges of the Template User.
- **LDAP Host** – IP or hostname of the LDAP server. To connect to a LDAP server by SSL, set this parameter as *ldaps://<IP>/*.
- **Login Attribute** – Enter the LDAP attribute that contains the username. For Active Directory it may be *mailNickname* or *sAMAccountName*, for OpenLDAP or IBM Directory Server it may be *uid*.
- **LDAP Base DN** – Specify the location in the LDAP hierarchy where Console should begin searching for usernames for authorization requests. The base DN may be something equivalent to the organization, group, or domain name (AD) of external directory: *dc=domain,dc=com*.
- **Bind User DN/Password** – Distinguished name and password for a LDAP user permitted to search within the defined Base DN.
- **Search Filter** – May contain rules that restrict which users are authenticated using the current configuration. For example, the string "*|(department=*NOC*)(department=ISP)*" will only allow users from departments containing the string "NOC" or (|) from the "ISP" department to authenticate in Console.

RADIUS server settings:

- **Priority** – You can set the order in which Console connects to multiple authentication services. The authentication process stops after the first successful authentication.
- **Template User** – Remotely authenticated users without a Console account have the privileges of the Template User.
- **RADIUS Host** – IP or hostname of the Radius server.

- **RADIUS Port** – Port through which the Radius server is listening for authentication requests.
- **RADIUS Protocol** – Protocol used for authentication purposes:
 - **PAP** (Password Authentication Protocol) – provides a simple method for the peer to establish its identity using a 2-way handshake
 - **CHAP** (Challenge-Handshake Authentication Protocol) – authenticates a user or network host to an authentication entity
 - **MSCHAP** – is the Microsoft version of the Challenge-handshake authentication protocol, CHAP
 - **MSCHAP2** – is another version of Microsoft version of the Challenge-handshake authentication protocol, CHAP
- **RADIUS Secret** – Enter the credentials for connecting to the Radius server.

The contents of the **Login Window Notification** field is shown inside the Console login window.

The contents of the **Successful Window Notification** field is shown inside the Console window after logging in.

Reports » Tools

The **Reports » Tools** panel contains links to the **Anomalies**, **BGP Operations**, **Firewall Rules**, **Flow Collectors** and **Packet Tracers** tabs.

Reports » Tools » Anomalies

The tab provides live and historical data related to DoS, DDoS or other traffic anomalies.

The number of active traffic anomalies is displayed inside the Reports » Tools panel. This number is refreshed every 10 seconds. The color of the number reflects the highest severity of the active anomalies.

The Anomalies tab contains 3 sub-tabs located at the lower left side of the window:

Active Anomalies

The tab contains a table visible only while Sensors detect active traffic anomalies. The rows represent active anomalies, sorted by start time in descending order. The columns are:

No	The unique index of the anomaly. Click it to open a detailed anomaly report.
Prefix	The IP address/class of the traffic anomaly and its reverse DNS. In front of the prefix, the arrow indicates the direction of traffic: inbound when the arrow is pointing towards the prefix, or outbound when the arrow is pointing away from the prefix. Click it to open a new tab with data specific to that prefix.
IP Group	The IP group of the prefix. Click it to open a new tab with data specific to that IP group.
Anomaly	A short description of the anomaly.
Value	The peak value of the abnormal traffic. The latest value is displayed between parentheses.
Sensor	Which Sensor detected the anomaly. Click it to open a new tab with data specific to that Sensor.
From	The time and date when the anomaly started.
Latest Alarm	How much time has passed since the most recent detection of the anomaly.
Pkts/s – Bits/s	The latest packets/second and bits/second throughput of the TOTAL decoder.
Actions	Actions available for administrators, operators and guests with proper permissions: <ul style="list-style-type: none"> <i>Generate Anomaly Report</i> – generates a full anomaly report that can be emailed.

	<ul style="list-style-type: none"> • <i>View Traffic Graph</i> – available if IP Graphing is enabled for the prefix. • <i>Open Packet Trace</i> – available for Packet Sensors when the Response contains a traffic capturing action. • <i>Open Flow Trace</i> – available for Flow Sensors with the Flow Collector feature enabled. Shows bi-directional flows that started or ended during the selected time interval. Flow Traces may have an up to 5 minute delay due to flow file buffering. Time-zone differences are not adjusted. • <i>Delete BGP Prefix</i> – available if a BGP announcement with the prefix was sent. • <i>Classify/Set Comment</i> – add or modify comments, or classify the impact of anomalies. It is used only for reporting purposes and does not impact IP profiling. • <i>Manual Actions</i> – execute all Response actions configured for manual execution. • <i>Expire Anomaly</i> – force Sensor to clear the anomaly. The Sensor must be running for the action to take effect.
Dropped	The percentage of abnormal traffic filtered by one or more WanGuard Filters.
Severity	<p>The exact rule severity and link severity are displayed as a tool-tip.</p> <p>The rule severity field graphically represents the ratio between the abnormal traffic and the threshold value. Every bar represents 100% of the threshold value.</p> <p>The color of the severity indicates the link's severity: 0-25% blue, 25%-50% yellow, 50%-75% orange, 75%-100% red. The link's severity is the ratio between the abnormal traffic and the overall traffic of the link (Sensor or interface).</p>
PARAMETERS VISIBLE WHEN DISPLAY IS SET TO “FULL”:	
Total Pkts	The number of packets counted since the anomaly started.
Total Bits	The number of bits counted since the anomaly started.
Overall Traffic	The percentage value between the anomaly traffic and the overall traffic.
Threshold	The threshold value.
IP Zone	The IP Zone used by Sensor. Click it to open the prefix settings from the IP Zone.
Template	The Threshold Template that contained the threshold rule, if any.
Expiration	The number of seconds that must pass for the anomaly to become inactive.
Response (Actions)	The name of the Response and the actions executed.
Comments	User comments. This field is hidden if no comments were set with the <i>Classify/Set Comment</i> action.

When a Filter is activated and detects attackers and filtering rules, a new table is displayed within the same row with the traffic anomaly. The rows of the Filter table have a red background for active filtering rules or a yellow background for inactive filtering rules.

Filter	Which Filter detected the filtering rule. Click it to open a new tab with specific data.
Filtering Rule	The filtering rule that isolates the malicious traffic. All possible filtering rules are listed in Configuration » General Settings » Mitigation Options. A white flag in the same row shows that the filtering rule conflicts with a whitelist rule.
Firewall	Indicates if the filtering rule was applied by a local software and/or hardware firewall.
Started	The date and time when the filtering rule was generated.
Latest Alarm	The last time the filtering rule matched traffic above the threshold value.
Duration	How much time has passed since the activation of the filtering rule.
Pkts/s (Peak)	The packets/second throughput for the traffic matching the filtering rule.
Bits/s (Peak)	The bits/second throughput for the traffic matching the filtering rule.
Pkts	The number of packets inside packets matched by the filtering rule.
Bits	The number of bits inside packets matched by the filtering rule.
Actions	<ul style="list-style-type: none"> • <i>Open Packet Trace</i> – available for Packet Filters when the Response contains a traffic capturing action. • <i>Open Flow Trace</i> – available for Flow Sensors with the Flow Collector feature enabled. Shows bi-directional flows that started or ended during the selected time interval. Flow Traces may have an up to 5 minute delay due to flow file buffering. Time-zone differences are not adjusted. • <i>Expire Filtering Rule</i> – force Filter to clear the filtering rule. The Filter must be running for the action to take effect.

Anomaly Archive

The tab lists all traffic anomalies sorted by time in descending order. By clicking the down arrow on any column header, you can apply row filters, change sorting direction and toggle the visibility of columns.

The <+> sign from the first column expands the anomaly for additional information, mitigation data, etc. The columns are explained in the previous section.

The <Expire Anomalies> button from the top toolbar clears all active anomalies.

Anomaly Overview

Shows trends and summarizations of traffic anomalies detected by the selected Sensors, using the selected decoders, for the selected time-frame.

Reports » Tools » BGP Operations

The **Reports » Tools** panel displays the number of BGP announcements (routing updates) that are active. The number is red when there is at least one active BGP announcement sent through a BGP Connection configured for black hole filtering, or blue when all active BGP announcements were sent through BGP Connections configured for traffic diversion.

The BGP Operations tab displays the BGP announcements sent by WanGuard, and provides a way for Console users to send BGP routing updates manually. The tab contains 3 sub-tabs, located at the lower left side of the window:

Active BGP Announcements

The tab displays all active BGP announcements sent by Sensors, Filters or Console.

Console users can send or withdraw BGP announcements manually. To send a new BGP announcement, click the **<Black Hole>** or the **<Divert Traffic>** button, enter the prefix and select a previously configured BGP Connection (see page 50) for Black Holing or Traffic Diversion.

The **<Clear All>** button deletes all announcements from the UI without updating the BGPd.

When there is at least one active BGP announcement you will see the following table:

BGP Connection	Which BGP Connection was used to send the announcement. When the grouping is set to "By BGP Connection", clicking the BGP Connection will allow you to delete all announcements that used it, with a single click.
BGP Role	Role configured for the BGP Connection. Can be <i>Unset</i> , <i>Black Hole</i> or <i>Traffic Diversion</i> .
Prefix	The prefix inside the BGP routing update. IPv4 hosts have a /32 CIDR. IPv6 hosts have a /128 CIDR. When the grouping is set to "By IP/Mask" clicking the prefix will allow you to delete all announcements for that prefix, with a single click.
From	When the BGP announcement was sent.
Until	When the BGP announcement will be withdrawn.
Anomaly	If the BGP announcement was triggered by a Response, the field contains a link to the anomaly report. To avoid announcements from overlapping, a single announcement is actually sent to BGPd when there are multiple anomalies for the same prefix and BGP Connection.
Comments	May contain user comments about the BGP announcement. If the cell contains the word "ERROR" look for BGP Connection errors in BGP Events (see page 66). If the cell contains the word "Orphan" then the anomaly that triggered the announcement is no longer active, but the announcement still is. In this case you should remove the announcement manually and investigate the cause.

Action	It contains a link for the manual removal of the BGP announcement. The column is hidden for Console guests without proper permissions.
---------------	--

BGP Announcement Archive

Lists all BGP announcements sent by Sensors, Filters or Console. By clicking the down arrow of any column header, you can apply row filters, change sorting direction and toggle the visibility of columns. All columns are explained in the previous section, except for the hidden User column that shows the Console user that sent the announcement.

You can modify the status of announcements manually by double-clicking rows. The modification affects only the UI, not the BGPd configuration.

BGP Connection Events

Lists events generated by BGP Connections for the selected time frame. Events are explained in the “Event Reporting” chapter on page 66.

Reports » Tools » Firewall Rules

The **Reports » Tools** panel displays the number of firewall rules that had matches in the last 5 seconds.

The Firewall Rules tab lists all firewall rules managed by WanGuard, and provides a quick and easy way for Console users to define their own rules. The tab contains 2 sub-tabs, located at the lower left side of the window:

Active Firewall Rules

This tab displays all firewall rules generated automatically by Filters or manually by Console users.

Administrators and operators can add or delete firewall rules manually. To add a new Firewall Rule click the **<Create Firewall Rule>** button. You will have to choose whether to apply the firewall rule using a software firewall (1st option) or using a hardware firewall (2nd option).

The **<Remove All>** button deletes all firewall rules from the UI without updating the actual firewall.

The **Create Software Firewall Rule** window provides the following options:

- **Rule Description** – A short name that will help you identify the firewall rule. This is the only mandatory field.
- **Direction** – Select *Inbound* to match packets entering your network (through interfaces defined as Inbound in the Filter Configuration window). Otherwise, select *Outbound*.
- **Filter(s)** – Select the Filters that must apply the firewall rule, according to their configuration (Interfaces, Netfilter Chain, Netfilter Table).
- **IP Protocol(s)** – Select one or more IP protocols, or *Any* to match all packets.
- **Src/Dst IP/mask** – Enter to match packets by their source or destination IP blocks. The mask is optional (defaults to /32 for IPv4 and /128 for IPv6).
- **Src/Dst Port(s)** – This field is available only for the following IP protocols: TCP, UDP, UDPLITE, DCCP and SCTP. It matches a set of source or destination ports. Up to 15 ports can be specified (e.g. 53, 1024:65535 would match ports 53 and all from 1024 through 65535).
- **IP Packet Length** – It is used to match the length of the layer-3 payload (e.g. layer-4 packet) of packets against a specific value or range of values separated by “:”.
- **IP TimeToLive** – It is used to match the time to live (TTL) field in the IP header. If the value is preceded by “>” then the traffic will be matched if TTL is greater than the given TTL value. If the value is preceded by “<” then the traffic will be matched if TTL is less than the given TTL value.
- **TCP Flags Set/Unset** – Select the TCP flags that must be explicitly set and/or unset. TCP flags not enabled in either fields are ignored by the packet matching mechanism.
- **Payload Content** – Enter to match a string anywhere in the packet. Use this match with caution as it consumes a lot of CPU resources.
- **Country(ies)** – Select to match packets by their country. This option can be used if the *xp_geoiip* Netfilter module is installed.

- **Firewall Policy** – Select the Software Firewall policy applied for the matched packets:
 - Drop – blocks packets and makes the connection appear to be to an unoccupied IP address
 - Reject – blocks packets and sends an ICMP packet indicating the port is unavailable
 - Accept – allows packets through the firewall
 - Rate Limit – allows a limited number of packets through the firewall
- **Rate Limit** – You can use this parameter to limit rate of packets / time unit to a predefined value. If the value ends with the character “b” then the rate-limiting is applied for bytes not packets.
- **Rate Limit Hashing** – You can apply the rate-limiting globally, to a single object (*Src. IP*, *Src. Port*, *Dst. IP* or *Dst. Port*) or to any combination of objects. If the rate-limiting should be connection-oriented, select all objects. To rate-limit the packet or byte rate of each source IP, select the *Src. IP* object.
- **Rule Active Until** – Select *Manually deleted* to apply the firewall rule indefinitely. Select the other options to delete the firewall rule after a predefined condition.

The **Create Chelsio Firewall Rule** window provides the following options:

- **Rule Description** – A short name that will help you identify the firewall rule. This is the only mandatory field.
- **Direction** – Select *Inbound* to match packets entering your network (through interfaces defined as Inbound in the Filter Configuration window). Otherwise, select *Outbound*.
- **Filter(s)** – Select the Filters that must apply the firewall rule, according to their configuration (Interfaces, Hardware Firewall Policy).
- **IP Protocol(s)** – Select one or more IP protocols, or *Any* to match all packets.
- **Src/Dst IP/mask** – Enter to match packets by their source or destination IP blocks. The mask is optional (defaults to /32 for IPv4 and /128 for IPv6).
- **Src/Dst Port(s)** – This field is available only for the following IP protocols: TCP, UDP, UDPLITE, DCCP and SCTP. It matches a set of source or destination ports. Up to 15 ports can be specified (e.g. 53, 1024:65535 would match ports 53 and all from 1024 through 65535).
- **Rule Active Until** – Select *Manually deleted* to apply the firewall rule indefinitely. Select the other options to delete the firewall rule after a predefined condition.

When there is at least one active firewall rule, you will see a table describing it and showing the exact number of matches.

Filtering Rules

Lists filtering rules detected by the selected Filter(s) for the selected time frame. Most fields are described in the “Reports » Tools » Anomalies” chapter on page 71.

Reports » Tools » Flow Collectors

The **Reports » Tools** panel contains a link to **Flow Collectors** if there is at least one Flow Sensor in use. The number of active Flow Collectors is displayed within the panel.

Here you can list, aggregate, filter and sort flow records, generate traffic tops and statistics.

The tab contains 2 sub-tabs, located at the left lower side of the window:

Flow Records

You can list and filter flow data. The options are:

- **Flow Sensors** – Select the interfaces you are interested in. Administrators can restrict the interfaces accessible by guest accounts.
- **Time Frame** – Select a predefined time frame, or select “Custom...” to enter a specific time interval, in order to list flows that started or ended inside the interval. Time-zone differences between the Console server and remote Flow Sensor servers are not automatically adjusted.
- **Flow Filtering Expression** – Here you can enter a filtering expression for flows. Click the light bulb icon on the right to open a window that will show you the correct syntax. Frequently-used flow filters can be saved there and reused at a later time.
- **Output** – You can select several output formats, or you can type your own format. Click the light bulb icon on the left to open a window that will show you the correct syntax.
- **Export** – If the output is not very large, it can be emailed or printed.

If you need to list huge amounts of flow data, doing it solely from within the web browser may not be a good idea. In this case, select the “Dump” option to view the CLI command used for flow listing. You can execute that CLI command from the shell and forward the output to a file.

- **Aggregation** – By default, flows are not aggregated. By clicking on the appropriate checkboxes, you can select how to aggregate flows. You can aggregate entire subnets by selecting src(dst)IPv4(IPv6)/<subnet bits>.
- **Limit Flows** – List only the first N flows of the selected time slot.
- **Sorting** – When listing flows sent by different interfaces, you can sort them according to the start time of the flows. Otherwise, flows are listed in the sequence of the selected interfaces.

Flow Tops

You can generate tops from flow data. The options are:

- **Flow Sensors** – Select the interfaces you are interested in. Administrators can restrict the interfaces accessible by guest accounts.
- **Time Frame** – Select a predefined time frame, or select “Custom...” to enter a specific time interval, in order to count only flows that started or ended inside the interval. Time-zone differences between a the

Console server and remote Flow Sensor servers are not automatically adjusted.

- **Flow Filtering Expression** – Here you can enter a filtering expression for flows. Click the light bulb icon on the right to open a window that will show you the correct syntax. Frequently-used flow filters can be saved there and reused at a later time.
- **Output** – You can select several output formats, or you can type your own format. Click the light bulb icon on the left to open a window that will show you the correct syntax.
- **Export** – If the output is not very large, it can be emailed or printed.

If you need to list huge amounts of flow data, doing so solely from within the web browser may not be a good idea. In this case, select the “Dump” option to view the CLI command used to list the top. You can execute that CLI command from the shell and forward the output to a text file.

- **Top Type** – Select the top type from the drop-down menu.
- **Aggregation** – By default, flows are not aggregated. By clicking on the appropriate checkboxes, you can select how to aggregate flows. You can aggregate entire subnets by selecting `src(dst)IPv4(IPv6)/<subnet bits>`.
- **Limit** – Limit the output to only those records whose packets or bytes match the specified condition.
- **Top** – Limit the top listing to the first N records.

Reports » Tools » Packet Tracers

The **Reports » Tools** panel contains a link to **Packet Tracers** when there is at least one Packet Sensor or Packet Filter in use. The number of active packet traces is displayed within the panel.

Here you can easily capture packets from various parts of your network using distributed Packet Sensors. You can view the contents of packets directly from Console using an integrated packet analyzer UI.

The tab contains 2 sub-tabs located at the lower left side of the window:

Active Packet Traces

Administrators, operators and guests with packet capturing privileges can generate packet dumps by clicking the **<Capture Packets>** button. The options are:

- **Description** – An optional short description to help you identify the packet trace.
- **Packet Sensor** – Select which Packet Sensors can capture the traffic you are interested in. Administrators can restrict which Packet Sensors are accessible by guest accounts.
- **BPF Expression** – Click the light bulb icon on the right to open a window that explains the Berkley Packet Filter (BPF) syntax. Frequently used BPF expressions can be saved there and reused at later time. Entering a BPF expression is mandatory. To capture all IP traffic enter “ip”.
- **Max. Running Time** – The maximum running time of the capturing thread (process).
- **Stop Capture Time** – When Max. Running Time is set to “Unlimited”, you can set the exact date when the capturing thread will stop.
- **Max. File Size (MB)** – The option is used for splitting packet dumps into multiple files of <number> Mbytes. Before writing a raw packet to a file, the Packet Sensor checks whether the file is currently larger than <number> and, if so, closes the current file and opens a new one.
- **Max. Packets** – The capture stops after receiving <number> packets.
- **Max. Files Number** – Setting this will limit the number of files created for the specified <number>, and begin overwriting files from the beginning, thus creating a “rotating” buffer. In addition, it will name the files with enough leading 0s to support the maximum number of files, allowing them to sort correctly.
- **Time Rotation (s)** – If specified, this rotates the file every <number> seconds.
- **Sampling Type & Value** – Select “None” when no packet sampling is required. Select “1 / Value” to save just one packet every <value> packets. Select “Value / 5s” to save maximum <value> packets every 5 seconds.
- **Filename Prefix** – The name of the capture file. If any file-rotation options are used, a number will be appended to the filename.
- **Snapshot (bytes/pkt)** – Snarf <number> bytes of data from each packet rather than the default of 65535 bytes. Note that taking larger snapshots both increases the amount of time it takes to process packets and, effectively, decreases the amount of packet buffering. This may cause packets to be lost. You should limit this <number> to the smallest number that will capture the protocol information you are interested in.

- **Comments** – This field may contain comments about the packet trace.

All active Packet Traces are listed as a table having the following format:

- **Description [BPF]** – The description and BPF expression of the trace.
- **Sampling** – The type of sampling being used.
- **From** – The date when the Packet Tracer started capturing packets.
- **Until** – The time or the conditions that will cause the Packet Tracer to stop capturing the traffic.
- **Status** – Indicates the status of the Packet Tracer. It is green if it's running, and red if it's not.
- **Packet Tracer** – The Packet Sensor or the Packet Filter used for capturing packets.
- **Files / Size** – The number of dump files generated and the size of the latest dump file.
- **Packets** – The number of packets captured.
- **Actions** – Click the first icon to view the latest dump file in an integrated packet analyzer tab. Click the second icon to download the latest dump file to your computer. If downloading does not work, but viewing does, increase the values of the *max_execution_time* and *memory_limit* from php.ini. Click the third icon to stop capturing packets.

Packet Trace Archive

By default, packet traces are sorted by time in descending order. By clicking the down arrow of any column header, you can apply row filters, change sorting direction and toggle the visibility of columns.

The <+> sign from the first column expands each row for additional information about the trace and provides access to packet dump files. The columns are explained in the previous section.

Reports » Components

The **Reports » Components** panel contains links to the **Overview**, **Device Group**, **Sensor** and **Filter** tabs.

The Overview tab provides a real-time view on the status of all active WanGuard components and servers. The Device Group tab provides a real-time view on the status of the Sensor(s) and Filter(s) assigned to the selected device group. The Sensor tab provides data specific to the selected Sensor. The Filter tab provides data specific to the selected Filter. Administrators can restrict which device groups, Sensors and Filters are accessible by guest accounts.

Reports » Components » Overview

The Overview tab shows self-refreshing tables that display real-time system parameters collected from all active WanGuard components and servers:

Console

The table displays the following data:

Status	A green check mark indicates that Console is functioning properly. When a red “X” is displayed, enable the WANsupervisor service on the Console server.
Online Users	Number of active Console sessions.
Avg. DB Bits/s (In/Out)	Average number of bits/s sent and received since the start of the Console database.
Avg. DB Queries/s	Average number of queries per second since the start of the Console database.
DB Clients	Number of DB clients that are currently using the Console database.
DB Connections	Number of active connections to the Console database.
DB Size	Disk space used by the Console database.
Free DB Disk	Disk space available on the partition configured to store the Console database.
Free Graphs Disk	Disk space available on the partition configured to store IP graphs.
Time Zone	Time zone of the Console server.
Console Time	Time on the Console server.
Uptime	Uptime of the Console database.

Servers

The table displays the following data for each server that runs components of WanGuard:

Status	A green check mark indicates that the server is connected to Console. When a red "X" is displayed, start the WANsupervisor service and make sure that the clocks are synchronized between the server and the Console server.
Server Name	Displays the name of the server and a colored square with the color defined in its configuration. Click to open a new tab with data specific to the server. Administrators and operators can right-click to open the Server Configuration window.
Load	Load average reported by the Linux kernel for the last 5 minutes.
Free RAM	Available RAM. Swap memory is not counted.
CPU% User	Percentage of CPU resources used by the user space processes. Can be >100% on multiple cores/CPU's (e.g. the maximum value for a quad-core system is 400%).
CPU% System	Percentage of CPU resources used by the kernel. Can be >100% on multiple cores/CPU's (e.g. the maximum value for a quad-core system is 400%).
CPU% IOWait	Percentage of CPU resources waiting for I/O operations. A high number indicates an I/O bottleneck.
CPU% Idle	Percentage of idle CPU resources. Can be >100% on multiple cores/CPU's (e.g. the maximum value for a quad-core system is 400%).
Free Flows Disk	Disk space available on the partition that is configured to store flows.
Free Dumps Disk	Disk space available on the partition that is configured to store packet dumps.
Contexts/IRQs/SoftIRQs	Number of context switches, hardware interrupts and software interrupts per second.
Uptime	Uptime of the operating system.

Sensor Clusters

The table is displayed while there is at least one active Sensor Cluster.

Status	A green check mark indicates that the Sensor Cluster is connected to Console. If you see a red "X" instead, make sure that the WANsupervisor service is running and look for errors in the event log (see page 66).
Sensor Name	Displays the name of the Sensor Cluster and a colored square with the color defined in its configuration. Click to open a new tab with data specific to the Sensor Cluster. Administrators and operators can right-click to open the Sensor Cluster configuration window.
Pkts/s (In / Out)	Inbound and outbound packets/second throughput.

Inbound Bits/s	Inbound bits/second throughput and the usage percent.
Outbound Bits/s	Outbound bits/second throughput and the usage percent.
Received Pkts/s	Number of packet/s reported by the associated Sensors.
IPs (Int.)	Number of IP addresses from to the IP Zone that send or receive traffic.
Dropped	Number of packets dropped by the Server Cluster.
CPU%	Percentage of CPUs used by the Sensor Cluster process.
RAM	Amount of memory used by the Sensor Cluster process.
Start Time	When the Sensor Cluster instance started.
Server	Which server runs the Sensor Cluster. Click to open a new tab with data specific to the server. Administrators and operators can right-click to open the Server Configuration window.

Packet Sensors

The table is displayed while there is at least one active Packet Sensor.

Status	A green check mark indicates that the Packet Sensor is connected to Console. If you see a red "X" instead, make sure that the WANsupervisor service is running and look for errors in the event log (see page 66).
Sensor Name	Displays the name of the Packet Sensor and a colored square with the color defined in its configuration. Click to open a new tab with data specific to the Packet Sensor. Administrators and operators can right-click to open the Packet Sensor Configuration window.
Pkts/s (In / Out)	Inbound and outbound packets/second throughput after IP or MAC validation.
Inbound Bits/s	Inbound bits/second throughput after IP or MAC validation and the usage percent.
Outbound Bits/s	Outbound bits/second throughput after IP or MAC validation and the usage percent.
Received Pkts/s	Rate of sniffed packets before IP or MAC validation.
IPs (Int / Ext)	Number of IP addresses that sent or received traffic. The Int(ernal)/Ext(ernal) IPs are the IPs inside/outside the IP Zone. The Top Generator parameter from the Sensor configuration enables or disables the monitoring of external IPs.
Dropped	Number of packets dropped by the packet capturing engine. A high number indicates a sniffing performance problem.
CPU%	Percentage of CPUs used by the Packet Sensor process.
RAM	Amount of memory used by the Packet Sensor process.
Start Time	When the Packet Sensor started.

Server	Which server runs the Packet Sensor. Click to open a new tab with data specific to the server. Administrators and operators can right-click to open the Server Configuration window.

Flow Sensors

The table is displayed while there is at least one active Flow Sensor.

Status	A green check mark indicates that the Flow Sensor is connected to Console. If you see a red “X” instead, make sure that the WANsupervisor service is running and look for errors in the event log (see page 66).
Sensor Name	Displays the name of the Flow Sensor. Click to open a new tab with data specific to the Flow Sensor. Administrators and operators can right-click to open the Flow Sensor Configuration window.
Interface	The interface name and a colored square with the configured graph color. If the interface names are missing for more than 5 minutes after the Flow Sensor has started, check the troubleshooting guide from page 44.
Pkts/s (In / Out)	Inbound and outbound packets/second throughput after IP or AS validation.
Inbound Bits/s	Inbound bits/second throughput after IP or AS validation and usage percent.
Outbound Bits/s	Outbound bits/second throughput after IP or AS validation and usage percent.
IPs (Int / Ext)	Number of IP addresses that send or receive traffic. The Int(ernal)/Ext(ernal) IPs are the IPs inside/outside the IP Zone. The Top Generator parameter from the Sensor configuration enables or disables the monitoring of external IPs.
Flows/s	Number of flows per second received by the Flow Sensor.
Flows Delay	Because traffic data must be aggregated first, flow devices export flows with a configured delay. Some devices export flows much later than the configured delay and this field contains the maximum flows delay detected by the Flow Sensor. Flow Sensor cannot run with flow delays of over 5 minutes.
Dropped	Number of unaccounted flows. A high number indicates a performance problem of the Flow Sensor or a network connectivity issue with the flow exporter.
CPU%	Percentage of CPU resources used by the Flow Sensor process.
RAM	Amount of RAM used by the Flow Sensor process.
Start Time	When the Flow Sensor started.
Server	Which server runs the Flow Sensor. Click it to open a new tab with data specific to the server. Administrators and operators can right-click to open the Server Configuration window.

SNMP Sensors

The table is displayed while there is at least one active SNMP Sensor.

Status	A green check mark indicates that the SNMP Sensor is connected to Console. If you see a red "X" instead, make sure that the WANsupervisor service is running and look for errors in the event log (see page 66).
Sensor Name	Displays the name of the SNMP Sensor and a colored square with the color defined in its configuration. Click to open a new tab with data specific to the SNMP Sensor. Administrators and operators can right-click to open the SNMP Sensor Configuration window.
Interface	Interface name and a colored square with the configured graph color.
Pkts/s (In / Out)	Inbound and outbound packets/second throughput.
Inbound Bits/s	Inbound bits/second throughput and usage percent.
Outbound Bits/s	Outbound bits/second throughput and usage percent.
Errors/s (In / Out)	For packet-oriented interfaces, it represents the number of inbound and outbound packets that contained errors, preventing them from being deliverable to a higher-layer protocol. For character-oriented or fixed-length interfaces, it represents the number of inbound transmission units that contained errors preventing them from being deliverable to a higher-layer protocol.
Discards/s (In / Out)	Number of inbound and outbound packets which were chosen to be discarded even though no errors had been detected to prevent them from being deliverable to a higher-layer protocol. One possible reason for discarding such a packet could be to free up buffer space.
Oper. Status	Current operational state of the interface. The <i>Testing</i> state indicates that no operational packets can be passed. If Administrative Status is <i>Down</i> then Operational Status should be <i>Down</i> . If Administrative Status is changed to <i>Up</i> then Operational Status should change to <i>Up</i> if the interface is ready to transmit and receive network traffic; it should change to <i>Dormant</i> if the interface is waiting for external actions (such as a serial line waiting for an incoming connection); it should remain in the <i>Down</i> state if and only if there is a fault that prevents it from going to the <i>Up</i> state; it should remain in the <i>NotPresent</i> state if the interface has missing (typically, hardware) components.
Admin. Status	Desired state of the interface. The <i>Testing</i> state indicates that no operational packets can be passed. When a managed system initializes, all interfaces start with the Administrative Status in the <i>Down</i> state. As a result of either explicit management action or per configuration information retained by the managed system, the Administrative Status is then changed to either the <i>Up</i> or <i>Testing</i> states (or remains in the <i>Down</i> state).
CPU%	Percentage of CPU resources used by the SNMP Sensor process.
RAM	Amount of RAM used by the SNMP Sensor process.
Start Time	When the SNMP Sensor started.
Server	Which server runs the SNMP Sensor. Click it to open a new tab with data specific to the server. Administrators and operators can right-click to open the Server Configuration window.

Filter Clusters, Packet Filters and Flow Filters

The tables are displayed while there is at least one active Filter Cluster, Packet Filter or Flow Filter.

Status	A green check mark indicates that the Filter is connected to Console. If you see a red “X” instead, make sure that the WANsupervisor service is running and look for errors in the event log (see page 66).
Filter Name	Displays the Filter and a colored square with the color defined in its configuration. Click to open a new tab with data specific to the Filter. Administrators and operators can right-click to open the Filter Configuration window.
Anomaly#	When a Filter instance is activated by a Response to mitigate an anomaly, the field contains the link to the anomaly report. Otherwise, the field contains the message “Filter offline”.
Prefix	IP address/mask of your network originating or being the target of the traffic anomaly. Click to open a tab with data specific to the IP block or address.
IP Group	IP group of the prefix. Click to open a tab with data specific to the IP group.
Decoder	Decoder used for detecting the abnormal traffic.
Pkts/s	Packets/second throughput sent to the attacked prefix.
Bits/s	Bits/second throughput sent to the attacked prefix.
IPs (Ext.)	Number of unique IP addresses sending traffic to the attacked prefix.
Dropped	Rate of packets dropped by the packet capturing engine. A high number indicates a sniffing performance problem.
Peak CPU%	Maximum percentage of CPU resources used by the Filter instance.
Peak RAM	Maximum amount of RAM used by the Filter instance.
Start Time	When the Filter instance started mitigating the anomaly.
Server	Which server runs the Filter instance. Click to open a new tab with data specific to the server. Administrators and operators can right-click to open the Server Configuration window.

Reports » Components » Sensors

Click on a Sensor name anywhere in Console to open a tab that contains specific information. The tab includes a few sub-tabs, located at the lower side of the window. All sub-tabs share the following common toolbar fields:

- **Sensors** – Select the Sensors you are interested in, or select “All” to select all Sensors. Administrators can restrict which Sensors are accessible by guest accounts.
- **Time Frame** – Select a predefined time frame, or select “Custom...” to enter a specific time interval.

Sensor Dashboard

The Sensor dashboard allows you to group the most relevant data collected by Sensors. The Sensor dashboard configuration does not apply to a particular Sensor, so the changes you make will be visible for other Sensor dashboards as well. The operation of dashboards is described in the “Reports » Dashboards” chapter on page 94.

The configuration of Sensor widgets is described in the following paragraphs.

Sensor Graphs

This sub-tab allows you to view a variety of Sensor-related histograms for the selected Sensor(s):

- **Data Units** – Select one or more data units:
 - *Most Used* – Frequently-used data units.
 - *Packets* – Inbound packets/second (+ on Y-axis) and outbound packets/second (- on Y-axis).
 - *Bits* – Inbound bits/second (+ on Y-axis) and outbound bits/second (- on Y-axis).
 - *Applications* – Sensor can collect application-specific distribution data for: HTTP, HTTPS, SMTP, POP3, IMAP, SNMP, FTP, SSH, TELNET, SQL, NETBIOS, MS-DS, MS-RDP, DNS, ICMP and OTHERS. The graphs are updated when the Sensor configuration has the Top Generator parameter set to “Basic”.
 - *Bytes* – Bytes/second throughput.
 - *Internal or External IPs* – Number of IP addresses that send or receive traffic. Internal and External IPs are hosts inside and respectively outside the IP Zone. The Top Generator parameter from the Sensor configuration enables or disables monitoring of External IPs. A spike in the Internal IPs graph usually means that an IP class scan was performed against your IP blocks. A spike in the External IPs graph usually means that you have received a spoofed attack.
 - *Received Frames* – For Packet Sensors, it represents the number of packets/s received before IP or MAC validation. For Flow Sensors, it represents the number of flows/s received before IP or AS validation.
 - *Dropped Frames* – For Packet Sensors, it represents the number of packets dropped by the packet capturing engine. A high number indicates a sniffing performance problem. For Flow Sensors, it

represents the number of unaccounted flows. A high number indicates a wrong configuration of the Flow Sensor or a network connectivity issue with the flow exporter.

- *Unknown Frames* – For Packet Sensors, it represents the rate of packets not passing IP validation. For Flow Sensors, it represents the rate of invalidated flows.
- *Unknown Sources* – Number of source IP addresses that did not pass IP validation.
- *Unknown Destinations* – Number of destination IP addresses that did not pass IP validation.
- *Avg. Packet Size* – Average packet size in bits/packet.
- *CPU%* – Percentage of CPU resources used by the Sensor process.
- *RAM* – Amount of RAM used by the Sensor process.
- *Load* – Load reported by the Linux kernel.
- *IP Graphs* – Number of updated IP graphs files.
- *IP Accounting* – Number of updated IP accounting records.
- *HW Graphs* – Number of updated traffic profiling files.
- *IP Graphs Time* – Number of seconds needed to update the IP graphs files.
- *HW Graphs Time* – Number of seconds needed to update the traffic profiling files.
- *Processing Time* – Number of seconds needed to perform traffic analysis functions.
- *IP Structures* – Number of internal IP structures.
- *IP Structure RAM* – Number of RAM bytes used by each IP structure.
- **Graphs Size** – Select a predefined dimension or enter a custom one in the “<X> x <Y>” format, where <X> and <Y> are the X-axis and Y-axis pixels.
- **Graphs Title** – Graphs can have an automatically-generated title for the “Auto” option, no title for the “None” option, or you can enter your own text to be rendered as a title.
- **Graph Legend** – Select the level of detail for the graph legend.
- **Consolidation** – If you are interested in spikes, select the *MAXIMUM* aggregation type. If you are interested in average values, select the *AVERAGE* aggregation type. If you are interested in low values, select the *MINIMUM* aggregation type.
- **Graph Options**
 - *Stack Sensors* – Select to generate a single stacked graph for all selected Sensors.
 - *Show Totals* – Select to show the summed up values from stacked Sensors.

Sensor Tops

This sub-tab allows you to generate various traffic tops for the selected Sensor(s). The Top Generator parameter from the Sensor configuration enables or disables data collection for various Sensor tops.

- **Top Type** – Select a top type:
 - *Talkers* – Hosts from your network that send or receive the most traffic for the selected decoder.

Available only when the Top Generator parameter from the Sensor configuration is set to “Basic”.

- *IP Groups* – IP groups that send or receive the most traffic for the selected decoder. Available only when the Top Generator parameter from the Sensor configuration is set to “Basic”.
- *External IPs* – External IPs that send or receive the most traffic for the selected decoder. Available when the Top Generator parameter from the Sensor configuration is set to “Extended” or “Full”.
- *Autonomous Systems* – Autonomous systems that send or receive the most traffic. Available only when the Top Generator parameter from the Sensor configuration is set to “Extended” or “Full”.
- *Countries* – Countries that send or receive the most traffic. Available when the Top Generator parameter from the Sensor configuration is set to “Extended” or “Full”.
- *TCP Ports* – Most-used TCP ports. Available when the Top Generator parameter from the Sensor configuration is set to “Basic”.
- *UDP Ports* – Most-used UDP ports. Available when the Top Generator parameter from the Sensor configuration is set to “Basic”.
- *IP Protocols* – Most-used IP protocols. Available when the Top Generator parameter from the Sensor configuration is set to “Basic”.
- *IP Versions* – Most-used IP versions: IPv4 or IPv6. Available when the Top Generator parameter from the Sensor configuration is set to “Basic”.
- **Decoder** – Select the decoder that analyzes the type of traffic that interests you.
- **Direction** – Direction of traffic, *Inbound* or *Outbound*.
- **Group Sensors** – When unchecked, each Sensor generates a different top. When checked, all selected Sensors generate a single top with combined data.
- **DNS** – When checked, it enables reverse DNS resolution for IP addresses. It may slow down generating tops for *Talkers* and *External IPs*.

You can increase the number of top records and change the available decoders in Configuration » General Settings » Graphs & Storage, see page 20.

Generating tops for many Sensors and long time frames may take minutes. If the report page timeouts, increase the *max_execution_time* parameter from *php.ini*.

Flow Records

You can list and filter the flow data collected by the selected Flow Sensors. The options are described in the “Flow Collectors” chapter on page 78. This sub-tab is visible only for tabs opened for Flow Sensors.

Flow Tops

You can generate tops from the flow data collected by the selected Flow Sensors. The options are described in the “Flow Collectors” chapter on page 78. This sub-tab is visible only for tabs opened for Flow Sensors.

AS Graphs

Flow Sensors and Packet Sensors can generate traffic and bandwidth histograms for autonomous systems. This feature is enabled for Packet Sensors that have the Top Generator parameter set to “Full”, and for Flow Sensors that have the Top Generator parameter set to “Full” or “Extended”.

- **AS Numbers** – Click the lightbulb icon on the right to open a window containing the correct syntax. Frequently-searched AS numbers can be saved there, and used at a later time. To see the list of AS numbers owned by a particular organization, go to Help » IP & AS Information » AS Numbers List.
- **Graphs Size** – Select a predefined dimension or enter a custom one in the “<X> x <Y>” format, where <X> and <Y> are the X-axis and Y-axis pixels.
- **Graphs Title** – Graphs can have an automatically-generated title for the “Auto” option or no title for the “None” option, or you can enter your own text to be rendered as a title.
- **Graph Options**
 - *Stack Sensors* – Select to generate a single stacked AS graph for all selected Sensors.
 - *Stack ASNs* – Select to show a single graph for multiple AS numbers.

Country Graphs

Flow Sensors and Packet Sensors can generate traffic and bandwidth histograms for countries. This feature is enabled for Sensors that have the Top Generator parameter set to “Full” or “Extended”.

- **Countries** – Select the country or countries from the drop-down list, or click the light bulb icon on the right to open a window with saved selections for continents and world regions.
- **Graphs Size** – Select a predefined dimension or enter a custom one in the “<X> x <Y>” format, where <X> and <Y> are the X-axis and Y-axis pixels.
- **Graphs Title** – Graphs can have an automatically-generated title for the “Auto” option or no title for the “None” option, or you can enter your own text to be rendered as a title.
- **Graph Options**
 - *Stack Sensors* – Select to generate a single stacked country graph for all selected Sensors.
 - *Stack Countries* – Select to show a single graph for multiple countries.

Sensor Events

This sub-tab lists events generated by the selected Sensor(s) for the selected time frame. The events are described in the “Event Reporting” chapter on page 66.

Anomaly Overview

This sub-tab displays trends and summarizations of traffic anomalies detected by the selected Sensor(s).

Reports » Components » Filters

Click on a Filter name anywhere in Console to open a tab that contains specific information. The tab includes few sub-tabs, located at the lower side of the window. All sub-tabs share the following common toolbar fields:

- **Filters** – Select the Filters you are interested in, or select “All” to select all Filters. Administrators can restrict which Filters are accessible by guest accounts.
- **Time Frame** – Select a predefined time frame, or select “Custom...” to enter a specific time interval.

Filter Dashboard

The Filter dashboard allows you to group the most relevant data provided by Filters. The configuration of Filter dashboard does not apply to a particular Filter, and the changes you make will be visible for other Filter dashboards as well. The operation of dashboards is described in the “Reports » Dashboards” chapter on page 94.

The configuration of Filter widgets is described in the following paragraphs.

Filter Graphs

This sub-tab allows you to view a variety of Filter-related histograms for the selected Filter(s):

- **Data Units** – Select one or more data units:
 - *Most Used* – Frequently-used data units.
 - *Anomalies* – Number of anomalies mitigated by the selected Filter(s).
 - *Filtering Rules* – Number of filtering rules detected by the selected Filter(s).
 - *SW Firewall Rules* – Number of filtering rules enforced using the software filter framework.
 - *HW Firewall Rules* – Number of filtering rules enforced using the hardware filter framework.
 - *Source IPs* – Number of unique IP addresses that have sent traffic to the attacked destination(s).
 - *CPU%* – Maximum percentage of CPU resources used by the selected Filter(s).
 - *Used RAM* – Amount of RAM used by the selected Filter(s).
 - *Filtered Packets* – How many packets were filtered by the Software Firewall.
 - *Filtered Bits* – How many bits were filtered by the Software Firewall.
 - *Dropped Packets* – Rate of packets dropped by the packet capturing engine of the selected Filter(s).
 - *Received Packets* – Rate of packets received by the selected Filter(s).
 - *Packets/s* – Rate of packets analyzed by the selected Filter(s).
 - *Bits/s* – Rate of bits/s analyzed by the selected Filter(s).
 - *Filtering Rules* – Number of filtering rules for each filtering rule type.

- *Total Excepted Rules* – Number of whitelisted filtering rules.
- **Graphs Size** – Select a predefined dimension or enter a custom one in the “<X> x <Y>” format, where <X> and <Y> are the X-axis and Y-axis pixels.
- **Graphs Title** – Graphs can have an automatically-generated title for the “Auto” option or no title for the “None” option, or you can enter your own text to be rendered as a title.
- **Graph Legend** – Select the level of detail for the graph legend.
- **Consolidation** – If you are interested in spikes, select the *MAXIMUM* aggregation type. If you are interested in average values, select the *AVERAGE* aggregation type. If you are interested in low values, select the *MINIMUM* aggregation type.
- **Graph Options**
 - *Stack Filters* – Select to generate a single stacked graph for all selected Filters.
 - *Show Totals* – Select to show the summed up values from stacked Filters.

Filter Events

Lists events generated by the selected Filter(s) for the selected time frame. The events are described in the “Event Reporting chapter” on page 66.

Filtering Rules

Lists filtering rules detected by the selected Filter(s) for the selected time frame. Most fields are described in the “Reports » Tools » Anomalies” chapter on page 71.

Filter Instances

Lists statistics collected by each Filter instance.

Reports » Dashboards

Wouldn't it be nice to see all the relevant data in a single tab? **Dashboards** allow you to group data from any report according to your needs.

Any dashboard can be configured to refresh itself on intervals ranging from 5 seconds to 15 minutes.

A few sample dashboards are included by default. If you are a Console administrator or operator you can **create** and configure your own dashboards by clicking Reports » Dashboards » <+> » Dashboard. Guest accounts are not allowed to add or make modifications to dashboards.

In the dashboard **configuration**, you can edit the name of the dashboard, set permissions, layout, or choose to override the time frame of widgets with the time frame of the dashboard.

The dashboard contains **widgets**. To sort them, click the title bar and move them around. To collapse a widget, click the first icon on the widget title bar. To configure a widget, click the second icon from its title bar. To delete a widget, click the third icon from its title bar.

Along with few specific fields, every widget has a configurable title and height. Leave the widget's height parameter set to "Auto" for the widget to take all the vertical space it needs. To restrict the height of a widget, enter a number of pixels instead.

Widget options are self-explanatory or described in other chapters.

Reports » IP Addresses & Groups

This chapter describes how to generate detailed traffic reports for any IP address, block or group included in Configuration » Network & Policy » IP Zones. Traffic graphs are available only for IP addresses, blocks or groups that have the IP graphing parameter set to “Yes”. Traffic accounting data is available only when the IP accounting parameter is set to “Yes”.

Reports » IP Addresses panel allows you to quickly generate traffic reports for IP addresses and blocks, either entered in the upper side of the panel, or selected from the expandable tree below.

Reports » IP Groups panel lists all IP groups defined in IP Zones. Select an IP group to generate a traffic report for all IP blocks belonging to it. To search for a specific IP group, enter a sub-string contained in its name in the upper side of the panel.

The traffic report tab includes few sub-tabs located at the lower side of the window. All sub-tabs share the following common toolbar fields:

- **Sensor** – Select the Sensors you are interested in or select “All” to select all Sensors. Administrators can restrict the Sensors accessible by guest accounts.
- **Time Frame** – Select a predefined time frame, or select “Custom...” to enter a specific time interval.

IP Dashboard

The IP dashboard allows you to group the most relevant data collected by the selected Sensors for the selected IP address, block or group. The configuration of IP dashboard does not apply to a particular IP address, block or group, and the changes you make will be visible for other IP dashboards as well. The operation of dashboards is described in the “Reports » Dashboards” chapter on page 94.

The configuration of Decoder Graph widget and IP Accounting widget is described by the following paragraphs.

IP Graphs

Allows you to view traffic histograms generated for the selected IP block, host or group:

- **Decoders & Data Unit** – Select the decoders and data unit you are interested in. Available data units: *Packets, Bits and Bytes*.
- **Graphs Size** – Select a predefined dimension or enter a custom one in the “<X> x <Y>” format, where <X> and <Y> are the X-axis and Y-axis pixels.
- **Graph Title** – Graphs can have an automatically-generated title for the “Auto” option or no title for the “None” option, or you can enter your own text to be rendered as a title.
- **Graph Legend** – Select the detail of the graph legend.
- **Consolidation** – If you are interested in spikes, select the *MAXIMUM* aggregation type. If you are

interested in average values, select the *AVERAGE* aggregation type. If you are interested in low values, select the *MINIMUM* aggregation type.

- **Graphs Stacking**

- *Stack Sensors* – Generates a single stacked graph for all selected Sensors.
- *Stack Decoders* – Generates a single stacked graph for all selected decoders.
- *Stack IPs* – Uncheck this option if you want a different traffic graph displayed for every IP address contained in the selected IP block or IP group.
- *Conflicting Decoders* – If decoders can be included one within the other (e.g. TOTAL contains TCP that contains HTTP and HTTPS), the graph will display stacked decoders to show the most specific ones. This generates both accurate and intuitive traffic graphs. In the example above, TOTAL will be displayed as TOTAL OTHER and TCP as TCP OTHER. However, when you select TCP, HTTP and TCP+SYN as decoders, the TCP+SYN decoder can be included in both TCP and HTTP, thus generating a decoder conflict. Check this option to stop detection of conflicting decoders, in order to generate intuitive but potentially inaccurate traffic graphs.
- *Sum IPs* – Creates a subnet graph by aggregating IP graphs generated for every IP address contained in the subnet. This option will increase the load of the server.

The number of decoders, data units and aggregation types can be modified in Configuration » General Settings » Graphs & Storage (see page 20).

IP Accounting

Allows you to generate traffic accounting reports for the selected IP block, host or group:

- **Decoders & Data Unit** – Select the decoders and data unit that you are interested in. Available data units: *Packets*, *Bits* and *Bytes*.
- **Report Type** – Select the interval used to aggregate the accounting data: *Daily*, *Weekly*, *Monthly*, *Yearly*. The maximum accuracy of traffic accounting reports is 1 day, therefore when you select a shorter time frame you will still see the accounting data collected for the whole day.
- **Sum IPs** – Uncheck this option if you want a different traffic accounting report displayed for each IP address contained in the selected IP block or group.
- **Sum Sensors** – Generates a single traffic accounting report for multiple Sensors.

The number of decoders can be modified in Configuration » General Settings » Graphs & Storage (see page 20).

Flow Records

The sub-tab is visible only when there is at least one Flow Sensor in use.

You can list and filter the flow data collected by the selected Flow Sensors, for the selected IP block, host or group. The options are described in the “Flow Collectors” chapter on page 78.

Flow Tops

This sub-tab is visible only when there is at least one Flow Sensor in use.

You can generate tops from the flow data collected by Flow Sensors, for the selected IP block, host or group. The options are described in the “Flow Collectors” chapter on page 78.

Profile Graphs

This sub-tab allows you to view traffic profiling graphs generated for the selected IP block or host.

Traffic profiling can be globally disabled from Configuration » General Settings » Anomaly Detection (see page 23). Sensor generates traffic profiling graphs only for IP blocks or hosts that have the Profiling Data parameter in the IP Zone set to “For Subnet”, “For IPs” or “For All”.

Anomaly Overview

This sub-tab generates a report with trends and summarizations of traffic anomalies sent or received by the selected IP address, block or group.

Reports » Servers

Click on a server name anywhere in Console to open a tab containing specific information. The server tab includes a few sub-tabs, located at the lower side of the window. All sub-tabs share the following common toolbar fields:

- **Servers** – Select the servers you are interested in, or select “All” to select all servers. Administrators can restrict the servers accessible by guest accounts.
- **Time Frame** – Select a predefined time frame, or select “Custom...” to enter a specific time interval.

Console / Server Dashboard

Allows you to group the most relevant data collected for a server. The configuration for the server dashboard does not apply to a particular server, and the changes you make will be visible for other server dashboards as well. The operation of dashboards is described in the “Reports » Dashboards” chapter on page 94.

The configuration of Server and Console widgets is described by the following paragraphs.

Console / Server Graphs

Server Graphs allows you to generate various histograms for the selected server(s):

- **Data Units** – Select one or more data units:
 - *Most Used* – Frequently-used data units.
 - *System Load* – Load reported by the Linux kernel.
 - *Free RAM* – Available RAM. The swap memory is not counted.
 - *Database/Graphs/SSD/Flow Collector/Package Dumps Disk - Free space* – How much disk space is available for each file-system path.
 - *Uptime* – Uptime of the operating system.
 - *CPU% system/userspace/niced/idle* – Percentages of CPU resources used by the system, userspace processes, processes running with increased (nice) priority, and idle loop.
 - *Number of processes* – Total number of processes that are running.
 - *Hardware/Software CPU Interrupts* – Number of CPU interrupts made by hardware and software events.
 - *Context Switches* – Indicates how much time the system spends on multi-tasking.
 - *Running Components* – Number of Sensors and Filter instances.
 - *Clock Delta* – Difference of time between the selected server and the Console server, in seconds. If the value is not zero run ntpd to keep the clock synchronized on all servers.

- *Database/Graphs/SSD/Packet Dumps/Flow Collector Disk - Total* – How much disk space is allocated for the partitions that store the paths.
- *Database/Graphs/SSD/Packet Dumps/Flow Collector Disk - Free Inodes* – Number of free inodes held by the partitions that store the paths.
- *Database/Graphs/SSD/Packet Dumps/Flow Collector Disk - Ops/s* – Number of reads and writes for the partitions that store the paths.
- *Database/Graphs/SSD/Packet Dumps/Flow Collector Disk - Bytes/s* – Number of bytes/s for the partitions that store the paths.
- *Server Interface(s) - Packets/Bits/Errors/Dropped* – Interface statistics collected for the network interfaces defined in the Configuration » Servers.
- **Graphs Size** – Select a predefined dimension or enter a custom one in the “<X> x <Y>” format, where <X> and <Y> are the X-axis and Y-axis pixels.
- **Graphs Title** – Graphs can have an automatically-generated title for the “Auto” option or title for the “None” option, or you can enter your own text to be rendered as a title.
- **Graph Legend** – Select the level of detail for the graph legend.
- **Consolidation** – If you are interested in spikes, select the *MAXIMUM* aggregation type. If you are interested in average values, select the *AVERAGE* aggregation type. If you are interested in low values, select the *MINIMUM* aggregation type.
- **Graph Options**
 - *Stack Servers & Interfaces* – Generate a single stacked graph for all selected servers and server interfaces.
 - *Show Totals* – Shows the summed up values from stacked servers.

Server Events

Lists events generated by the selected server(s). The events are described in the “Event Reporting” chapter on page 66.

Console Events

The sub-tab is visible only when opening the Console tab. It lists events generated by Console. Events are described in the “Event Reporting” chapter on page 66.

Server Commands

Console administrators can execute CLI commands on the selected server(s) and see the output in this sub-tab. The commands are executed by the WANsupervisor service with normal user (non-root) privileges. To prevent the execution of CLI commands through Console, start the WANsupervisor service with the “-n” option.

Appendix 1 – IPv4 Subnet CIDR Notation

WanGuard uses extensively IP addresses and IP classes with the CIDR notation. To view details about any IPv4 subnet click [Help](#) → [Subnet Calculator](#).

CIDR	CLASS	HOSTS NO.	MASK
/32	1/256 C	1	255.255.255.255
/31	1/128 C	2	255.255.255.254
/30	1/64 C	4	255.255.255.252
/29	1/32 C	8	255.255.255.248
/28	1/16 C	16	255.255.255.240
/27	1/8 C	32	255.255.255.224
/26	1/4 C	64	255.255.255.192
/25	1/2 C	128	255.255.255.128
/24	1 C	256	255.255.255.000
/23	2 C	512	255.255.254.000
/22	4 C	1024	255.255.252.000
/21	8 C	2048	255.255.248.000
/20	16 C	4096	255.255.240.000
/19	32 C	8192	255.255.224.000
/18	64 C	16384	255.255.192.000
/17	128 C	32768	255.255.128.000
/16	256 C, 1 B	65536	255.255.000.000
/15	512 C, 2 B	131072	255.254.000.000
/14	1024 C, 4 B	262144	255.252.000.000
/13	2048 C, 8 B	524288	255.248.000.000
/12	4096 C, 16 B	1048576	255.240.000.000
/11	8192 C, 32 B	2097152	255.224.000.000
/10	16384 C, 64 B	4194304	255.192.000.000
/9	32768 C, 128B	8388608	255.128.000.000
/8	65536 C, 256B, 1 A	16777216	255.000.000.000
/7	131072 C, 512B, 2 A	33554432	254.000.000.000
/6	262144 C, 1024 B, 4 A	67108864	252.000.000.000
/5	524288 C, 2048 B, 8 A	134217728	248.000.000.000
/4	1048576 C, 4096 B, 16 A	268435456	240.000.000.000
/3	2097152 C, 8192 B, 32 A	536870912	224.000.000.000
/2	4194304 C, 16384 B, 64 A	1073741824	192.000.000.000
/1	8388608 C, 32768 B, 128 A	2147483648	128.000.000.000
/0	16777216 C, 65536 B, 256 A	4294967296	000.000.000.000

Appendix 2 – Configuring NetFlow Data Export

This appendix is a brief guide to setting up the NetFlow data export (NDE) on Cisco and Juniper routers or intelligent Cisco Layer 2/Layer 3/Layer 4 switches. If you have problems with the configuration contact your network administrator or consultant. For devices that run hybrid mode on a Supervisor Engine (Catalyst 65xx series) it is recommended to configure IOS NDE on the MSFC card and CatOS NDE on the Supervisor Engine. For more information about setting up NetFlow on Cisco please visit <http://www.cisco.com/go/netflow>.

Configuring NDE on an IOS Device

In the configuration mode on the router or MSFC, issue the following to start NetFlow Export.

First enable Cisco Express Forwarding:

```
router(config)# ip cef
router(config)# ip cef distributed
```

And turn on flow accounting for each input interface with the interface command:

```
interface
ip route-cache flow
```

For example:

```
interface FastEthernet0
ip route-cache flow
interface Serial2/1
ip route-cache flow
```

It is necessary to enable NetFlow on all interfaces through which traffic (you are interested in) will flow. Now, verify that the router (or switch) is generating flow stats – try command 'show ip cache flow'. Note that for routers with distributed switching (GSR's, 75XX's) the RP cli will only show flows that made it up to the RP. To see flows on the individual line cards use the 'attach' or 'if-con' command and issue the 'sh ip ca fl' on each LC.

Enable the exports of these flows with the global commands:

```
router(config)# ip flow-export version 5
router(config)# ip flow-export destination <ip_address> 2000
router(config)# ip flow-export source FastEthernet0
```

Use the IP address of the server running the Flow Sensor and the configured listening port. UDP port 2000 is used as an example. The 'ip flow-export source' command is used to set up the source IP address of the exports sent by the equipment.

If your router uses the BGP protocol, you can configure AS to be included in exports with command:

```
router(config)# ip flow-export version 5 [peer-as | origin-as]
```

The following commands break up flows into shorter segments: 1 minute for active traffic and 30 seconds for inactive traffic. Flow Sensor drops flows older than 5 minutes!

```
router(config)# ip flow-cache timeout active 1
router(config)# ip flow-cache timeout inactive 30
```

In enable mode you can see current NetFlow configuration and state.

```
router# show ip flow export
router# show ip cache flow
router# show ip cache verbose flow
```

Configuring NDE on a CatOS Device

In privileged mode on the Supervisor Engine enable NDE:

```
switch> (enable) set mls nde <ip_address> 2000
```

Use the IP address of the server running the Flow Sensor and the configured listening port. UDP port 2000 is used only as an example.

```
switch> (enable) set mls nde version 5
```

The following command is required to set up flow mask to full flows.

```
switch> (enable) set mls flow full
```

The following commands break up flows into shorter segments: ~1 minute for active flows and ~ 30 seconds for inactive flows. Flow Sensor drops flows older than 5 minutes!

```
switch> (enable) set mls agingtime long 8
switch> (enable) set mls agingtime 4
```

If you want to account all traffic within the specified VLANs rather than inter VLAN traffic use CatOS 7.2 or higher and issue the following command:

```
switch> (enable) set mls bridged-flow-statistics enable
```

And enable NDE:

```
switch> (enable) set mls nde enable
```

To see current NetFlow configuration and state issue the following commands:

```
switch> (enable) show mls nde
switch> (enable) show mls debug
```

Configuring NDE on a Native IOS Device

To configure NDE use the same commands as for the IOS device. In the enable mode on the Supervisor Engine, issue the following to set up the NetFlow export version 5.

```
switch(config)# mls nde sender version 5
```

The following commands break up flows into shorter segments: ~1 minute for active flows and ~ 30 seconds for inactive flows. Flow Sensor drops flows older than 5 minutes!

```
switch(config)# mls aging long 8
switch(config)# mls aging normal 4
```

On the Supervisor Engine 1 issue the following to put full flows into the NetFlow exports:

```
switch(config)# mls flow ip full
```

If you have a Supervisor Engine 2 or 720 running IOS version 12.1.13(E) or higher, issue the following commands instead:

```
switch(config)# mls flow ip interface-full
switch(config)# mls nde interface
```

Configuring NDE on a 4000 Series Switch

Configure the switch the same as an IOS device, but instead of command 'ip route cache flow' use command 'ip route-cache flow infer-fields'. This series requires a Supervisor IV with a NetFlow Services daughter card to support NDE.

Configuring NDE on a Juniper Router (non-MX)

Juniper supports flow exports by the routing engine sampling packet headers and aggregating them into flows. Packet sampling is done by defining a firewall filter to accept and sample all traffic, applying that rule to the interface and then configuring the sampling forwarding option.

```
interfaces {
    ge-0/1/0 {
        unit 0 {
            family inet {
                filter {
                    input all;
                    output all;
                }
                address 192.168.1.1/24;
            }
        }
    }
}
firewall {
    filter all {
        term all {
            then {
                sample;
                accept;
            }
        }
    }
}
```

```
forwarding-options {  
  sampling {  
    input {  
      family inet {  
        rate 100;  
      }  
    }  
    output {  
      cflowd 192.168.1.100 {  
        port 2000;  
        version 5;  
      }  
    }  
  }  
}
```


Appendix 3 – BGP Black Hole Guideline for WanGuard Sensor

Understanding of RTBH using WanGuard

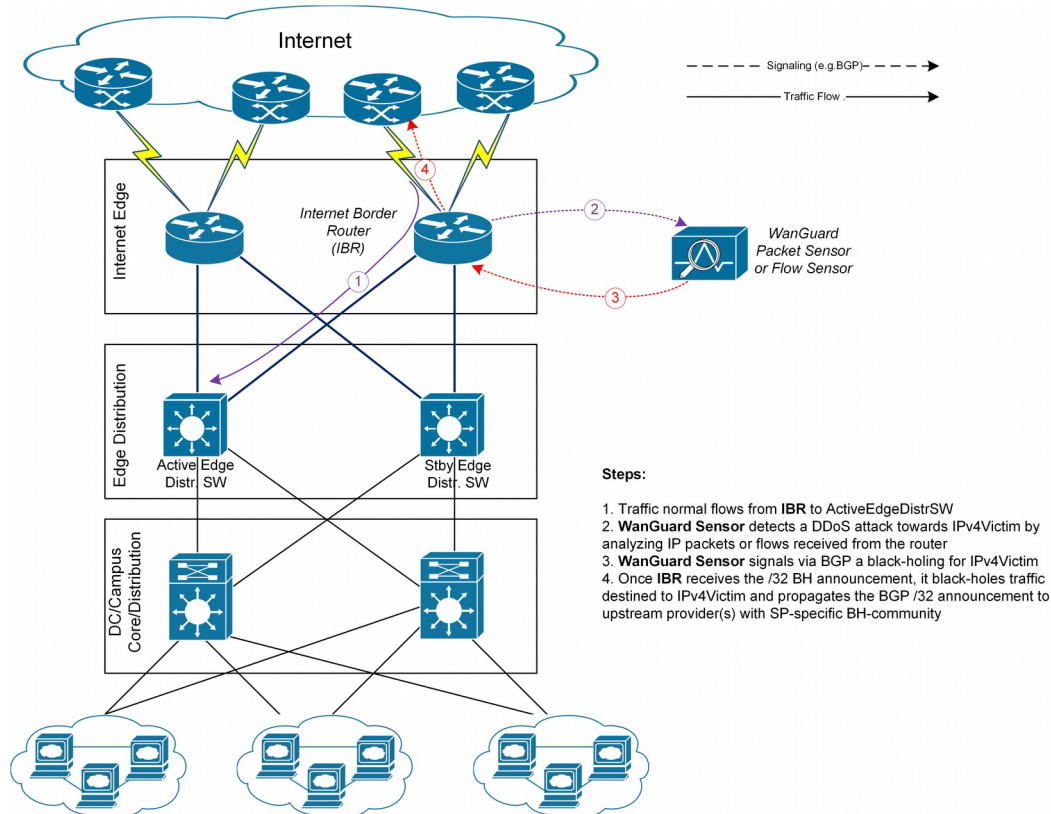
To simplify, we will start from the following scenario: an attack is detected by WanGuard Sensor (hereby referred simply as **Sensor**) that decides to react by using the BGP black hole approach rather than diverting traffic for scrubbing by WanGuard Filter.

In RTBH setup, **Sensor** would play the role of **Trigger**.

After an attack is detected, **Sensor** signals the **IBR** (Internet Border Router) via BGP that all traffic destined to **IPv4-Victim** has to be dropped. In more details:

- **Sensor** advertise via BGP an **IPv4-Victim/32** prefix with a specific community to be identified as a Black Hole announcement
- The IBR receives the announcement and it inserts the route in its routing table as **IPv4-Victim/32** with next-hop Null0.
- Furthermore, the **IBR** advertises this route to its upstream providers (**ISPs**) changing at the same time the community used for internal purposes, to a community which is relevant to the correspondent ISP.

For a better understanding you may refer to the diagram below:



Black-holing on upstream

The principle of DDoS mitigation using black hole BGP advertisements is to propagate the BH-prefix from the destination of the attack closest as possible to the source. Most ISPs have defined a public community, based on which their IBRs take decision to black hole the traffic destined to victim by routing it to Null0. In comparison to redirect announcements, the black-holing announcements have to be advertised to upstream ISPs.

In order to black hole the attack on upstream provider, the black hole route must be tagged/marked with an appropriate BGP standard community. This community is provider-specific and has to be requested by each customer to the provider, or it might be found on IRR ASN details (e.g. RIPE, APNIC, ARIN, etc).

On IBR there shall be a routing-policy applied on to-ISP-BGP neighbor (export-direction) which shall **rewrite** the internal BH-community to appropriate ISP's BH-community.

From a BGP configuration point of view, the Sensor's configuration is quite similar with Filter's BGP configuration explained in Annex 4 on page 110, having one exception in regards to the BGP community that will be used to mark black hole routes. Considering this, only the IBR's configuration will be further detailed.

IBR BGP Session with WanGuard Sensor – Cisco Router BGP Configuration

```
r7500(config)# ip bgp-community new-format
r7500(config)# ip community-list <WanGuard-Sensor-community-name> permit <BH-community> →
e.g. 65000:66
r7500(config)# route-map WanGuard-Filter-in permit 10
r7500(config-route-map)# match community <WanGuard-Sensor-community-name>
r7500(config-route-map)# set local-preference 200 → it will assure a higher priority against
redirect-route
r7500(config-route-map)# set ip next-hop 192.168.255.255 → this target-IP must not be used
on your network
r7500(config-route-map)# exit
r7500(config)# route-map WanGuard-Sensor-out deny 10
r7500(config-route-map)# exit
r7500(config)# ip route 192.168.255.255 255.255.255.255 Null0 → BH route for target-IP
r7500(config)# router bgp <Router-AS-number>
r7500(config-router)# bgp log-neighbor-changes
r7500(config-router)# neighbor <WanGuard-Sensor-IP-address> remote-as <WanGuard-Sensor-AS-
number>
r7500(config-router)# neighbor <WanGuard-Sensor-IP-address> description <description>
r7500(config-router)# neighbor <WanGuard-Sensor-IP-address> soft-reconfiguration-inbound
r7500(config-router)# neighbor <WanGuard-Sensor-IP-address> route-map WanGuard-Sensor-out
out
r7500(config-router)# neighbor <WanGuard-Sensor-IP-address> route-map WanGuard-Sensor-in in
r7500(config-router)# no synchronization
r7500(config-router)# exit
```

BGP Session with Two ISPs – Cisco Router BGP Configuration

```
r7500(config)# route-map IBR-ISP1-out permit 5 → assumes that additional entries are defined
and allow customer-routes
r7500(config-route-map)# match community <WanGuard-Sensor-community-name>
r7500(config-route-map)# set community <ISP1-BH-Community> → e.g.111:9999
r7500(config-route-map)# exit
r7500(config)# route-map IBR-ISP2-out permit 5 → assumes that additional entries are defined
```

and allow customer-routes

```
r7500(config-route-map)# match community <WanGuard-Sensor-community-name>
r7500(config-route-map)# set community <ISP1-BH-Community> → e.g.222:9999
r7500(config-route-map)# exit
r7500(config)# router bgp <Router-AS-number>
r7500(config-router)# neighbor <ISP1-IP-address> remote-as <ISP1-AS-number>
r7500(config-router)# neighbor <ISP1-IP-address> route-map IBR-ISP1-out out
r7500(config-router)# neighbor <ISP2-IP-address> remote-as <ISP2-AS-number>
r7500(config-router)# neighbor <ISP2-IP-address> route-map IBR-ISP2-out out
r7500(config-router)# no synchronization
r7500(config-router)# exit
```

When multiple ISPs and IBRs exist, it makes sense to have different BH communities, one for each IBR. In this way you may isolate the source of attack and not whole traffic directed to victim would be black-holed.

Interaction with traffic diversion / WanGuard Filter

It might be the case when:

- Filter advertises redirect BGP route to IBR (initially)
- Sensor advertises a black-hole BGP route to IBR (afterwards)

The priority shall be on black-hole advertisement, rather than redirect. This can be achieved easily by using a routing-policy which sets a higher priority on black hole route (e.g. set Local-Preference at 200 for BH-route).

The direction and place where BGP routing-policy has to be implemented is strongly dependent on:

- What role plays on the network the Sensor's peer-router (e.g. IBR, Route-Reflector, etc.)
- Type of BGP relation between Sensor and the peer-router (e.g. iBGP or eBGP)

In order to distinguish between a black hole and a redirect announcement, it is recommended to use different BGP communities on each type of announcement.

The action shall be like on the table below:

Type of BGP announcement (community)	Route to (next-hop)	Propagated to ISP
Redirect (e.g. 65000:99)	WanGuard Filter	No
Black-hole (e.g. 65000:66)	Null0	Yes

Table 1 – BGP Communities and actions

In the special case when the peer-router of Sensor is the Route-Reflector, then the black-hole action still has to be implemented on IBR. To achieve this, the above sample router configuration has to be adapted and applied on IBR BGP-import policy in relation to the Route-Reflector. No action has to be implemented on RR, while its purpose is route-signaling rather than routing traffic.

To use a single bgpd / peer router for both redirect and black-hole, define *bgp multiple-instance* in bgpd.conf and use 2 BGP Connections configured with distinct AS views. For the second AS view assign a different IP for *bgp router-id* and make sure the IP (sub-interface) can reach the router.

Appendix 4 – Network Integration Guideline for WanGuard Filter

This appendix describes how to configure the network for traffic scrubbing by **WanGuard Filter**, starting from a couple of common deployment scenarios of the filtering server.

WanGuard Filter, hereby referred simply as **Filter**, can be deployed following two scenarios:

- **In-line filtering.** This deployment scenario can have two possible deployments, depending on the role of the filtering server on the forwarding path:
 - *Routing mode*
 - *Bridging mode*
- **Out-of-line filtering.** Due to the complexity of the **Out-of-line filtering** solution, this appendix will further focus on this setup.

When the **Out-of-line filtering** solution is deployed, the following two major operations have to be considered, operations that have to be performed from network point of view:

1. **Traffic diversion** – how the traffic for a certain destination (**IP-Victim**) is diverted from network to the filtering server
2. **Traffic forwarding or Re-injection** – how the cleaned traffic is put back on network to be routed / forwarded towards its destination (**IP-Victim**)

The information provided here regarding router configurations is for informational purposes only. Please refer to the appropriate router user guides for more detailed and up-to-date information.

Understanding Traffic Diversion Method

The method relies on a basic routing principle implemented on all routers according to which a router selects the path with the longest prefix match present on routing table (also known as the “most specific” entry from routing table).

BGP has been chosen as routing protocol to inject/advertise the most specific redirect-prefix (e.g. a /32 for IPv4, a /128 for IPv6) towards *Internet Border Router (IBR)*. The IBR is the router which assures routing between ISP and internal network (customer network).

To simplify, we'll consider an **IPv4-Victim**. In this case, **Filter** sends a BGP routing update towards IBR for **IPv4-Victim/32** with a next-hop to itself forcing in this way the IBR to choose the path to **IPv4-Victim** via **Filter**. The main condition for this to work is to have the redirect announcement to be the best from BGP election process and from *Routing-Table Manager (RTM)*.

If on the routing-table there is already a /32 present, then additional configuration have to be made in order to assure that redirect-announcement will be inserted into the routing table and used upon deciding the forwarding path.

Please refer to the following logical diagram which describes the high-level process of detection-diversion-cleaning and re-injection.

The following terminology is used:

- **Divert-from router** – The router from which traffic, initially intended for the victim, is diverted towards **Filter** (e.g. *IBR*) – this router has to receive a redirect-prefix via BGP
- **Inject-to router** – The router where **Filter** will forward the cleaned traffic towards the attacked destinations (IP-Victims)
- **Next-hop router** – The router that is normally the next-hop to the destinations according to the routing-table on the **Divert-from router** before traffic diversion is activated.

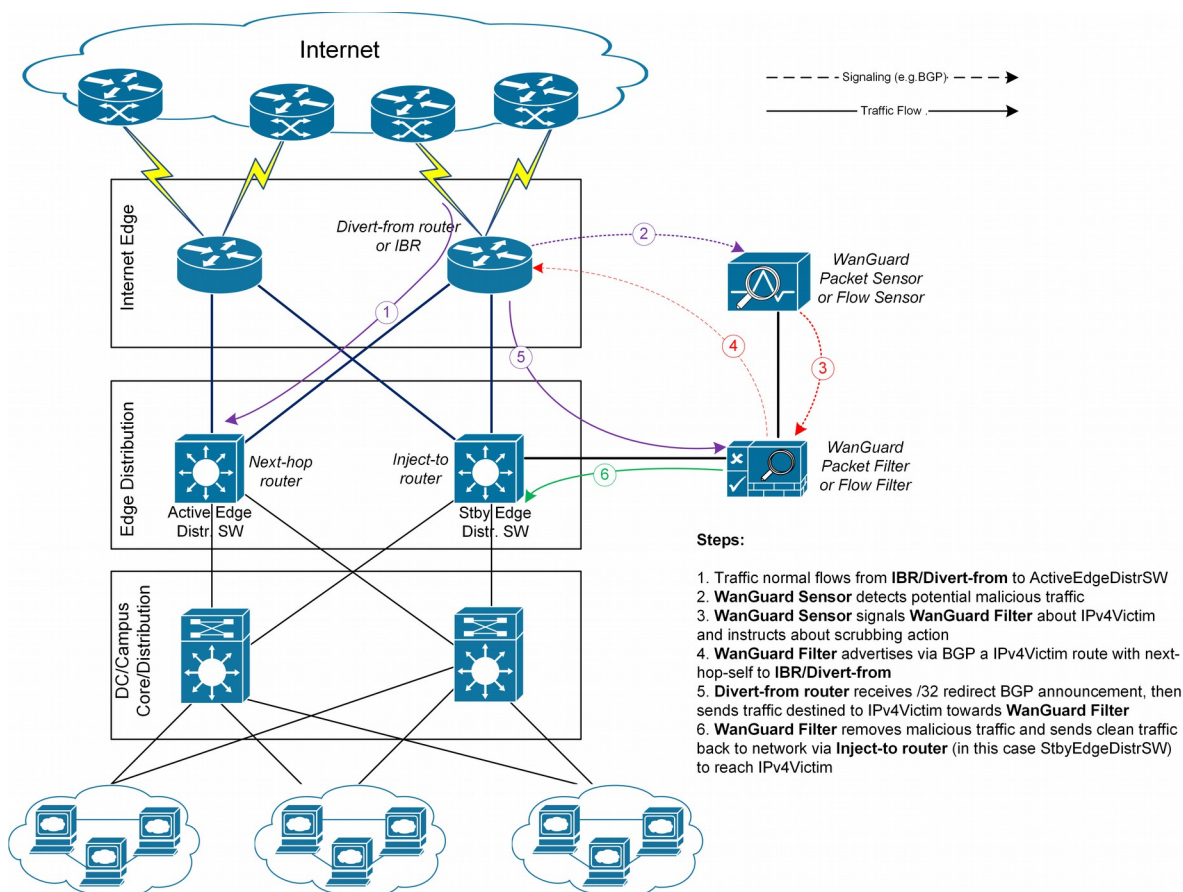


Figure-1. Logical Diagram for an Enterprise Network – how traffic diversion works

From a configuration point of view the following steps have to be performed:

1. Configure *traffic-diversion* using BGP as the signaling method
2. Configure an appropriate clean *traffic-injection* method to send clean traffic back on the network to be forwarded towards the victim

BGP Configuration Guideline

This section provides a general guideline for BGP configuration on the **Filter** server and on a *Divert-from*

router. The guidelines provided in this section apply to the BGP configuration on any router from which **Filter** diverts traffic.

To simplify, the following examples are provided using eBGP (external BGPv4). This solution is not limited to eBGP, iBGP may be considered as well, depending on existing network setup, case in which “*set nexthop-self*” feature might be required.

The steps below have to be followed:

1. Configure BGPd on **Filter** with an easily recognizable autonomous system number. This can be a private ASN for eBGP (e.g. ASN16bit 64512-65534), or your own public ASN in case you’re using iBGP. The BGPd sends routing information only when it diverts traffic. This route appears in the router's routing tables. Using a recognizable value allows you to identify easily the *redirect-prefixes* in the router's routing tables.
2. Configure additional precaution measures to prevent any undesirable routing behavior:
 - a. Configure **Filter** to not accept any prefix/advertisements from **Divert-from router**
 - b. Configure **Divert-from router** to not advertise any prefix towards **Filter**
 - c. Configure **Divert-from router** to accept only redirect-prefixes from **Filter** (e.g. /32 prefixes)
 - d. Configure **Filter** to advertise the redirect-prefixes with well-known community *no-advertise* – this would prevent redirect-prefixes/announcements to be propagated to other peers thorough BGP. The *no-export* community might be used in case redirect-prefix has to be advertised to additional routers, or Route-Reflectors are used in-between **Filter** and **Divert-from router**. Both communities will prevent BGP-redirect-announcements to be advertised towards upstream providers. However, as a good practice is to mark this announcement with a dedicated BGP community to distinguish between redirect and black hole announcements.

3. To ease the trouble-shooting process you may consider the *soft-reconfiguration inbound* command on **Divert-from-router** during the setup procedures.

Quagga / bgpd Configuration

WanGuard uses the BGPd daemon provided by the Quagga routing software suite (<http://www.quagga.net>).

After installing Quagga, you will have to do few distribution-specific configuration changes:

- On Red Hat or CentOS systems, edit `/etc/sysconfig/quagga` and replace `BGPD_OPTS="-A 127.0.0.1"` with `BGPD_OPTS=""`.

```
[root@localhost ~]# nano /etc/sysconfig/quagga → on Red Hat or CentOS systems
```

- On Debian or Ubuntu systems, edit `/etc/quagga/daemons` and replace `bgpd=no` with `bgpd=yes`. Edit `/etc/quagga/debian.conf` and replace `bgpd_options=" --daemon -A 127.0.0.1"` with `bgpd_options="--daemon"`.

```
[root@localhost ~]# nano /etc/quagga/daemons → on Debian or Ubuntu systems
[root@localhost ~]# nano /etc/quagga/debian.conf → on Debian or Ubuntu systems
```


WanGuard needs to connect to bgpd through the public IP of the server. This is why the “-A 127.0.0.1”, used for binding bgpd to the loopback interface, must be deleted.

To be able to start the bgpd service, create a basic configuration file. Setting a password for the bgpd daemon is usually enough to get it started. You should replace “bgppass” with your own password.

```
[root@localhost ~]# echo 'password bgppass' > /etc/quagga/bgpd.conf
[root@localhost ~]# chown quagga /etc/quagga/bgpd.conf
[root@localhost ~]# service bgpd start → on Red Hat or CentOS systems
[root@localhost ~]# service quagga start → on Debian or Ubuntu systems
```

It is a good idea to tighten the security of the bgpd daemon. Connect to the bgp daemon with telnet on localhost port 2605 (default bgpd port) with the previously-defined password (“bgppass”). Issue the following commands and replace “enablepass” with your own configuration-mode password.

```
[root@localhost ~]# telnet 127.0.0.1 2605
localhost> enable
localhost# config terminal
localhost(config)# service password-encryption
localhost(config)# enable password enablepass
localhost(config)# write
```

Configure routing on BGPd using the commands shown in the following example. Please note that you can use the prefix-list, route-map, or distribute-list method for filtering outgoing routing information about the router. To have a uniform approach, the following example uses route-maps. Optionally, BGP authentication can be configured to increase security and avoid any illegal BGP announcement which may lead to a security breach.

```
localhost(config)# router bgp <WanGuard-Filter-AS-number>
localhost(config-router)# bgp router-id <WanGuard-Filter-IP-address>
localhost(config-router)# neighbor <Router-IP-address> remote-as <Router-AS-number>
localhost(config-router)# neighbor <Router-IP-address> description <description>
localhost(config-router)# neighbor <Router-IP-address> password <BGP MD5 password>
localhost(config-router)# neighbor <Router-IP-address> route-map WanGuard-Filter-in in
localhost(config-router)# neighbor <Router-IP-address> route-map WanGuard-Filter-out out
localhost(config-router)# exit
localhost(config)# route-map WanGuard-Filter-in deny 10
localhost(config-route-map)# exit
localhost(config)# route-map WanGuard-Filter-out permit 10
localhost(config-route-map)# set community no-advertise <WanGuard-Filter-community>
localhost(config-route-map)# exit
localhost(config)# write
localhost(config)# exit
```

To display the router configuration, enter the *show running-config* command from the “enable” command level. In the following example, the router's AS number is 1000, and the BGPd AS number is 65000.

The following partial sample output is displayed:

```
localhost# show running-config
... skipped ...
router bgp 65000
bgp router-id 192.168.1.100
```

```

neighbor 192.168.1.1 remote-as 1000
neighbor 192.168.1.1 description divert-from router
neighbor 192.168.1.1 soft-reconfiguration inbound
neighbor 192.168.1.1 route-map WanGuard-Filter-in in
neighbor 192.168.1.1 route-map WanGuard-Filter-out out
!
route-map WanGuard-Filter-in deny 10
!
route-map WanGuard-Filter-out permit 10
set community no-advertise
!
line vty
... skipped ...

```

WanGuard connects to bgpd using the BGP Connection component documented on page 50.

Cisco Router BGP Configuration

This section describes the router's BGP configuration used when configuring traffic diversion. The syntax of the commands is taken from the BGP configuration on a Cisco router. The following configuration steps show the commands used to configure BGP on a Cisco router:

```

r7200(config)# ip bgp-community new-format
r7200(config)# ip community-list standard <WanGuard-Filter-community-name> permit no-
advertise
r7200(config)# ip community-list standard <WanGuard-Filter-community-name> permit <WanGuard-
Filter-community>
r7200(config)# route-map WanGuard-Filter-in permit 10
r7200(config-route-map)# match community <WanGuard-Filter-community-name> exact
r7200(config-route-map)# exit
r7200(config)# route-map WanGuard-Filter-out deny 10
r7200(config-route-map)# exit
r7200(config)# router bgp <Router-AS-number>
r7200(config-router)# bgp log-neighbor-changes
r7200(config-router)# neighbor <WanGuard-Filter-IP-address> remote-as <WanGuard-Filter-ASn>
r7200(config-router)# neighbor <WanGuard-Filter-IP-address> description <description>
r7200(config-router)# neighbor <WanGuard-Filter-IP-address> soft-reconfiguration-inbound
r7200(config-router)# neighbor <WanGuard-Filter-IP-address> route-map WanGuard-Filter-out
out
r7200(config-router)# neighbor <WanGuard-Filter-IP-address> route-map WanGuard-Filter-in in
r7200(config-router)# exit

```

To display the router configuration, enter the *show running-config* command from the router global command level. In the following example, the router's AS number is 1000 and the BGPd AS number is 64000. The following partial output is displayed:

```

r7200# show running-config
... skipped ...
router bgp 1000
bgp log-neighbor-changes
neighbor 192.168.1.100 remote-as 64000

```



```
neighbor 192.168.1.100 description Filter appliance
neighbor 192.168.1.100 soft-reconfiguration inbound
neighbor 192.168.1.100 route-map WanGuard-Filter-out out
neighbor 192.168.1.100 route-map WanGuard-Filter-in in
no synchronization
!
ip bgp community new-format
ip community-list expanded WanGuard-Filter permit no-advertise
ip community-list expanded WanGuard-Filter permit <WanGuard-Filter-community>
!
route-map WanGuard-Filter-in permit 10
    match community WanGuard-Filter exact match
!
route-map WanGuard-Filter-out deny 10
!
... skipped ...
```

Understanding Traffic Forwarding Methods

This section provides details on the available traffic forwarding methods. A traffic forwarding method must be used to re-inject cleaned traffic from the **Filter** system back to network in order to reach its destination.

Couple of options can be identified depending on the existing network setup and which device may have the role of **Divert-from**, **Inject-to** and **Next-hop** router:

1. **Layer 2 Forwarding Method**
2. **Layer 3 Forwarding Method**

Layer 2 Forwarding Method

The following characteristics will describe this option:

- **Filter** system, **Divert-from** router, and **Next-hop** router are on the same network or VLAN sharing the same subnet
- **Divert-from** and **Inject-to** routers are two different devices
- **Next-hop** and **Inject-to** routers are the same device

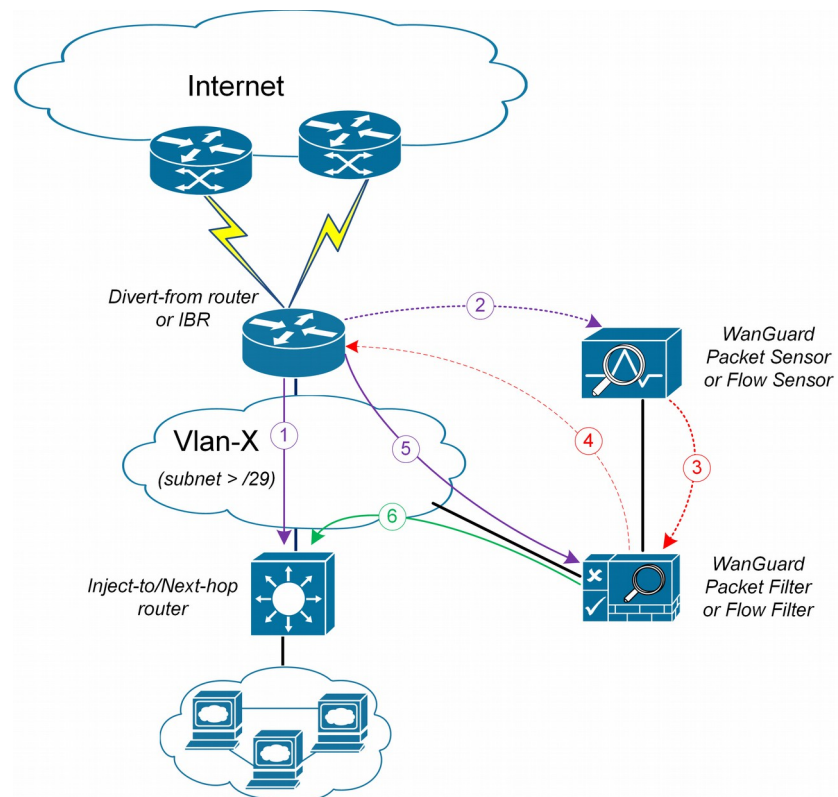


Figure-2. Logical Diagram for Layer 2 Forwarding (*same steps as per Fig.1)

While the above solution assumes one **Divert-from** and one **Inject-to** router, couple of variations may be considered starting from this option:

- a) Multiple **Divert-from** routers
- b) Multiple **Inject-to** routers
- c) Combination of above and/or multiple VLANs in between **Divert-from** and **Inject-to** routers

Considering the last scenario, the **Filter** system has to be connected on each VLAN and to have static routes for each destination via the **Inject-to/Next-hop** routers.

Warning: Any special L2 configuration on Filter interface (e.g. bonding, VLAN-tagging, etc) will impact scrubbing/forwarding performance of Filter, while hardware optimizations from NICs are bypassed.

In case the VLAN/LAN cannot be extended to also include the **Filter** system on it, then a dedicated point-to-point connection might be considered between (**Filter** and **Divert-from**) or (**Filter** and **Inject-to/Next-hop**)

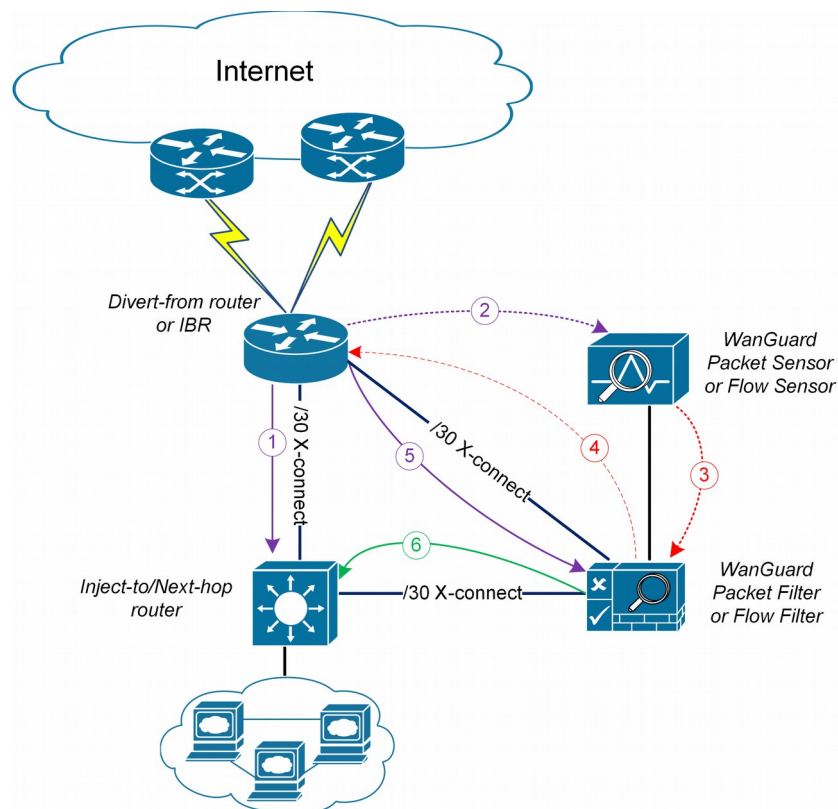


Figure-3. Logical Diagram Layer 2 Forwarding – dedicated cross-connects (*same steps as per Fig.1)

Layer 3 Forwarding Method

The following characteristics will describe this option:

1. **Divert-from** and **Inject-to** routers are the same device – referred in this case as “the router”

2. Depending on **Next-hop** router role we may have following sub-options:
 - a) **Next-hop** router is on dedicated device, but is not direct connected to **Filter**
 - b) **Next-hop** router is on same device as **Diver-from/Inject-to** routers

In scenario 2a, a routing-loop issue may occur between **Diver-from/Inject-to** router and **Filter**:

- **Filter** sends a BGP redirect announcement to **Diver-from** router (e.g. a /32 prefix route)
- **Diver-from** router will send all traffic for that **Victim-IP** to **Filter**
- **Filter** cleans the traffic and returns the cleaned traffic to the same router – Inject-to/Diver-from
- The **Inject-to** router has the redirect route /32 on its routing table and will send back the clean-traffic towards the **Filter** system resulting a routing-loop

There are a couple of solutions to overcome this issue (these are suggestions and solution shall not be limited on these):

1. Using **GRE** (Generic Routing Encapsulation) or any L3-tunneling between **Filter** and **Next-hop** router – in this case routing-loop is avoided by pushing clean traffic over the GRE-tunnel to **Next-hop** router through **Diver-from/Inject-to** router, bypassing in this way the /32 diversion-route from **Diver-from/Inject-to**:

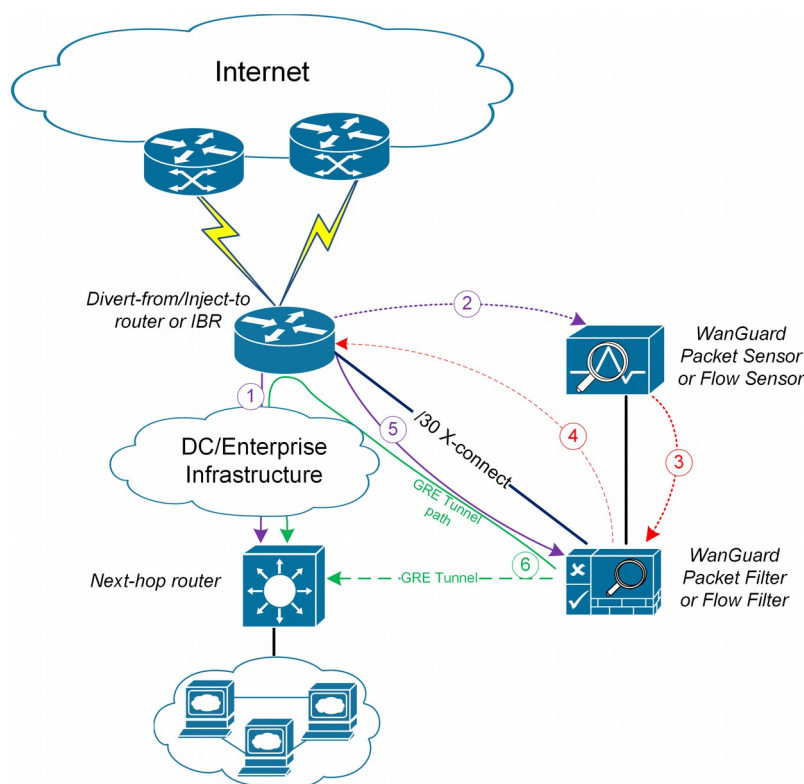


Figure-4. Logical Diagram Layer 3 Forwarding using GRE (*same steps as per Fig.1)

- Using **PBR** (Policy Base Routing) to override the normal routing decision from **Divert-from/Inject-to** router:

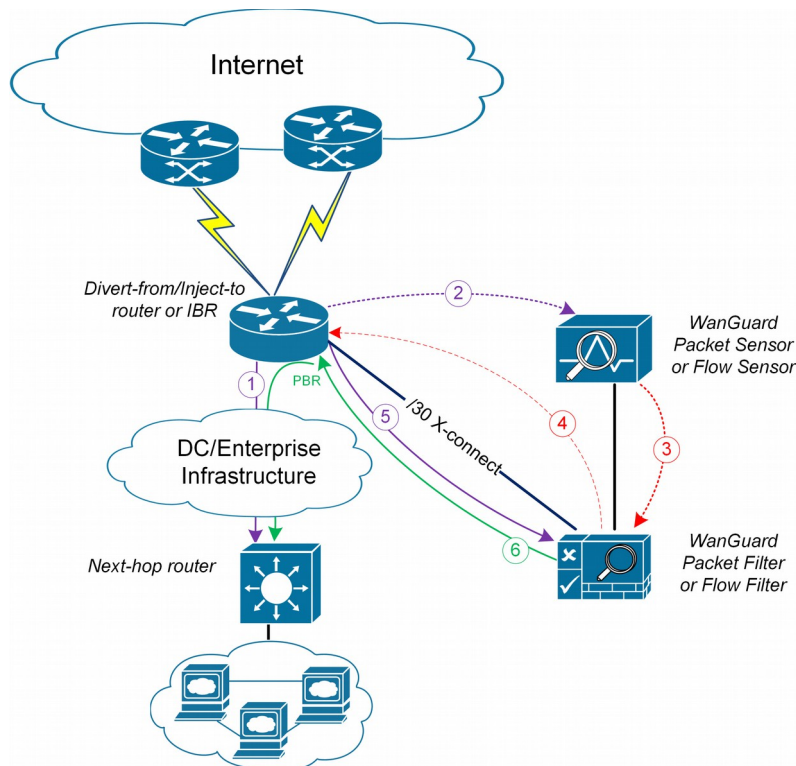


Figure-5. Logical Diagram Layer 3 Forwarding using PBR (*same steps as per Fig.1)

Warning: PBR may impact router performance – depending on platform type, some optimizations may exist. However, by default PBR relies on packet-by-packet processing (process-switching) which have significant impact on router's CPU.

In case multiple **Next-hop** routers exist, then the following have to be considered too:

- multiple GRE tunnels have to be deployed and static routes at **Filter** level have to be considered, or
- multiple entries on PBR matching each zone, depending on which option is chosen

In case of GRE, you must run on **Filter** the standard Linux tool *ip* to create and route GRE / IP over IP tunnels that will be used to inject the cleaned traffic back into the network. You must then configure **Filter** (see Packet Filter Configuration) with the Outbound Interface set to the virtual network interface created by the tunnel.

Please refer to the below router configuration samples for both GRE and PBR options:

- The GRE method (using Cisco CLI) – configuration from **Next-hop** router:

```

r7200(config)# interface Tunnel 1
r7200(config-if)# ip address <X.X.X.X> 255.255.255.252
r7200(config-if)# ip mtu 1500
r7200(config-if)# ip tcp adjust-mss 1456
r7200(config-if)# tunnel source <Y.Y.Y.Y> → where Y.Y.Y.Y is the IP from Next-hop router
r7200(config-if)# tunnel destination <Z.Z.Z.Z> → where Z.Z.Z.Z is the IP from Filter

```

Notes:

- *source IP and destination IP have to be reachable*
- *default tunneling encapsulation is GRE*
- *routing of tunnel-destination must be assured (e.g. using static routes)*
- **Filter** will have X.X.X.X-1 IP on its Tunnel interface
- *If transport between **Filter** and **Next-hop** router supports jumbo frames, then adjust MTUs accordingly in order to avoid additional packet fragmentation, and implicitly performance degradation*

2. The PBR method (using Cisco CLI):

```

r7200(config)#ip access-list standard WanGuard-Filter-IPScope
r7200(config-std-acl)#permit A.A.A.A/BB → multiple entries may exists
r7200(config-std-acl)#exit

r7200(config)#route-map WanGuard-Filter-PBR permit 10
r7200(config-route-map)# match ip address WanGuard-Filter-IPScope
r7200(config-route-map)# set ip next-hop <C.C.C.C> → where C.C.C.C is the IP of Next-hop
router which is direct connected to Divert-from router
r7200(config-route-map)#exit
r7200(config)#interface GigabitEthernet 0/0
r7200(config-if)#ip policy route-map WanGuard-Filter-PBR
r7200(config-if)#exit
r7200(config)#

```

On scenario 2b when only one device has all three roles: **Divert-from**, **Inject-to** and **Next-hop** – neither of above options can be considered. PBR might be considered in case a “set interface” configuration may take traffic and put it on the right Layer 2 path to its destination; since this is dependent on the type of platform used as router, this would have limited applicability and will not be treated further more.

For scenario 2b a much more elaborated solution has to be considered. The main idea is to separate virtually the routing domain used by **Divert-from** and **Inject-to/Next-hop** router – falling in this way somehow on **Layer 2 Forwarding Method**:

- use VRF-Lite by defining two VRF's:
 - one for “**outside**” where **Divert-from** router is (and also its BGP peering with upstream providers and **Filter**)
 - and another one for “**inside**” where **Inject-to/Next-hop** router are
- **Filter** must have two Layer 3 interfaces/sub-interface:
 - one in **VRF-outside**
 - one in **VRF-inside**
- like on **Layer 2 Forwarding Method**, static routes have to be defined on **Filter** towards subnets destinations

- in order to assure normal routing between these two VRF's, MPBGP have to be activated on “**the router**”; no MPBGP neighbor have to be defined
- on VRF's definitions special policies for import/export Route-Targets(RT) have to be defined in following manner:
 - e.g. mark outside routes with RT 65000:100 and inside routes RT 65000:200
 - on **VRF-outside**:
 - import the routes having outside-RT(e.g. 65000:100) and also inside-RT(e.g. 65000:200)
 - export routes with outside-RT – excepting the redirect/diversion routes
 - on **VRF-inside**:
 - import the routes having inside-RT and specific routes having outside-RT: the default-route and/or all other outside routes excepting the routes for diversion learned from **Filter**
 - export routes with inside-RT

In this way, the inside routing table will not know about the /32 redirect prefix and will forward/route traffic normally.

For a better understanding please refer to **Figure-6** and configuration on “**router**” using Cisco-CLI as example:

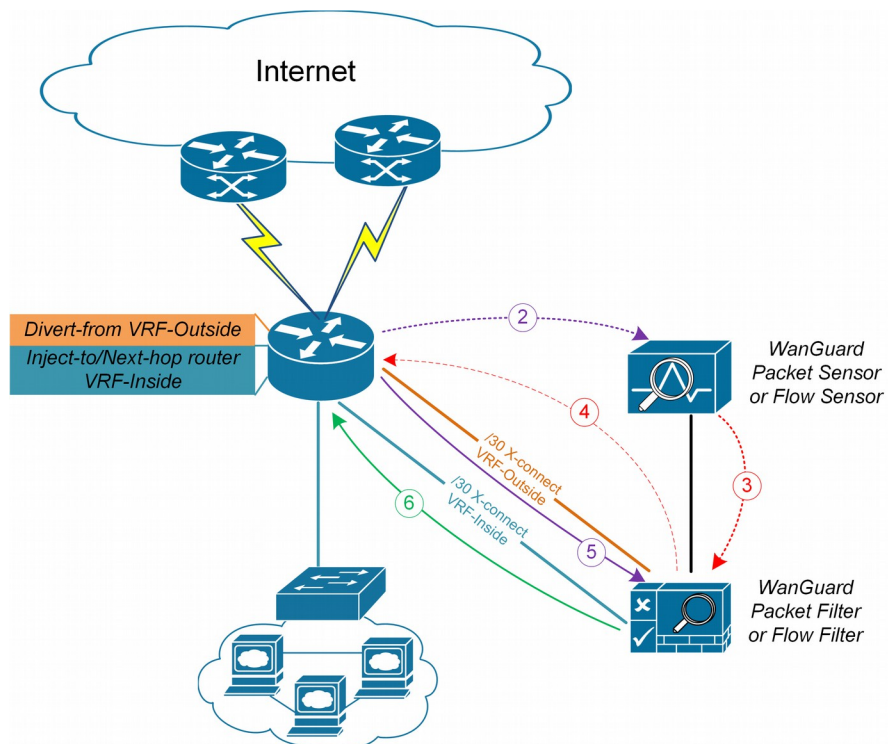


Figure-6. Logical Diagram Layer 3 Forwarding using VRF-Lite (*same steps as per Fig.1)


```

r7200(config)#ip extcommunity-list standard VRF-Inside permit rt 65000:200
r7200(config)#route-map VRF-Inside-Import deny 10
r7200(config-route-map)#match community WanGuard-Filter → The WanGuard-Filter community has
been already configured above; this will deny redirect-routes
r7200(config-route-map)#exit
r7200(config)#route-map VRF-Inside-Import permit 20 → This will allow any other routes
r7200(config-route-map)#exit
r7200(config)#
r7200(config)#ip vrf Outside
r7200(config-vrf)#rd 65000:100
r7200(config-vrf)#route-target import 65000:100
r7200(config-vrf)#route-target import 65000:200
r7200(config-vrf)#route-target export 65000:100
r7200(config-vrf)#exit
r7200(config)#
r7200(config)#ip vrf Inside
r7200(config-vrf)#rd 65000:200
r7200(config-vrf)#route-target import 65000:100
r7200(config-vrf)#route-target import 65000:200
r7200(config-vrf)#import map VRF-Inside-Import
r7200(config-vrf)#route-target export 65000:200
r7200(config-vrf)#exit
r7200(config)#
r7200(config)# interface Loopback0 → This is needed to have a BGP router-id (any existing
Loopback from global can be reused)
r7200(config-if)# ip address <Z.Z.Z.Z/32>
r7200(config-if)#no shut
r7200(config-if)#exit
r7200(config)#
r7200(config)# interface <to Upstream Provider>
r7200(config-if)#ip vrf forwarding Outside → Warning! This will remove IP address from
interface/IP-address has to be reconfigured again
r7200(config-if)#exit
r7200(config)#
r7200(config)# interface <to Filter off-ramp interface>
r7200(config-if)#ip vrf forwarding Outside → Warning! This will remove IP address from
interface/IP-address has to be reconfigured again
r7200(config-if)#exit
r7200(config)#
r7200(config)# interface <to Filter on-ramp interface>
r7200(config-if)#ip vrf forwarding Inside → Warning! This will remove IP address from
interface/IP-address has to be reconfigured again
r7200(config-if)#exit
r7200(config)#
r7200(config)# interface <to Inject-to/Next-hop>
r7200(config-if)#ip vrf forwarding Inside → Warning! This will remove IP address from
interface/IP-address has to be reconfigured again
r7200(config-if)#exit
r7200(config)#
r7200(config)#router bgp 65000 → You may use your ASN instead of 65000
r7200(config-router)# no synchronization
r7200(config-router)#bgp log-neighbor-changes
r7200(config-router)#no auto-summary
r7200(config-router)#address-family vpnv4
r7200(config-router-af)# no synchronization
r7200(config-router-af)#exit-address-family
r7200(config-router)# address-family ipv4 vrf Inside
r7200(config-router-af)# no synchronization
r7200(config-router-af)# redistribute connected
r7200(config-router-af)# redistribute <other IGP/static if needed>
r7200(config-router-af)#exit-address-family

```



```
r7200(config-router)# address-family ipv4 vrf Outside
r7200(config-router-af)# no synchronization
r7200(config-router-af)# redistribute connected
r7200(config-router-af)# redistribute <other IGP/static if needed>
r7200(config-router-af)#exit-address-family
r7200(config-router)#exit
r7200(config)#
```

If too many GRE tunnels or PBR entries have to configured/maintained, consider the VRF-Lite solution.

Appendix 5 – Software Changelog

WanGuard 6.1

Release date: December 3 2015

- Administrators can create custom decoders that identify flows or packets sharing a certain pattern (e.g. to differentiate and classify the underlying protocols) in Configuration » General Settings » Custom Decoders.
- New Filter mitigation options in Configuration » General Settings » Mitigation Options:
 - TCP SYN Proxy
 - invalid TCP flags
 - invalid DNS packets
 - private/reserved IPs
 - connection-oriented or connection-less traffic rate-limiting
 - blacklisting by IP reputation services
- Filter can apply new filtering rules for: specific packet payloads, countries, DNS transaction IDs.
- Filtering rules can be disabled, re-ordered and fine-tuned for each decoder.
- A tighter integration between Filter and the software firewall (Netfilter framework) and Chelsio hardware filters. Newly generated anomaly reports contain pass/drop graphs for mitigated attacks.
- Console users can create custom firewall rules in Reports » Tools » Firewall Rules.
- New Software Firewall options in the Filter Configuration window. A new “FW Policy” field on Whitelist rules that explicitly permits traffic through the Software Firewall.
- Configuration » General Settings » Anomaly Detection contains a new option for deduplicating anomalies that indicate the same attack matched by different decoders.
- Filter Clusters can be associated with other Filter Clusters.
- BGP Connections can be configured to allow BGP announcement withdrawals to be done after business hours.
- Sensor graphs now use RRDCached when it is defined in Configuration » General Settings » Graphs & Storage Configuration.
- Enhanced user authentication methods. New RADIUS options and a new HTTP authentication option.
- A new TCP-ALL decoder.
- The Latest Events tab from the South Region contains selectors for severity and components.
- User role renamed Guest. Administrators can allow Guest access to Reports » Tools with greater granularity.
- Configuration » General Settings » Anomalies renamed Anomaly Detection. Reports » Alerts & Tools renamed Tools.

- Unattended installation when the following shell environment variables are set: WANGUARD_INSTALL_DB_USER, WANGUARD_INSTALL_DB_PASS, WANGUARD_CONSOLE_IP, WANGUARD_CONSOLE_DB_PASS.
- User Guide updated. Contains new Appendixes describing advanced BGP configurations.
- Various small fixes.

WanGuard 6.0

Release date: February 16 2015

System

- The software can be installed on new Linux distributions: Red Hat 7, CentOS 7, Debian 7, Ubuntu Server 14.
- Console supports PHP 5.5 and PHP 5.6.
- Graphs for iowait in Reports » Servers » Server Graphs.
- Configuration » General Settings » Software Updates displays the latest software version and upgrading instructions.
- Emails can be sent directly by Console without requiring a local MTA. New Configuration » General Settings » Outgoing Email Settings, with configurable Sender Email.
- Fixed sending emails to CC addresses.
- Corrupted Console database can be repaired with `"/opt/andrisoft/bin/WANmainenance repair"`.
- 32-bit architectures are no longer supported.

Console

- A new graphical slider for quick selection of custom time frames in Reports.
- Reports and Configuration side regions can be set apart by user preference, e.g. one on the right and one on the left. New Ctrl→R keyboard shortcut toggles side regions.
- Configuration » General Settings » Data Retention shows disk usage for newly created RRD files containing IP graph data.
- Graphing IP sweeps can be enabled or disabled for IPv6 and/or IPv4 in Configuration » General Settings » Graphs & Storage.
- Changed Conditional and Dynamic Parameters: {operation}, {sensor_type}, {domain}, {class}, {filter_*}, {filter_tcpdump_size}.
- New Dynamic Parameters: {from_year}, {from_month}, {from_day}, {from_dow}, {from_hour}, {from_minute}, {until_year}, {until_month}, {until_day}, {until_dow}, {until_hour}, {until_minute}, {direction_to_from}, {software_version}, {comparison}, {direction_receives_sends}, {duration_clock}, {*_decoder_prefix} for {*_prefix}, {filter_type}, {filter}, {filter_id}, {response_actions},

{filtering_rule_log_size}, {filtering_rule_max_unit}, {filtering_rule_unit}.

- Redesigned Response Configuration window. New email templates.
- Redesigned IP Zone Configuration window.
- New widgets: Flow Records and Flow Tops.
- Dashboards can be configured to have a unique time frame for all containing widgets.
- Unprivileged users can open reports for IPs included in the allowed subnets.
- Loading of IP Zones with thousands of IPs and subnets is around 8 times faster.
- Moved Configuration » General Settings » User Management » Authentication & Login to Configuration » General Settings » User Authentication.
- Add Configuration » General Settings » User Authentication » Login Window Notification and Successful Login Notification.
- Radius authentication fixed.
- New statistics in by Reports » Components » Overall » Console.
- Reports » Attacks & Tools » Anomalies » Active Anomalies » Reverse DNS unchecked by default.
- Reports » Attacks & Tools » Anomalies » Active Anomalies shows a Flow Trace button for anomalies detected by Flow Sensors.
- The visibility of items in Reports » Components and Reports » Servers can be toggled. Right-click opens their configuration.
- Configuration » Components and Configuration » Schedulers items can be activated/inactivated with a single right click.
- Configuration » General Settings » License Manager » Requirements lists all the required licensing data.
- Various aesthetic improvements.

Sensor

- Add a new SNMP Sensor, able to monitor networking devices supporting SNMP v1, v2c or v3. One SNMP Sensor license is free.
- The Sniffing Sensor renamed Packet Sensor.
- The Virtual Sensor renamed Sensor Cluster.
- New decoders: IP fragmented, TCP=NULL, TCP+RST, TCP+ACK, TCP+SYNACK, SSDP.
- The BAD decoder matches IP NULL, SYN decoder doesn't match packets/flows with ACK flag set anymore.
- The Packet Sensor is compatible with PF_RING version 6 (Zero Copy, LibZero or DNA license not needed). PF_RING version 5 is not compatible anymore.
- The Packet Sensor supports new capturing engines: System PCAP, Myricom Sniffer10G, SolarCapture (beta).
- The Sensor Cluster can aggregate IP graphs data.
- Packet Sensors listening to the same interface (e.g. for multi-queue load balancing) do not require additional

licenses.

- The Packet Sensor has a new CPU affinity option.
- A new "Manage Interfaces" button in the Flow Sensor Configuration window that provides a quick way to add multiple interfaces.
- The Flow Sensor Configuration window has advanced SNMP options.
- On Flow Sensor's Traffic Direction option. "Mixed" renamed "Auto", "Inbound" renamed "Upstream", "Outbound" renamed "Downstream".

BGP

- Reports » Attacks & Tools » BGP Prefixes renamed BGP Operations.
- Added buttons Reports » Attacks & Tools » BGP Operations » Black Hole, Divert Traffic and Remove All.
- BGP Connections can be configured to announce subnets with configurable masks for BGP peers that do not accept /32 prefixes for null-routing or cloud-based DDoS mitigation services.
- All connections to remote quagga/bgpd services are initialized solely from the Console server.
- Deleting BGP announcements manually works for delayed announcements.
- BGP Announcement Archive displays BGP Connection Role.

Filter

- The Filter renamed Packet Filter.
- A new Flow Filter, able to detect attackers from flow data analyzed by a Flow Sensor.
- A new Filter Cluster, able to cluster multiple Packet Filters and Flow Filters.
- The Filters can use the hardware-based packet filter from Chelsio T4 and T5 10/40 gigabit adapters.
- New Whitelist Templates, for sharing whitelists between Filters. Add them in Configurations » Network & Policy » <+>.
- Support for adding IPv4 and IPv6 subnets in Whitelists and Whitelist Templates.
- The Packet Filter supports new capturing engines: System PCAP, Myricom Sniffer10G, SolarCapture (beta).
- The Packet Filter has a new CPU affinity option.
- The Packet Filter can block private IPs when using the Software Firewall.
- The Filter also works for outgoing attacks.
- The Packet Filter is compatible with PF_RING version 6 (Zero Copy, LibZero or DNA license not needed). PF_RING version 5 is not compatible anymore.