Supported Flow Standards

Technical Specification

Overview

The following document lists all flow standards supported by Flowmon Probe and Flowmon Collector. For each standard, it provides a list of available fields with descriptions and expected input lengths. Where applicable, it also provides details on the content of specific fields or outlines configuration changes necessary for enabling the export/collection of those fields. Fields marked with **Not supported** are not implemented, fields marked with **Not applicable** cannot be, conceptually, supported by the product.

The **Probe Configuration** column in the tables below refers to the configuration options found in **Flowmon Configuration Center – Monitoring Ports – Global Settings – Advanced Settings**. There, you can enable the export of specific IPFIX fields by the probe by checking the corresponding checkboxes and saving your configuration. This configuration can also be applied to each monitoring port separately. Fields will be populated with data only if the required type of network traffic is passing through the probe.

The **Collector Configuration** column in the tables below refers to the configuration options found in **Flowmon Configuration Center – FMC Configuration – Flow Database Fields**. There, you can enable the collection of specific fields by checking the corresponding checkboxes and saving your configuration. A collector will be populated with data only if the required fields are present in flow data sent to the collector.

NetFlow v5/7

Note: In the following table, the value "Exported by default" refers to a monitoring port that has been configured to use **NetFlow v5** as its export protocol.

Name	Description	Input Length (bytes)	Probe Configuration	Collector Configuration
srcaddr	Source IP address.	4	Exported by default	Collected by default
dstaddr	Destination IP address.	4	Exported by default	Collected by default
nexthop	IP address of next hop router.	4	**Not applicable**	Enabled by **Next HOP IP address**
input	SNMP index of input interface.	2	Exported by default	Collected by default
output	SNMP index of output interface.	2	Exported by default **Always set to 0**	Collected by default
dPkts	Packets in the flow.	4	Exported by default	Collected by default
dOctets	Total number of Layer 3 bytes in the packets of the flow.	4	Exported by default	Collected by default
First	SysUptime at start of flow.	4	Exported by default	Collected by default
Last	SysUptime at the time the last packet of the flow was received.	4	Exported by default	Collected by default
srcport	TCP/UDP source port number or equivalent.	2	Exported by default	Collected by default
dstport	TCP/UDP destination port number or equivalent.	2	Exported by default	Collected by default
tcp_flags	Cumulative OR of TCP flags.	1	Exported by default	Collected by default
prot	IP protocol type (for example, TCP = 6; UDP = 17).	1	Exported by default	Collected by default
tos	IP type of service (ToS).	1	Exported by default	Collected by default
src_as	Autonomous system number of the source, either origin or peer.	2	Enabled by **Use autonomous system list**	Collected by default
dst_as	Autonomous system number of the destination, either origin or peer.	2	Enabled by **Use autonomous system list**	Collected by default
src_mask	Source address prefix mask bits.	1	**Not supported**	Enabled by **SRC/DST mask, (dst) TOS, Direction**
dst_mask	Destination address prefix mask bits.	1	**Not supported**	Enabled by **SRC/DST mask, (dst) TOS, Direction**
router_sc (**NetFlow v7 only**)	IP address of the router that is bypassed by the Catalyst 5000 series switch. This is the same address the router uses when it sends NetFlow export packets. This IP address is propagated to all switches bypassing the router through the FCP protocol.	4	**Not applicable**	Collected by default

NetFlow v9

Note: In the following table, the value "Exported by default" refers to a monitoring port that has been configured to use **NetFlow v9** as its export protocol.

ID	Name	Description	Input Length (bytes)	Probe Configuration	Collector Configuration
1	NF9_IN_BYTES	Incoming counter with length N x 8 bits for number of bytes associated with an IP Flow.	4/8	Exported by default	Collected by default
2	NF9_IN_PACKETS	Incoming counter with length N x 8 bits for the number of packets associated with an IP Flow.	4/8	Exported by default	Collected by default
3	NF9_FLOWS_AGGR	Number of flows that were aggregated; default for N is 4.	4/8	**Not supported**	Enabled by **Counter aggregated flows**
4	NF9_IN_PROTOCOL	IP protocol byte.	1	Exported by default	Collected by default
5	NF9_SRC_TOS	Type of Service byte setting when entering the incoming interface.	1	Enabled by **L3/L4 Extended**	Collected by default
6	NF9_TCP_FLAGS	Cumulative of all the TCP flags seen for this flow.	1	Exported by default	Collected by default
7	NF9_L4_SRC_PORT	TCP/UDP source port number i.e.: FTP, Telnet, or equivalent.	2	Exported by default	Collected by default
8	NF9_IPV4_SRC_ADDR	IPv4 source address.	4	Exported by default	Collected by default
9	NF9_SRC_MASK	The number of contiguous bits in the source address subnet mask i.e.: the submask in slash notation.	1	**Not supported**	Enabled by **SRC/DST mask, (dst) TOS, Direction**
10	NF9_INPUT_SNMP	Input interface index; default for N is 2 but higher values could be used.	4/2	Exported by default	Collected by default
11	NF9_L4_DST_PORT	TCP/UDP destination port number i.e.: FTP, Telnet, or equivalent.	2	Exported by default	Collected by default
12	NF9_IPV4_DST_ADDR	IPv4 destination address.	4	Exported by default	Collected by default
13	NF9_DST_MASK	The number of contiguous bits in the destination address subnet mask i.e.: the submask in slash notation.	1	**Not supported**	Enabled by **SRC/DST mask, (dst) TOS, Direction**
14	NF9_OUTPUT_SNMP	Output interface index; default for N is 2 but higher values could be used.	4/2	Exported by default **Configurable per monitoring port**	Collected by default
15	NF9_V4_NEXT_HOP	IPv4 address of next-hop router.	4	**Not applicable**	Enabled by **Next HOP IP address**
16	NF9_SRC_AS	Source BGP autonomous system number where N could be 2 or 4.	4/2	Enabled by **Use autonomous system list**	Collected by default
17	NF9_DST_AS	Destination BGP autonomous system number where N could be 2 or 4.	4/2	Enabled by **Use autonomous system list**	Collected by default

18	NF9_BGP_V4_NEXT_HOP	Next-hop router's IP in the BGP domain.	4	Not applicable	Enabled by **BGP next HOP IP address**
21	NF9_LAST_SWITCHED	System uptime at which the last packet of this flow was switched.	4	Exported by default	Collected by default
22	NF9_FIRST_SWITCHED	System uptime at which the first packet of this flow was switched.	4	Exported by default	Collected by default
23	NF9_OUT_BYTES	Outgoing counter with length N x 8 bits for the number of bytes associated with an IP Flow.	4/8	Not applicable	Enabled by **Counter output bytes**
24	NF9_OUT_PKTS	Outgoing counter with length N x 8 bits for the number of packets associated with an IP Flow.	4/8	Not applicable	Enabled by **Counter output packets**
27	NF9_IPV6_SRC_ADDR	IPv6 Source Address.	16	Exported by default	Collected by default
28	NF9_IPV6_DST_ADDR	IPv6 Destination Address.	16	Exported by default	Collected by default
29	NF9_IPV6_SRC_MASK	Length of the IPv6 source mask in contiguous bits.	1	Not supported	Enabled by **SRC/DST mask, (dst) TOS, Direction**
30	NF9_IPV6_DST_MASK	Length of the IPv6 destination mask in contiguous bits.	1	Not supported	Enabled by **SRC/DST mask, (dst) TOS, Direction**
31	NF9_IPV6_FLOW_LABEL	IPv6 flow label as per RFC 2460 definition.	4	Not supported	Collected by default
32	NF9_ICMP_TYPE	Internet Control Message Protocol (ICMP) packet type; reported as ((ICMP Type*256) + ICMP code).	2	Exported by default	Collected by default
34	NF9_SAMPLING_INTERVAL	When using sampled NetFlow, the rate at which packets are sampled i.e.: a value of 100 indicates that one of every 100 packets is sampled.	4	Exported by default	Collected by default
35	NF9_SAMPLING_ALGORITHM	The type of algorithm used for sampled NetFlow: 0x01 Deterministic Sampling ,0x02 Random Sampling.	1	Exported by default	Collected by default
38	NF9_ENGINE_TYPE	Type of flow switching engine: RP = 0, VIP/Linecard = 1.	1	Not applicable	Collected by default
39	NF9_ENGINE_ID	ID number of the flow switching engine.	1	Not applicable	Collected by default
48	NF9_FLOW_SAMPLER_ID	Identifier shown in "show flow-sampler".	1/2/4	Not supported	Collected by default
49	NF9_FLOW_SAMPLER_MODE	The type of algorithm used for sampling data: 0x02 random sampling. Use in connection with NF9_FLOW_SAMPLER_ID.	1	Not supported	Collected by default
50	NF9_FLOW_SAMPLER_RANDOM _INTERVAL	Packet interval at which to sample. Use in connection with NF9_FLOW_SAMPLER_MODE.	2/4	Not supported	Collected by default
55	NF9_DST_TOS	Type of Service byte setting when exiting outgoing interface.	1	Not applicable	Collected by default
56	NF9_IN_SRC_MAC	Incoming source MAC address.	6	Enabled by **MAC**	Enabled by **In SRC/out DST MAC address**
57	NF9_OUT_DST_MAC	Outgoing destination MAC address.	6	Not applicable	Enabled by **In SRC/out DST MAC address**

58	NF9_SRC_VLAN	Virtual LAN identifier associated with ingress interface.	2	Enabled by **VLAN**	Enabled by **SRC/DST VLAN ID labels**
59	NF9_DST_VLAN	Virtual LAN identifier associated with egress interface.	2	**Not applicable**	Enabled by **SRC/DST VLAN ID labels**
61	NF9_DIRECTION	Flow direction: 0 - ingress flow, 1 - egress flow.	1	**Not applicable**	Enabled by **SRC/DST mask, (dst) TOS, Direction**
62	NF9_V6_NEXT_HOP	IPv6 address of the next-hop router.	16	**Not applicable**	Enabled by **Next HOP IP address**
63	NF9_BPG_V6_NEXT_HOP	Next-hop router in the BGP domain.	16	**Not applicable**	Enabled by **BGP next HOP IP address**
70	NF9_MPLS_LABEL_1	MPLS label at position 1 in the stack. This comprises 20 bits of MPLS label, 3 EXP (experimental) bits and 1 S (end-of-stack) bit.	3	Enabled by **MPLS**	Enabled by **MPLS labels 1-10**
71	NF9_MPLS_LABEL_2	MPLS label at position 2 in the stack. This comprises 20 bits of MPLS label, 3 EXP (experimental) bits and 1 S (end-of-stack) bit.	3	Enabled by **MPLS**	Enabled by **MPLS labels 1-10**
72	NF9_MPLS_LABEL_3	MPLS label at position 3 in the stack. This comprises 20 bits of MPLS label, 3 EXP (experimental) bits and 1 S (end-of-stack) bit.	3	Enabled by **MPLS**	Enabled by **MPLS labels 1-10**
73	NF9_MPLS_LABEL_4	MPLS label at position 4 in the stack. This comprises 20 bits of MPLS label, 3 EXP (experimental) bits and 1 S (end-of-stack) bit.	3	Enabled by **MPLS**	Enabled by **MPLS labels 1-10**
74	NF9_MPLS_LABEL_5	MPLS label at position 5 in the stack. This comprises 20 bits of MPLS label, 3 EXP (experimental) bits and 1 S (end-of-stack) bit.	3	**Not supported**	Enabled by **MPLS labels 1-10**
75	NF9_MPLS_LABEL_6	MPLS label at position 6 in the stack. This comprises 20 bits of MPLS label, 3 EXP (experimental) bits and 1 S (end-of-stack) bit.	3	**Not supported**	Enabled by **MPLS labels 1-10**
76	NF9_MPLS_LABEL_7	MPLS label at position 7 in the stack. This comprises 20 bits of MPLS label, 3 EXP (experimental) bits and 1 S (end-of-stack) bit.	3	**Not supported**	Enabled by **MPLS labels 1-10**
77	NF9_MPLS_LABEL_8	MPLS label at position 8 in the stack. This comprises 20 bits of MPLS label, 3 EXP (experimental) bits and 1 S (end-of-stack) bit.	3	**Not supported**	Enabled by **MPLS labels 1-10**
78	NF9_MPLS_LABEL_9	MPLS label at position 9 in the stack. This comprises 20 bits of MPLS label, 3 EXP (experimental) bits and 1 S (end-of-stack) bit.	3	**Not supported**	Enabled by **MPLS labels 1-10**
79	NF9_MPLS_LABEL_10	MPLS label at position 10 in the stack. This comprises 20 bits of MPLS label, 3 EXP (experimental) bits and 1 S (end-of-stack) bit.	3	**Not supported**	Enabled by **MPLS labels 1-10**
80	NF9_IN_DST_MAC	Incoming destination MAC address.	6	Enabled by **MAC**	Enabled by **In DST/out SRC MAC address**
81	NF9_OUT_SRC_MAC	Outgoing source MAC address.	6	**Not applicable**	Enabled by **In DST/out SRC MAC address**
89	NF9_FORWARDING_STATUS	Forwarding status is encoded on 1 byte with the 2 left bits giving the status and the 6 remaining bits giving the reason code.	1	**Not applicable**	Collected by default
128	NF9_BGP_ADJ_NEXT_AS	The autonomous system (AS) number of the first AS in the AS path to the destination IP address. The path is deduced by looking up the destination IP address of the Flow in the BGP routing information base. If AS path information for this Flow is only available as an unordered AS	4	**Not applicable**	Enabled by **BGP adjacent prev/next AS**

		set (and not as an ordered AS sequence), then the value of this Information Element is 0.			
129	NF9_BGP_ADJ_PREV_AS	The autonomous system (AS) number of the last AS in the AS path from the source IP address. The path is deduced by looking up the source IP address of the Flow in the BGP routing information base. If AS path information for this Flow is only available as an unordered AS set (and not as an ordered AS sequence), then the value of this Information Element is 0. In case of BGP asymmetry, the bgpPrevAdjacentAsNumber might not be able to report the correct value.	4	Not applicable	Enabled by **BGP adjacent prev/next AS**
95	NF9_NBAR2_APP_TAG	8 bits of engine ID, followed by n bits of classification.	4	Not supported	Enabled by **NBAR2 application tag**
85	NF_F_FLOW_BYTES	Running byte counter for a permanent flow.	4/8	Not applicable	Enabled by **NSEL Common block**
148	NF_F_CONN_ID	An identifier of a unique flow for the device.	4	Not applicable	Enabled by **NSEL Common block**
152	NF_F_FLOW_CREATE_TIME_MSEC	The time that the flow was created, which is included in extended flow-teardown events in which the flow-create event was not sent earlier. The flow duration can be determined with the event time for the flow-teardown and flow-create times.	8	Not applicable	Enabled by **NSEL Common block**
153	NF_F_FLOW_END_TIME_MSEC	The time when the flow ended.	8	Not applicable	Collected by default
176	NF_F_ICMP_TYPE	ICMP type value.	1	Not applicable	Enabled by **NSEL Common block**
177	NF_F_ICMP_CODE	ICMP code value.	1	Not applicable	Enabled by **NSEL Common block**
178	NF_F_ICMP_TYPE_IPV6	ICMP IPv6 type value.	1	Not applicable	Enabled by **NSEL Common block**
179	NF_F_ICMP_CODE_IPV6	ICMP IPv6 code value.	1	Not applicable	Enabled by **NSEL Common block**
231	NF_F_FWD_FLOW_DELTA_BYTES	The delta number of bytes from source to destination.	4/8	Not applicable	Enabled by **NSEL Common block**
232	NF_F_REV_FLOW_DELTA_BYTES	The delta number of bytes from destination to source.	4/8	Not applicable	Enabled by **Counter output bytes**
233	NF_F_FW_EVENT84	Indicates a firewall event.	1	Not applicable	Enabled by **NSEL Common block**
323	NF_F_EVENT_TIME_MSEC	The time that the event occurred, which comes from IPFIX. Use 324 for time in microseconds, and 325 for time in nanoseconds.	8	Not applicable	Collected by default
33000	NF_F_INGRESS_ACL_ID	The input ACL that permitted or denied the flow. All ACL IDs are composed of the following three, four-byte values: - Hash value or ID of the ACL name - Hash value, ID, or line of an ACE within the ACL - Hash value or ID of an extended ACE configuration.	12	Not applicable	Enabled by **NSEL ACL ingress/egress acl ID**
33001	NF_F_EGRESS_ACL_ID	The output ACL that permitted or denied a flow.	12	Not applicable	Enabled by **NSEL ACL ingress/egress acl ID**

33002	NF_F_FW_EXT_EVENT	Extended event code. These values provide additional information about the event.	2	Not applicable	Enabled by **NSEL Common block**
40000	NF_F_USERNAME	AAA username.	20/65	Not applicable	Enabled by **NSEL username**
40001	NF_F_XLATE_SRC_ADDR_IPV4	Source IPv4 address.	4	Not applicable	Enabled by **NSEL xlate IPv4 address**
40002	NF_F_XLATE_DST_ADDR_IPV4	Destination IPv4 address.	4	Not applicable	Enabled by **NSEL xlate IPv4 address**
40003	NF_F_XLATE_SRC_PORT	Post NATT Source Transport Port.	2	Not applicable	Enabled by **NSEL xlate ports**
40004	NF_F_XLATE_DST_PORT	Post NATT Destination Transport Port.	2	Not applicable	Enabled by **NSEL xlate ports**
40005	NF_F_FW_EVENT	High-level event code. Values are as follows: 0-Collected by default (ignore),1-Flow created,2-Flow deleted, 3-Flow denied, 4-Flow alert, 5-Flow update.	1	Not applicable	Enabled by **NSEL Common block**
230	NF_N_NAT_EVENT	Indicates a NAT event.	1	Not applicable	Enabled by **NEL Common block**
234	NF_N_INGRESS_VRFID	An unique identifier of the VRF name where the packets of this flow are being received. This identifier is unique per Metering Process.	4	Not applicable	Enabled by **NEL Common block**
225	NF_N_NAT_INSIDE_GLOBAL_IPV4	The definition of this Information Element is identical to the definition of Information Element 'sourceIPv4Address', except that it reports a modified value caused by a NAT middlebox function after the packet passed the Observation Point.	4	Not applicable	Enabled by **NEL global IPv4 address**
226	NF_N_NAT_OUTSIDE_GLOBAL_IPV4	The definition of this Information Element is identical to the definition of Information Element 'destinationIPv4Address', except that it reports a modified value caused by a NAT middlebox function after the packet passed the Observation Point.	4	Not applicable	Enabled by **NEL global IPv4 address**
227	NF_N_POST_NAPT_SRC_PORT	The definition of this Information Element is identical to the definition of Information Element 'sourceTransportPort', except that it reports a modified value caused by a Network Address Port Translation (NAPT) middlebox function after the packet passed the Observation Point.	2	Not applicable	Enabled by **NEL Common block**
228	NF_N_POST_NAPT_DST_PORT	The definition of this Information Element is identical to the definition of Information Element 'destinationTransportPort', except that it reports a modified value caused by a Network Address Port Translation (NAPT) middlebox function after the packet passed the Observation Point.	2	Not applicable	Enabled by **NEL Common block**

IPFIX

Note: In the following table, the value "Exported by default" refers to a monitoring port that has been configured to use **IPFIX** as its export protocol.

PEN	ID	Field	Description	Input Length (bytes)	Probe Configuration	Collector Configuration
-	1	**IPFIX_octetDeltaCount**	The number of octets since the previous report (if any) in incoming packets for this Flow at the Observation Point. The number of octets includes IP header(s) and IP payload.	8/4	Exported by default	Collected by default
-	2	**IPFIX_packetDeltaCount**	The number of incoming packets since the previous report (if any) for this Flow at the Observation point.	8/4	Exported by default	Collected by default
-	4	**IPFIX_protocolIdentifier**	The value of the protocol number in the IP packet header. The protocol number identifies the IP packet payload type. Protocol numbers are defined in the IANA Protocol Numbers registry. In Internet Protocol version 4 (IPv4), this is carried in the Protocol field. In Internet Protocol version 6 (IPv6), this is carried in the Next Header field in the last extension header of the packet.	1	Exported by default	Collected by default
-	5	**IPFIX_ipClassOfService**	For IPv4 packets, this is the value of the TOS field in the IPv4 packet header. For IPv6 packets, this is the value of the Traffic Class field in the IPv6 packet header.	1	Enabled by **L3/L4 Extended**	Collected by default
-	6	**IPFIX_tcpControlBits**	TCP control bits observed for packets of this Flow. The information is encoded in a set of bit fields. For each TCP control bit, there is a bit in this set. A bit is set to 1 if any observed packet of this Flow has the corresponding TCP control bit set to 1. A value of 0 for a bit indicates that the corresponding bit was not set in any of the observed packets of this Flow.	1	Exported by default	Collected by default
-	7	**IPFIX_SourceTransportPort**	The source port identifier in the transport header. For the transport protocols UDP, TCP, and SCTP, this is the source port number given in the respective header. This field MAY also be used for future transport protocols that have 16-bit source port identifiers.	2	Exported by default	Collected by default
-	8	**IPFIX_SourceIPv4Address**	The IPv4 source address in the IP packet header.	4	Exported by default	Collected by default
-	9	**IPFIX_SourceIPv4PrefixLength**	The number of contiguous bits that are relevant in the sourceIPv4Prefix Information Element.	1	Exported by default	Enabled by **SRC/DST mask, (dst) TOS, Direction**
-	10	**IPFIX_ingressInterface**	The index of the IP interface where packets of this Flow are being received. The value matches the	4/2	Exported by default	Collected by default

-			value of the managed object 'ifIndex' as defined in RFC 2863. Note that ifIndex values are not assigned statically to an interface and that the interfaces may be renumbered every time the device's management system is re-initialized, as specified in RFC 2863.			
-	11	IPFIX_DestinationTransportPort	The destination port identifier in the transport header. For the transport protocols UDP, TCP, and SCTP, this is the destination port number given in the respective header. This field MAY also be used for future transport protocols that have 16-bit destination port identifiers.	2	Exported by default	Collected by default
-	12	IPFIX_DestinationIPv4Address	The IPv4 destination address in the IP packet header. Abstract Data Type: ipv4Address.	4	Exported by default	Collected by default
-	13	IPFIX_DestinationIPv4PrefixLength	The number of contiguous bits that are relevant in the destinationIPv4Prefix Information Element.	1	Exported by default	Enabled by **SRC/DST mask, (dst) TOS, Direction**
-	14	IPFIX_egressInterface	The index of the IP interface where packets of this Flow are being sent. The value matches the value of the managed object 'ifIndex' as defined in RFC 2863. Note that ifIndex values are not assigned statically to an interface and that the interfaces may be renumbered every time the device's management system is re-initialized, as specified in RFC 2863.	4/2	Exported by default **Configurable per monitoring port**	Collected by default
-	15	IPFIX_ipNextHopIPv4Address	The IPv4 address of the next IPv4 hop.	4	**Not applicable**	Enabled by **Next HOP IP address**
-	16	IPFIX_bgpSourceAsNumber	The autonomous system (AS) number of the source IP address. If AS path information for this Flow is only available as an unordered AS set (and not as an ordered AS sequence), then the value of this Information Element is 0.	4/2	Enabled by **Use autonomous system list**	Collected by default
-	17	IPFIX_bgpDestinationAsNumber	The autonomous system (AS) number of the destination IP address. If AS path information for this Flow is only available as an unordered AS set (and not as an ordered AS sequence), then the value of this Information Element is 0.	4/2	Enabled by **Use autonomous system list**	Collected by default
-	18	IPFIX_bgpNextHopIPv4Address	The IPv4 address of the next (adjacent) BGP hop.	4	**Not applicable**	Enabled by **BGP next HOP IP address**
-	21	IPFIX_flowEndSysUpTime	The relative timestamp of the last packet of this Flow. It indicates the number of milliseconds since the last (re-)initialization of the IPFIX Device (sysUpTime).	4	**Not supported**	Collected by default
-	22	IPFIX_flowStartSysUpTime	The relative timestamp of the first packet of this Flow. It indicates the number of milliseconds since	4	**Not supported**	Collected by default

-			the last (re-)initialization of the IPFIX Device (sysUpTime).			
-	23	IPFIX_postOctetDeltaCount	The definition of this Information Element is identical to the definition of Information Element 'octetDeltaCount', except that it reports a potentially modified value caused by a middlebox function after the packet passed the Observation Point.	8/4	**Not applicable**	Enabled by **Counter output bytes**
-	24	IPFIX_postPacketDeltaCount	The definition of this Information Element is identical to the definition of Information Element 'packetDeltaCount', except that it reports a potentially modified value caused by a middlebox function after the packet passed the Observation Point.	8/4	**Not applicable**	Enabled by **Counter output packets**
-	27	IPFIX_SourceIPv6Address	The IPv6 source address in the IP packet header.	16	Exported by default	Collected by default
-	28	IPFIX_DestinationIPv6Address	The IPv6 destination address in the IP packet header.	16	Exported by default	Collected by default
-	29	IPFIX_SourceIPv6PrefixLength	The number of contiguous bits that are relevant in the sourceIPv6Prefix Information Element.	1	Exported by default	Enabled by **SRC/DST mask, (dst) TOS, Direction**
-	30	IPFIX_DestinationIPv6PrefixLength	The number of contiguous bits that are relevant in the destinationIPv6Prefix Information Element.	1	Exported by default	Enabled by **SRC/DST mask, (dst) TOS, Direction**
-	31	IPFIX_flowLabelIPv6	The value of the IPv6 Flow Label field in the IP packet header.	4	**Not supported**	Collected by default
-	32	IPFIX_icmpTypeCodeIPv4	Type and Code of the IPv4 ICMP message. The combination of both values is reported as (ICMP type * 256) + ICMP code.	2	Exported by default	Collected by default
-	34	NF9_SAMPLING_INTERVAL	When using sampled NetFlow, the rate at which packets are sampled i.e.: a value of 100 indicates that one of every 100 packets is sampled.	4	Exported by default	Collected by default
-	35	NF9_SAMPLING_ALGORITHM	The type of algorithm used for sampled NetFlow: 0x01 Deterministic Sampling, 0x02 Random Sampling.	1	Exported by default	Collected by default
-	48	NF9_FLOW_SAMPLER_ID	Identifier shown in "show flow-sampler".	1/2/4	**Not supported**	Collected by default
-	49	NF9_FLOW_SAMPLER_MODE	The type of algorithm used for sampling data: 0x02 random sampling. Use in connection with NF9_FLOW_SAMPLER_ID.	4	**Not supported**	Collected by default
-	50	NF9_FLOW_SAMPLER_RANDOM_INTERVAL	Packet interval at which to sample. Use in connection with NF9_FLOW_SAMPLER_MODE.	2/4	**Not supported**	Collected by default
-	55	IPFIX_postIpClassOfService	The definition of this Information Element is identical to the definition of Information Element 'ipClassOfService', except that it reports a potentially modified value caused by a middlebox function after the packet passed the Observation Point.	1	**Not applicable**	Enabled by **SRC/DST mask, (dst) TOS, Direction**

-	56	IPFIX_SourceMacAddress	The IEEE 802 source MAC address field.	6	Enabled by **MAC**	Enabled by **In SRC/out DST MAC address**
-	57	IPFIX_postDestinationMacAddress	The definition of this Information Element is identical to the definition of Information Element 'destinationMacAddress', except that it reports a potentially modified value caused by a middlebox function after the packet passed the Observation Point.	6	**Not applicable**	Enabled by **In SRC/out DST MAC address**
-	58	IPFIX_vlanId	The IEEE 802.1Q VLAN identifier (VID) extracted from the Tag Control Information field that was attached to the IP packet.	2	Enabled by **VLAN**	Enabled by **SRC/DST VLAN ID labels**
-	59	IPFIX_postVlanId	The definition of this Information Element is identical to the definition of Information Element 'vlanId', except that it reports a potentially modified value caused by a middlebox function after the packet passed the Observation Point.	2	**Not applicable**	Enabled by **SRC/DST VLAN ID labels**
-	61	IPFIX_flowDirection	The direction of the Flow observed at the Observation Point. There are only two values defined.	1	**Not applicable**	Enabled by **SRC/DST mask, (dst) TOS, Direction**
-	62	IPFIX_ipNextHopIPv6Address	The IPv6 address of the next IPv6 hop.	16	**Not applicable**	Enabled by **Next HOP IP address**
-	63	IPFIX_bgpNextHopIPv6Address	The IPv6 address of the next (adjacent) BGP hop.	16	**Not applicable**	Enabled by **BGP next HOP IP address**
-	70	IPFIX_mplsTopLabelStackSection	The Label, Exp, and S fields from the top MPLS label stack entry, i.e., from the last label that was pushed. The size of this Information Element is 3 octets.	3	Enabled by **MPLS**	Enabled by **MPLS labels 1-10**
-	71	IPFIX_mplsLabelStackSection2	The Label, Exp, and S fields from the label stack entry that was pushed immediately before the label stack entry that would be reported by mplsTopLabelStackSection. See the definition of mplsTopLabelStackSection for further details. The size of this Information Element is 3 octets.	3	Enabled by **MPLS**	Enabled by **MPLS labels 1-10**
-	72	IPFIX_mplsLabelStackSection3	The Label, Exp, and S fields from the label stack entry that was pushed immediately before the label stack entry that would be reported by mplsLabelStackSection2. See the definition of mplsTopLabelStackSection for further details. The size of this Information Element is 3 octets.	3	Enabled by **MPLS**	Enabled by **MPLS labels 1-10**
-	73	IPFIX_mplsLabelStackSection4	The Label, Exp, and S fields from the label stack entry that was pushed immediately before the label stack entry that would be reported by mplsLabelStackSection3. See the definition of	3	Enabled by **MPLS**	Enabled by **MPLS labels 1-10**

-			mplsTopLabelStackSection for further details. The size of this Information Element is 3 octets.			
-	74	IPFIX_mplsLabelStackSection5	The Label, Exp, and S fields from the label stack entry that was pushed immediately before the label stack entry that would be reported by mplsLabelStackSection4. See the definition of mplsTopLabelStackSection for further details. The size of this Information Element is 3 octets.	3	Not supported	Enabled by **MPLS labels 1-10**
-	75	IPFIX_mplsLabelStackSection6	The Label, Exp, and S fields from the label stack entry that was pushed immediately before the label stack entry that would be reported by mplsLabelStackSection5. See the definition of mplsTopLabelStackSection for further details. The size of this Information Element is 3 octets.	3	Not supported	Enabled by **MPLS labels 1-10**
-	76	IPFIX_mplsLabelStackSection7	The Label, Exp, and S fields from the label stack entry that was pushed immediately before the label stack entry that would be reported by mplsLabelStackSection6. See the definition of mplsTopLabelStackSection for further details. The size of this Information Element is 3 octets.	3	Not supported	Enabled by **MPLS labels 1-10**
-	77	IPFIX_mplsLabelStackSection8	The Label, Exp, and S fields from the label stack entry that was pushed immediately before the label stack entry that would be reported by mplsLabelStackSection7. See the definition of mplsTopLabelStackSection for further details. The size of this Information Element is 3 octets.	3	Not supported	Enabled by **MPLS labels 1-10**
-	78	IPFIX_mplsLabelStackSection9	The Label, Exp, and S fields from the label stack entry that was pushed immediately before the label stack entry that would be reported by mplsLabelStackSection8. See the definition of mplsTopLabelStackSection for further details. The size of this Information Element is 3 octets.	3	Not supported	Enabled by **MPLS labels 1-10**
-	79	IPFIX_mplsLabelStackSection10	The Label, Exp, and S fields from the label stack entry that was pushed immediately before the label stack entry that would be reported by mplsLabelStackSection9. See the definition of mplsTopLabelStackSection for further details. The size of this Information Element is 3 octets.	3	Not supported	Enabled by **MPLS labels 1-10**
-	80	IPFIX_DestinationMacAddress	The IEEE 802 destination MAC address field.	6	Enabled by **MAC**	Enabled by **In DST/out SRC MAC address**
-	81	IPFIX_postSourceMacAddress	The definition of this Information Element is identical to the definition of Information Element 'sourceMacAddress', except that it reports a potentially modified value caused by a middlebox function after the packet passed the Observation Point.	6	Not applicable	Enabled by **In DST/out SRC MAC address**

-	85	IPFIX_octetTotalCount	The total number of octets in incoming packets for this Flow at the Observation Point since the Metering Process (re-)initialization for this Observation Point. The number of octets includes IP header(s) and IP payload.	8/4	**Not supported**	Collected by default
-	86	IPFIX_packetTotalCount	The total number of incoming packets for this Flow at the Observation Point since the Metering Process (re-)initialization for this Observation Point.	8/4	**Not supported**	Collected by default
-	90	IPFIX_mplsVpnRouteDistinguisher	The value of the VPN route distinguisher of a corresponding entry in a VPN routing and forwarding table. Route distinguisher ensures that the same address can be used in several different MPLS VPNs and that it is possible for BGP to carry several completely different routes to that address, one for each VPN. According to RFC 4364, the size of mplsVpnRouteDistinguisher is 8 octets. However, in RFC 4382 an octet string with flexible length was chosen for representing a VPN route distinguisher by object MplsL3VpnRouteDistinguisher. This choice was made in order to be open to future changes of the size. This idea was adopted when choosing octetArray as an abstract data type for this Information Element. The maximum length of this Information Element is 256 octets.	8	**Not applicable**	Enabled by **MPLS VPN Route Distinguisher**
-	95	IPFIX_applicationId	Specifies an Application ID.	4	Enabled by **NBAR2**	Enabled by **NBAR2 application tag**
-	128	IPFIX_bgpAdjNextAs	The autonomous system (AS) number of the first AS in the AS path to the destination IP address. The path is deduced by looking up the destination IP address of the Flow in the BGP routing information base. If AS path information for this Flow is only available as an unordered AS set (and not as an ordered AS sequence), then the value of this Information Element is 0.	**4**	**Not applicable**	Enabled by **BGP adjacent prev/next AS**
-	129	IPFIX_bgpAdjPrevAs	The autonomous system (AS) number of the last AS in the AS path from the source IP address. The path is deduced by looking up the source IP address of the Flow in the BGP routing information base. If AS path information for this Flow is only available as an unordered AS set (and not as an ordered AS sequence), then the value of this Information Element is 0. In case of BGP asymmetry, the bgpPrevAdjacentAsNumber might not be able to report the correct value.	4	**Not applicable**	Enabled by **BGP adjacent prev/next AS**
-	150	IPFIX_flowStartSeconds	The absolute timestamp of the first packet of this Flow.	4	**Not supported**	Collected by default

-	151	IPFIX_flowEndSeconds	The absolute timestamp of the last packet of this Flow.	4	**Not supported**	Collected by default
-	152	IPFIX_flowStartMilliseconds	The absolute timestamp of the first packet of this Flow.	8	Exported by default	Collected by default
-	153	IPFIX_flowEndMilliseconds	The absolute timestamp of the last packet of this Flow.	8	Exported by default	Collected by default
-	186	IPFIX_tcpWindowSize	The window field in the TCP header. If the TCP window scale is supported, then TCP window scale must be known to fully interpret the value of this information.	2	Enabled by **L3/L4 Extended**	Enabled by **L3/L4 extended fields**
-	302	IPFIX_selectorId	A unique identifier of the application for a specific Classification Engine ID. Note that the Selector ID length varies depending on the Classification Engine ID.	8	**Not supported**	Collected by default
-	304	IPFIX_selectorAlgorithm	This Information Element identifies the Intermediate Flow Selection Process technique (e.g., Filtering, Sampling) that is applied by the Intermediate Flow Selection Process. Most of these techniques have parameters. Its configuration parameter(s) MUST be clearly specified.	2	**Not supported**	Collected by default
-	305	IPFIX_samplingPacketInterval	Interval of packet sampling.	4	**Not supported**	Collected by default
-	306	IPFIX_samplingPakcetSpace	Packet space sampling.	4	**Not supported**	Collected by default
39499	1	IPFIX_INVEA_HOST	HTTP Hostname string (last 32 bytes).	32	Enabled by **HTTP**	Truncated to 31 bytes; Enabled by **HTTP Hostname**
39499	2	IPFIX_INVEA_URL	HTTP URL string (first 64 bytes).	64	Enabled by **HTTP**	Enabled by **HTTP URL**
39499	4	IPFIX_INVEA_HTTP_METHOD_ID	HTTP Method	2	Enabled by **HTTP**	Enabled by **HTTP method and result**
39499	12	IPFIX_INVEA_HTTP_STATUS_CODE	HTTP Status Code	2	Enabled by **HTTP**	Enabled by **HTTP method and result**
39499	22	IPFIX_INVEA_USER_AGENT_OS	Identifies version of host operating system.	2	Enabled by **HTTP**	Enabled by **HTTP OS & Application info**
39499	23	IPFIX_INVEA_USER_AGENT_OS_MAJ	Identifies the major version of the used host operating system.	2	Enabled by **HTTP**	Enabled by **HTTP OS & Application info**
39499	24	IPFIX_INVEA_USER_AGENT_OS_MIN	Identifies the minor version of the used host operating system.	2	Enabled by **HTTP**	Enabled by **HTTP OS & Application info**
39499	25	IPFIX_INVEA_USER_AGENT_OS_BLD	Identifies build version of used host operating system.	2	Enabled by **HTTP**	Enabled by

						HTTP OS & Application info
39499	26	IPFIX_INVEA_USER_AGENT_APP	Identifies versions of used applications.	2	Enabled by **HTTP**	Enabled by **HTTP OS & Application info**
39499	27	IPFIX_INVEA_USER_AGENT_APP_MAJ	Identifies the major version of used applications.	2	Enabled by **HTTP**	Enabled by **HTTP OS & Application info**
39499	28	IPFIX_INVEA_USER_AGENT_APP_MIN	Identifies the minimal version of used applications.	2	Enabled by **HTTP**	Enabled by **HTTP OS & Application info**
39499	29	IPFIX_INVEA_USER_AGENT_APP_BLD	Identifies build version of application build.	2	Enabled by **HTTP**	Enabled by **HTTP OS & Application info**
39499	33	IPFIX_INVEA_SIP_CALL_ID	The value of the SIP Call-ID.	64	Enabled by **VoIP** or **Extended VoIP**	Enabled by **VoIP SIP basic**
39499	34	IPFIX_INVEA_SIP_CALLING_PARTY	Parameter that distinguishes the station used to originate a call.	48	Enabled by **VoIP** or **Extended VoIP**	Enabled by **VoIP SIP basic**
39499	35	IPFIX_INVEA_SIP_CALLED_PARTY	The addr-spec URI, including any URI parameters, of the SIP P-Called-Party-ID header, as a UTF-8 string, escaped according to SIP rules as received by the metering process.	48	Enabled by **VoIP** or **Extended VoIP**	Enabled by **VoIP SIP basic**
39499	36	IPFIX_INVEA_SIP_VIA	The value of the first/top-most Via header as a UTF-8 string, escaped according to SIP rules as received by the metering process.	48	Enabled by **VoIP** or **Extended VoIP**	Enabled by **VoIP SIP basic**
39499	32	IPFIX_INVEA_VOIP_PACKET_TYPE	Type of VoIP packets.	1	Enabled by **VoIP** or **Extended VoIP**	Enabled by **VoIP SIP advanced**
39499	37	IPFIX_INVEA_SIP_INVITE_RINGING_TIME	SIP Ringing Time (microsec timestamp).	8	Enabled by **VoIP** or **Extended VoIP**	Enabled by **VoIP SIP advanced**
39499	38	IPFIX_INVEA_SIP_OK_TIME	SIP OK Time (microsec timestamp).	8	Enabled by **VoIP** or **Extended VoIP**	Enabled by **VoIP SIP advanced**
39499	39	IPFIX_INVEA_SIP_BYE_TIME	SIP Bye Time (microsec timestamp).	8	Enabled by **VoIP** or **Extended VoIP**	Enabled by **VoIP SIP advanced**
39499	40	IPFIX_INVEA_SIP_RTP_IP4	IPv4 Real time protocol.	4	Enabled by **VoIP** or **Extended VoIP**	Enabled by **VoIP SIP advanced**
39499	41	IPFIX_INVEA_SIP_RTP_IP6	IPv6 Real time protocol.	16	Enabled by **VoIP** or **Extended VoIP**	Enabled by **VoIP SIP advanced**
39499	42	IPFIX_INVEA_SIP_RTP_AUDIO	Real time protocol audio quality.	2	Enabled by	Enabled by

					VoIP or Extended VoIP	VoIP SIP advanced
39499	43	IPFIX_INVEA_SIP_RTP_VIDEO	Real time protocol video quality.	2	Enabled by **VoIP or Extended VoIP**	Enabled by **VoIP SIP advanced**
39499	44	IPFIX_INVEA_SIP_STATS	SIP statistic.	8	Enabled by **VoIP or Extended VoIP**	Enabled by **VoIP SIP advanced**
39499	45	IPFIX_INVEA_RTP_CODEC	Type of codec.	1	Enabled by **Extended VoIP**	Enabled by **VoIP RTP**
39499	46	IPFIX_INVEA_RTP_JITTER	Jitter.	4	Enabled by **Extended VoIP**	Enabled by **VoIP RTP**
39499	47	IPFIX_INVEA_RTCP_LOST	Packet loss.	4	Enabled by **Extended VoIP**	Enabled by **VoIP RTP**
39499	48	IPFIX_INVEA_RTCP_PACKETS	Quantity of packets.	8	Enabled by **Extended VoIP**	Enabled by **VoIP RTP**
39499	49	IPFIX_INVEA_RTCP_OCTETS	Quantity of octets.	8	Enabled by **Extended VoIP**	Enabled by **VoIP RTP**
39499	50	IPFIX_INVEA_RTCP_SOURCE_COUNT	Real time control protocol source count.	1	Enabled by **Extended VoIP**	Enabled by **VoIP RTP**
39499	69	IPFIX_INVEA_NPM_RTT	Round trip time.	4	Enabled by **NPM**	Enabled by **NPM Basic Metrics**
39499	70	IPFIX_INVEA_NPM_SRT	Server response time.	4	Enabled by **NPM**	Enabled by **NPM Basic Metrics**
39499	61	IPFIX_INVEA_NPM_JITTER_DEV	Deviation jitter.	4	Enabled by **Extended NPM**	Enabled by **NPM Basic Metrics**
39499	62	IPFIX_INVEA_NPM_JITTER_AVG	Average jitter.	4	Enabled by **Extended NPM**	Enabled by **NPM Basic Metrics**
39499	63	IPFIX_INVEA_NPM_JITTER_MIN	Minimal jitter.	4	Enabled by **Extended NPM**	Enabled by **NPM Basic Metrics**
39499	64	IPFIX_INVEA_NPM_JITTER_MAX	Maximal jitter.	4	Enabled by **Extended NPM**	Enabled by **NPM Basic Metrics**
39499	65	IPFIX_INVEA_NPM_DELAY_DEV	Deviation time between packets.	4	Enabled by **Extended NPM**	Enabled by **NPM Basic Metrics**
39499	66	IPFIX_INVEA_NPM_DELAY_AVG	Average time between packets.	4	Enabled by **Extended NPM**	Enabled by **NPM Basic Metrics**
39499	67	IPFIX_INVEA_NPM_DELAY_MIN	Minimal time between packets.	4	Enabled by **Extended NPM**	Enabled by **NPM Basic Metrics**
39499	68	IPFIX_INVEA_NPM_DELAY_MAX	Maximal time between packets.	4	Enabled by **Extended NPM**	Enabled by **NPM Basic Metrics**
39499	71	IPFIX_INVEA_NPM_RETRANSMISSION	Retransmissions.	4	Enabled by **NPM**	Enabled by **NPM Retransmission & Out of Order**
39499	72	IPFIX_INVEA_NPM_OUT_OF_ORDER	Out of order packets.	4	Enabled by **NPM**	Enabled by

						NPM Retransmission & Out of Order
39499	80	**IPFIX_INVEA_TCP_SYN_SIZE**	TCP SYN Size	1	Enabled by **L3/L4 Extended**	Enabled by **L3/L4 extended fields**
39499	81	**IPFIX_INVEA_TCP_SYN_TTL**	TCP SYN TTL	1	Enabled by **L3/L4 Extended**	Enabled by **L3/L4 extended fields**
39499	110	**IPFIX_INVEA_DNS_ID**	DNS Identifier.	2	Enabled by **DNS**	Enabled by **DNS Fields**
39499	111	**IPFIX_INVEA_DNS_FLAGS_CODES**	DNS Flags (includes Operation Code, Response Code, Query/Response flag).	2	Enabled by **DNS**	Enabled by **DNS Fields**
39499	112	**IPFIX_INVEA_DNS_QUESTION_COUNT**	DNS Questions Count.	2	Enabled by **DNS**	Enabled by **DNS Fields**
39499	113	**IPFIX_INVEA_DNS_ANSWREC_COUNT**	DNS Answers Count.	2	Enabled by **DNS**	Enabled by **DNS Fields**
39499	114	**IPFIX_INVEA_DNS_AUTHREC_COUNT**	DNS Authority Count.	2	Enabled by **DNS**	Enabled by **DNS Fields**
39499	115	**IPFIX_INVEA_DNS_ADDTREC_COUNT**	DNS Additional Count.	2	Enabled by **DNS**	Enabled by **DNS Fields**
39499	116	**IPFIX_INVEA_DNS_CRR_NAME**	DNS Response Name.	64	Enabled by **DNS**	Enabled by **DNS Fields**
39499	117	**IPFIX_INVEA_DNS_CRR_TYPE**	DNS Response Type.	2	Enabled by **DNS**	Enabled by **DNS Fields**
39499	118	**IPFIX_INVEA_DNS_CRR_CLASS**	DNS Response Class.	2	Enabled by **DNS**	Enabled by **DNS Fields**
39499	119	**IPFIX_INVEA_DNS_CRR_TTL**	DNS Response TTL.	4	Enabled by **DNS**	Enabled by **DNS Fields**
39499	120	**IPFIX_INVEA_DNS_CRR_RDATA**	DNS Response Data.	64	Enabled by **DNS**	Enabled by **DNS Fields**
39499	121	**IPFIX_INVEA_DNS_QNAME**	DNS Question Name.	64	Enabled by **DNS**	Enabled by **DNS Fields**
39499	122	**IPFIX_INVEA_DNS_QTYPE**	DNS Question Type.	2	Enabled by **DNS**	Enabled by **DNS Fields**
39499	123	**IPFIX_INVEA_DNS_QCLASS**	DNS Question Class.	2	Enabled by **DNS**	Enabled by **DNS Fields**
39499	124	**IPFIX_INVEA_DNS_CRR_RDATA_LEN**	DNS Response Data Length.	2	Enabled by **DNS**	Enabled by **DNS Fields**
39499	150	**IPFIX_INVEA_SMB_CMD**	Samba operation code version 2	4	Enabled by **Samba**	Enabled by **SMB Fields**
39499	151	**IPFIX_INVEA_SMB_TREE**	Samba tree structure	128	Enabled by **Samba**	Enabled by **SMB Fields**
39499	152	**IPFIX_INVEA_SMB_FILE**	Samba file name	128	Enabled by **Samba**	Enabled by **SMB Fields**

39499	153	IPFIX_INVEA_SMB_FILE_TYPE	Samba file type	1	Enabled by **Samba**	Enabled by **SMB Fields**
39499	154	IPFIX_INVEA_SMB_OP	Samba file operation type	1	Enabled by **Samba**	Enabled by **SMB Fields**
39499	190	IPFIX_INVEA_ARP_HRD	ARP Hardware Type	2	Exported by default	Collected by default
39499	191	IPFIX_INVEA_ARP_OP	ARP Operation Code	2	Exported by default	Collected by default
39499	200	IPFIX_INVEA_DHCP_OFFERED_IP	DHCP offered IP address	4	Enabled by **DHCP**	Enabled by **DHCP Fields**
39499	201	IPFIX_INVEA_DHCP_HOST_MAC_ADDR	DHCP MAC address of host	6	Enabled by **DHCP**	Enabled by **DHCP Fields**
39499	202	IPFIX_INVEA_DHCP_TYPE	DHCP message type	1	Enabled by **DHCP**	Enabled by **DHCP Fields**
39499	203	IPFIX_INVEA_DHCP_IP_LEASE_TIME	DHCP IP address lease time	4	Enabled by **DHCP**	Enabled by **DHCP Fields**
39499	204	IPFIX_INVEA_DHCP_SERVER_IP	DCP server IP address	4	Enabled by **DHCP**	Enabled by **DHCP Fields**
39499	205	IPFIX_INVEA_DHCP_DOMAIN_NAME	DHCP server domain name	32	Enabled by **DHCP**	Enabled by **DHCP Fields**
39499	206	IPFIX_INVEA_DHCP_HOST_NAME	DHCP hostname	32	Enabled by **DHCP**	Enabled by **DHCP Fields**
39499	207	IPFIX_INVEA_DHCP_IP_REQUEST	DHCP requested IP address	4	Enabled by **DHCP**	Enabled by **DHCP Fields**
39499	210	IPFIX_INVEA_TDS_REQUEST_TYPE	MSSQL (TDS) request type	1	Enabled by **MSSQL**	Enabled by **MSSQL Fields**
39499	211	IPFIX_INVEA_TDS_VERSION	MSSQL (TDS) version (protocol type)	4	Enabled by **MSSQL**	Enabled by **MSSQL Fields**
39499	212	IPFIX_INVEA_TDS_CLIENT_VERSION	MSSQL (TDS) client version	4	Enabled by **MSSQL**	Enabled by **MSSQL Fields**
39499	213	IPFIX_INVEA_TDS_SERVER_VERSION	MSSQL (TDS) server version	4	Enabled by **MSSQL**	Enabled by **MSSQL Fields**
39499	214	IPFIX_INVEA_TDS_DATABASE	MSSQL (TDS) database context	64	Enabled by **MSSQL**	Enabled by **MSSQL Fields**
39499	215	IPFIX_INVEA_TDS_USERNAME	MSSQL (TDS) username	64	Enabled by **MSSQL**	Enabled by **MSSQL Fields**
39499	216	IPFIX_INVEA_TDS_HOSTNAME	MSSQL (TDS) hostname	64	Enabled by **MSSQL**	Enabled by **MSSQL Fields**
39499	217	IPFIX_INVEA_TDS_RESPONSE_TYPE	MSSQL (TDS) response type	1	Enabled by **MSSQL**	Enabled by **MSSQL Extended fields**
39499	218	IPFIX_INVEA_TDS_TOKEN	MSSQL (TDS) token of response	1	Enabled by **MSSQL**	Enabled by **MSSQL Extended fields**
39499	219	IPFIX_INVEA_TDS_TMR_TYPE	MSSQL (TDS) transaction manager request type	2	Enabled by **MSSQL**	Enabled by

						MSSQL Extended fields
39499	220	IPFIX_INVEA_TDS_ERROR_CODE	MSSQL (TDS) error code	4	Enabled by **MSSQL**	Enabled by **MSSQL Extended fields**
39499	221	IPFIX_INVEA_TDS_ENVCHANGE_TYPE	MSSQL (TDS) environment change type	1	Enabled by **MSSQL**	Enabled by **MSSQL Extended fields**
39499	222	IPFIX_INVEA_TDS_SQL_QUERY	MSSQL (TDS) SQL query	64	Enabled by **MSSQL**	Enabled by **MSSQL Extended fields**
39499	223	IPFIX_INVEA_TDS_RPC_NAME	MSSQL (TDS) remote procedure name	64	Enabled by **MSSQL**	Enabled by **MSSQL Extended fields**
39499	224	IPFIX_INVEA_TDS_SERVER_NAME	MSSQL (TDS) server name	64	Enabled by **MSSQL**	Enabled by **MSSQL Extended fields**
39499	250	IPFIX_INVEA_SMTP_EHLO_HELO	SMTP HELO content	64	Enabled by **Email**	Enabled by **E-mail Fields**
39499	251	IPFIX_INVEA_SMTP_MAIL_FROM	SMTP FROM content	64	Enabled by **Email**	Enabled by **E-mail Fields**
39499	252	IPFIX_INVEA_MAIL_USERNAME	SMTP username	64	Enabled by **Email**	Enabled by **E-mail Fields**
39499	253	IPFIX_INVEA_MAIL_FAILED_AUTHENTICATION_COUNT	SMTP failed authentication counter	1	Enabled by **Email**	Enabled by **E-mail Fields**
39499	254	IPFIX_INVEA_MAIL_IS_ENCRYPTED	SMTP TLS flag	1	Enabled by **Email**	Enabled by **E-mail Fields**
39499	290	IPFIX_FLOWMON_MYSQL_PROTOCOL_VERSION	MySQL protocol version	1	Enabled by **MySQL**	Enabled by **MySQL fields**
39499	291	IPFIX_FLOWMON_MYSQL_SERVER_VERSION	MySQL server version	64	Enabled by **MySQL**	Enabled by **MySQL fields**
39499	292	IPFIX_FLOWMON_MYSQL_USER_AUTH_STATUS	MySQL user authentication status	1	Enabled by **MySQL**	Enabled by **MySQL fields**
39499	293	IPFIX_FLOWMON_MYSQL_USERNAME	MySQL username	64	Enabled by **MySQL**	Enabled by **MySQL fields**
39499	294	IPFIX_FLOWMON_MYSQL_AUTH_METHOD	MySQL authentication method	64	Enabled by **MySQL**	Enabled by **MySQL fields**
39499	295	IPFIX_FLOWMON_MYSQL_DATABASE	MySQL database	64	Enabled by **MySQL**	Enabled by **MySQL fields**
39499	296	IPFIX_FLOWMON_MYSQL_CPBLT_SERVER	MySQL server capabilities	4	Enabled by **MySQL**	Enabled by **MySQL extended fields**
39499	297	IPFIX_FLOWMON_MYSQL_CPBLT_CLIENT	MySQL client capabilities	4	Enabled by **MySQL**	Enabled by **MySQL extended fields**

39499	298	IPFIX_FLOWMON_MYSQL_ERROR	MySQL error code	2	Enabled by **MySQL**	Enabled by **MySQL extended fields**
39499	299	IPFIX_FLOWMON_MYSQL_COMMAND	MySQL command	1	Enabled by **MySQL**	Enabled by **MySQL extended fields**
39499	300	IPFIX_FLOWMON_MYSQL_SQL_QUERY	MySQL SQL query	64	Enabled by **MySQL**	Enabled by **MySQL extended fields**
39499	310	IPFIX_FLOWMON_PGSQL_PROTOCOL_VERSION	PostgreSQL protocol version	4	Enabled by **PostgreSQL**	Enabled by **PostgreSQL fields**
39499	311	IPFIX_FLOWMON_PGSQL_SERVER_VERSION	PostgreSQL server version	4	Enabled by **PostgreSQL**	Enabled by **PostgreSQL fields**
39499	312	IPFIX_FLOWMON_PGSQL_AUTH_METHOD	PostgreSQL authentication method	1	Enabled by **PostgreSQL**	Enabled by **PostgreSQL fields**
39499	313	IPFIX_FLOWMON_PGSQL_USERNAME	PostgreSQL username	64	Enabled by **PostgreSQL**	Enabled by **PostgreSQL fields**
39499	314	IPFIX_FLOWMON_PGSQL_DATABASE	PostgreSQL database	64	Enabled by **PostgreSQL**	Enabled by **PostgreSQL fields**
39499	315	IPFIX_FLOWMON_PGSQL_ERROR_SQLSTATE	PostgreSQL error code	5	Enabled by **PostgreSQL**	Enabled by **PostgreSQL extended fields**
39499	316	IPFIX_FLOWMON_PGSQL_ERROR_SEVERITY	PostgreSQL error severity	1	Enabled by **PostgreSQL**	Enabled by **PostgreSQL extended fields**
39499	317	IPFIX_FLOWMON_PGSQL_SQL_QUERY	PostgreSQL SQL query	64	Enabled by **PostgreSQL**	Enabled by **PostgreSQL extended fields**
39499	318	IPFIX_FLOWMON_PGSQL_MSG_TYPE_CLIENT	PostgreSQL client message type	4	Enabled by **PostgreSQL**	Enabled by **PostgreSQL extended fields**
39499	319	IPFIX_FLOWMON_PGSQL_MSG_TYPE_SERVER	PostgreSQL server message type	4	Enabled by **PostgreSQL**	Enabled by **PostgreSQL extended fields**
39499	330	IPFIX_FLOWMON_TLS_CONTENT_TYPE	TLS content type	1	Enabled by **TLS main**	Enabled by **TLS main fields**
39499	331	IPFIX_FLOWMON_TLS_HANDSHAKE_TYPE	TLS handshake type	4	Enabled by **TLS main**	Enabled by **TLS main fields**
39499	332	IPFIX_FLOWMON_TLS_SETUP_TIME	TLS setup time	8	Enabled by **TLS main**	Enabled by **TLS main fields**
39499	333	IPFIX_FLOWMON_TLS_SERVER_VERSION	TLS server version	2	Enabled by **TLS main**	Enabled by **TLS main fields**
39499	334	IPFIX_FLOWMON_TLS_SERVER_RANDOM	TLS server random ID	32	Enabled by **TLS main**	Enabled by **TLS main fields**
39499	335	IPFIX_FLOWMON_TLS_SERVER_SESSION	TLS server session ID	32	Enabled by **TLS main**	Enabled by **TLS main fields**

39499	336	IPFIX_FLOWMON_TLS_CIPHER_SUITE	TLS cipher suite	2	Enabled by **TLS main**	Enabled by **TLS main fields**
39499	337	IPFIX_FLOWMON_TLS_ALPN	TLS application layer protocol negotiation	19	Enabled by **TLS main**	Enabled by **TLS main fields**
39499	338	IPFIX_FLOWMON_TLS_SNI	TLS server name indication	64	Enabled by **TLS main**	Enabled by **TLS main fields**
39499	339	IPFIX_FLOWMON_TLS_SNI_LENGTH	TLS server name indication length	2	Enabled by **TLS main**	Enabled by **TLS main fields**
39499	340	IPFIX_FLOWMON_TLS_CLIENT_VERSION	TLS client version	2	Enabled by **TLS client**	Enabled by **TLS client fields**
39499	341	IPFIX_FLOWMON_TLS_CIPHER_SUITES	TLS cipher suites	16	Enabled by **TLS client**	Enabled by **TLS client fields**
39499	342	IPFIX_FLOWMON_TLS_CLIENT_RANDOM	TLS client random ID	32	Enabled by **TLS client**	Enabled by **TLS client fields**
39499	343	IPFIX_FLOWMON_TLS_CLIENT_SESSION	TLS client session ID	32	Enabled by **TLS client**	Enabled by **TLS client fields**
39499	344	IPFIX_FLOWMON_TLS_EXTENSION_TYPES	TLS extension types	56	Enabled by **TLS client**	Enabled by **TLS client fields**
39499	345	IPFIX_FLOWMON_TLS_EXTENSION_LENGTHS	TLS extension lengths	56	Enabled by **TLS client**	Enabled by **TLS client fields**
39499	346	IPFIX_FLOWMON_TLS_ELLIPTIC_CURVES	TLS elliptic curves	16	Enabled by **TLS client**	Enabled by **TLS client fields**
39499	347	IPFIX_FLOWMON_TLS_EC_POINT_FORMATS	TLS elliptic curves point formats	4	Enabled by **TLS client**	Enabled by **TLS client fields**
39499	348	IPFIX_FLOWMON_TLS_CLIENT_KEY_LENGTH	TLS client key length	4	Enabled by **TLS client**	Enabled by **TLS client fields**
39499	349	IPFIX_FLOWMON_TLS_ISSUER_CN	TLS certificate issuer common name	64	Enabled by **TLS certificate**	Enabled by **TLS certificate fields**
39499	350	IPFIX_FLOWMON_TLS_SUBJECT_CN	TLS subject common name	64	Enabled by **TLS certificate**	Enabled by **TLS certificate fields**
39499	351	IPFIX_FLOWMON_TLS_SUBJECT_ON	TLS subject organization name	64	Enabled by **TLS certificate**	Enabled by **TLS certificate fields**
39499	352	IPFIX_FLOWMON_TLS_VALIDITY_NOT_BEFORE	TLS certificate validity since	8	Enabled by **TLS certificate**	Enabled by **TLS certificate fields**
39499	353	IPFIX_FLOWMON_TLS_VALIDITY_NOT_AFTER	TLS certificate validity until	8	Enabled by **TLS certificate**	Enabled by **TLS certificate fields**
39499	354	IPFIX_FLOWMON_TLS_SIGNATURE_ALG	TLS signature algorithm	2	Enabled by **TLS certificate**	Enabled by **TLS certificate fields**
39499	355	IPFIX_FLOWMON_TLS_PUBLIC_KEY_ALG	TLS public key algorithm	2	Enabled by **TLS certificate**	Enabled by

							TLS certificate fields
39499	356	IPFIX_FLOWMON_TLS_PUBLIC_KEY_LENGTH	TLS public key length	4		Enabled by **TLS certificate**	Enabled by **TLS certificate fields**
39499	357	IPFIX_FLOWMON_TLS_JA3_FINGERPRINT	TLS JA3 fingerprint	16		Enabled by **TLS JA3**	Enabled by **TLS JA3 fields**
39499	371	IPFIX_FLOWMON_IEC104_FRAME_FMT	IEC104 frame format	1		Enabled by **IEC 104**	Enabled by **IEC104**
39499	372	IPFIX_FLOWMON_IEC104_ASDU_TYPE	IEC104 ASDU type	1		Enabled by **IEC 104**	Enabled by **IEC104**
39499	373	IPFIX_FLOWMON_IEC104_ASDU_OBJ_COUNT	IEC104 ASDU object count	1		Enabled by **IEC 104**	Enabled by **IEC104**
39499	374	IPFIX_FLOWMON_IEC104_ASDU_COT	IEC104 ASDU cause of transmission	1		Enabled by **IEC 104**	Enabled by **IEC104**
39499	375	IPFIX_FLOWMON_IEC104_ASDU_ORG	IEC104 ASDU originator address	1		Enabled by **IEC 104**	Enabled by **IEC104**
39499	376	IPFIX_FLOWMON_IEC104_ASDU_ADDRESS	IEC104 common ASDU address	2		Enabled by **IEC 104**	Enabled by **IEC104**
39499	390	IPFIX_FLOWMON_COAP_VERSION	CoAP version	1		Enabled by **COAP**	Enabled by **COAP**
39499	391	IPFIX_FLOWMON_COAP_MID	CoAP message ID	2		Enabled by **COAP**	Enabled by **COAP**
39499	392	IPFIX_FLOWMON_COAP_CODE	CoAP code	1		Enabled by **COAP**	Enabled by **COAP**
39499	393	IPFIX_FLOWMON_COAP_OPTIONS_COUNT	CoAP options count	2		Enabled by **COAP**	Enabled by **COAP**
39499	394	IPFIX_FLOWMON_COAP_TYPE	CoAP type	1		Enabled by **COAP**	Enabled by **COAP**
39499	395	IPFIX_FLOWMON_COAP_ACCEPT	CoAP accept	2		Enabled by **COAP**	Enabled by **COAP**
39499	396	IPFIX_FLOWMON_COAP_CONTENT_FORMAT	CoAP content format	2		Enabled by **COAP**	Enabled by **COAP**
39499	397	IPFIX_FLOWMON_COAP_TOKEN_LENGTH	CoAP token length	1		Enabled by **COAP**	Enabled by **COAP**
39499	398	IPFIX_FLOWMON_COAP_TOKEN	CoAP token	8		Enabled by **COAP**	Enabled by **COAP**
39499	399	IPFIX_FLOWMON_COAP_URI_HOST	CoAP URI host	64		Enabled by **COAP**	Enabled by **COAP**
39499	400	IPFIX_FLOWMON_COAP_URI_PATH	CoAP URI path	64		Enabled by **COAP**	Enabled by **COAP**
39499	401	IPFIX_FLOWMON_COAP_URI_QUERY	CoAP URI query	64		Enabled by **COAP**	Enabled by **COAP**
39499	410	IPFIX_FLOWMON_GOOSE_APPID	GOOSE application ID	2		Enabled by **GOOSE**	Enabled by **GOOSE**

39499	411	IPFIX_FLOWMON_GOOSE_CB_REF	GOOSE control block reference	64	Enabled by **GOOSE**	Enabled by **GOOSE**
39499	412	IPFIX_FLOWMON_GOOSE_DATA_SET	GOOSE data set	64	Enabled by **GOOSE**	Enabled by **GOOSE**
39499	413	IPFIX_FLOWMON_GOOSE_ID	GOOSE id	64	Enabled by **GOOSE**	Enabled by **GOOSE**
39499	414	IPFIX_FLOWMON_GOOSE_ST_NUM	GOOSE status number	4	Enabled by **GOOSE**	Enabled by **GOOSE**
39499	420	IPFIX_FLOWMON_MMS_TYPE	MMS message type	1	Enabled by **MMS**	Enabled by **MMS**
39499	421	IPFIX_FLOWMON_MMS_CONF_SERVICE_REQ	MMS confirmed service request type	1	Enabled by **MMS**	Enabled by **MMS**
39499	422	IPFIX_FLOWMON_MMS_CONF_SERVICE_RESP	MMS confirmed service response type	1	Enabled by **MMS**	Enabled by **MMS**
39499	423	IPFIX_FLOWMON_MMS_UNCONF_SERVICE	MMS unconfirmed service type	1	Enabled by **MMS**	Enabled by **MMS**
39499	430	IPFIX_FLOWMON_DLMS_TYPE	DLMS message type	1	Enabled by **DLMS**	Enabled by **DLMS**
39499	431	IPFIX_FLOWMON_DLMS_SUBTYPE	DLMS message subtype	2	Enabled by **DLMS**	Enabled by **DLMS**
39499	432	IPFIX_FLOWMON_DLMS_CLASS_ID	DLMS class id	2	Enabled by **DLMS**	Enabled by **DLMS**
39499	433	IPFIX_FLOWMON_DLMS_OBIS	DLMS OBIS code	6	Enabled by **DLMS**	Enabled by **DLMS**
39499	434	IPFIX_FLOWMON_DLMS_ATTR_METHOD_ID	DLMS attribute/method id	1	Enabled by **DLMS**	Enabled by **DLMS**
39499	435	IPFIX_FLOWMON_DLMS_DATA_TYPE	DLMS data type	1	Enabled by **DLMS**	Enabled by **DLMS**
39499	436	IPFIX_FLOWMON_DLMS_DATA_LENGTH	DLMS data length	2	Enabled by **DLMS**	Enabled by **DLMS**
39499	437	IPFIX_FLOWMON_DLMS_DATA_ACCESS_RESULT	DLMS data access result	1	Enabled by **DLMS**	Enabled by **DLMS**
39499	438	IPFIX_FLOWMON_DLMS_ACTION_RESULT	DLMS action result	1	Enabled by **DLMS**	Enabled by **DLMS**
39499	450	IPFIX_FLOWMON_VXLAN_VNI	VxLAN VNI	4	Enabled by **VxLAN**	Enabled by **VxLAN**

IPFIX - Other Vendors

Flowmon Collector supports IPFIX flow data from the following vendors:

- **Cisco**, PEN = 9.
- **VMware**, PEN = 6876.
- **Gigamon**, PEN = 26866.
- **IXIA**, PEN = 3054.
- **Nokia (ALU)**, PEN = 637.

Not all IPFIX fields from other vendors are supported. See the table below for a list of supported IPFIX fields.

Note: None of the following fields are applicable to Flowmon Probe, they can only be collected by Flowmon Collector.

PEN	ID	Field	Description	Input Length (bytes)	Collector Configuration
9	9316	IPFIX_CISCO_ART_CNT_SUM	The round trip time between SYN-ACK & ACK. Collect art client network time sum.	4	Enabled by **AVC Metrics**
9	9317	IPFIX_CISCO_ART_CNT_MAX	The round trip time between SYN-ACK & ACK. Collect art client network time maximum.	4	Enabled by **AVC Metrics**
9	9318	IPFIX_CISCO_ART_CNT_MIN	The round trip time between SYN-ACK & ACK. Collect art client network time minimum.	4	Enabled by **AVC Metrics**
9	9319	IPFIX_CISCO_ART_SNT_SUM	The round trip time between SYN & SYN-ACK and also called Server Network Delay (SND). Collect art server network time sum.	4	Enabled by **AVC Metrics**
9	9320	IPFIX_CISCO_ART_SNT_MAX	The round trip time between SYN & SYN-ACK and also called Server Network Delay (SND). Collect art server network time maximum.	4	Enabled by **AVC Metrics**
9	9321	IPFIX_CISCO_ART_SNT_MIN	The round trip time between SYN & SYN-ACK and also called Server Network Delay (SND). Collect art server network time minimum.	4	Enabled by **AVC Metrics**
9	9306	IPFIX_CISCO_ART_SRT_SUM	The time taken by an application to respond to a request. It is also called Application Delay (AD) or Application Response Time. Collect art server response time sum.	4	Enabled by **AVC Metrics**
9	9307	IPFIX_CISCO_ART_SRT_MAX	The time taken by an application to respond to a request. It is also called Application Delay (AD) or Application Response Time. Collect art server response time maximum.	4	Enabled by **AVC Metrics**
9	9308	IPFIX_CISCO_ART_SRT_MIN	The time taken by an application to respond to a request. It is also called Application Delay (AD) or Application Response Time. Collect art server response time minimum.	4	Enabled by **AVC Metrics**
9	9292	IPFIX_CISCO_ART_CNT_RESPONSES	Number of Req-Rsp pairs received within the monitoring interval.	4	Enabled by **AVC Metrics**
9	9268	IPFIX_CISCO_ART_CLIENT_RETRANSMISSIONS	ART Count Retransmissions metric is the packet count for all the retransmitted client packets	4	Enabled by **AVC Metrics**

9	9270	IPFIX_CISCO_ART_SERVER_RETRANSMISSIONS	ART Count Retransmissions metric is the packet count for all the retransmitted client packets.	4	Enabled by **AVC Metrics**
9	9313	IPFIX_CISCO_NETWORK_TIME_SUM	Network Time is known as the round-trip time that is the summation of CND and SND. It is also called Network Delay (ND).	4	Enabled by **AVC Metrics**
9	9293	IPFIX_CISCO_ART_RES_HIST_1	Number of responses by response time in 7-bucket histogram. Bucket 1, response time < 2 milliseconds.	4	Enabled by **AVC Histogram**
9	9294	IPFIX_CISCO_ART_RES_HIST_2	Number of responses by response time in 7-bucket histogram. Bucket 2, response time is between 2-5 milliseconds.	4	Enabled by **AVC Histogram**
9	9295	IPFIX_CISCO_ART_RES_HIST_3	Number of responses by response time in 7-bucket histogram. Bucket 3, response time is between 5-10 milliseconds.	4	Enabled by **AVC Histogram**
9	9296	IPFIX_CISCO_ART_RES_HIST_4	Number of responses by response time in 7-bucket histogram. Bucket 4, response time is between 10-50 milliseconds.	4	Enabled by **AVC Histogram**
9	9297	IPFIX_CISCO_ART_RES_HIST_5	Number of responses by response time in 7-bucket histogram. Bucket 5, response time is between 50-100 milliseconds.	4	Enabled by **AVC Histogram**
9	9298	IPFIX_CISCO_ART_RES_HIST_6	Number of responses by response time in 7-bucket histogram. Bucket 6, response time is between 100-500 milliseconds.	4	Enabled by **AVC Histogram**
9	9299	IPFIX_CISCO_ART_RES_HIST_7	Number of responses by response time in 7-bucket histogram. Bucket 7, response time is between 500 - 1000 milliseconds.	4	Enabled by **AVC Histogram**
9	9300	IPFIX_CISCO_ART_RES_LATE	Number of responses received after the max Response Time. Current threshold of timeout is 1 second. Also called Number of late responses (timeouts).	4	Enabled by **AVC Histogram**
9	12235	IPFIX_CISCO_HTTP_INFO	Cisco HTTP Host and URL	Variable	Enabled by **AVC HTTP**
6876	950	IPFIX_VMWARE_RULEID	Firewall rule ID	4	Enabled by **NSX**
6876	951	IPFIX_VMWARE_VMUUID	Uniquely identifies virtual machine	16	Enabled by **NSX**
6876	952	IPFIX_VMWARE_VNICINDEX	Index of the VNIC for the specified virtual machine	4	Enabled by **NSX**
26866	1	IPFIX_GIGAMON_HTTP_REQUEST_URL	Gigamon HTTP Request URL	128	Enabled by **HTTP Host and URL**
26866	1	IPFIX_GIGAMON_HTTP_RESPONSE_CODE	Gigamon HTTP Response Code	2	Enabled by **HTTP Host and URL**
26866	201	IPFIX_GIGAMON_DNS_IDENTIFIER	Gigamon DNS identifier generated by the device that creates DNS query	1	Enabled by **DNS**
26866	202	IPFIX_GIGAMON_DNS_OP_CODE	Gigamon DNS Op code which specifies query type	1	Enabled by **DNS**
26866	203	IPFIX_GIGAMON_DNS_DNS_RESPONSE_CODE	Gigamon DNS response code which specifies response	128	Enabled by **DNS**
26866	204	IPFIX_GIGAMON_DNS_QUERY_NAME	Gigamon DNS query name	128	Enabled by **DNS**
26866	205	IPFIX_GIGAMON_DNS_REPONSE_NAME	Gigamon DNS response name	128	Enabled by **DNS**

26866	206	IPFIX_GIGAMON_DNS_REPONSE_TTL	Gigamon DNS response TTL	4	Enabled by **DNS**
26866	207	IPFIX_GIGAMON_DNS_REPONSE_IPV4_ADDR	Gigamon DNS IPv4 address in the response	4	Enabled by **DNS**
26866	208	IPFIX_GIGAMON_DNS_REPONSE_IPV6_ADDR	Gigamon DNS IPv6 address in the response	16	Enabled by **DNS**
26866	101	IPFIX_GIGAMON_SSL_CER_ISSUER_NAME	Gigamon SSL certificate issuer name	Variable	Enabled by **SSL**
26866	102	IPFIX_GIGAMON_SSL_CER_SUBJECT_NAME	Gigamon SSL certificate subject name	Variable	Enabled by **SSL**
26866	105	IPFIX_GIGAMON_SSL_CER_VALID_NOT_BEFORE	Gigamon SSL certificate valid not before	13	Enabled by **SSL**
26866	106	IPFIX_GIGAMON_SSL_CER_VALID_NOT_AFTER	Gigamon SSL certificate valid not after	13	Enabled by **SSL**
26866	107	IPFIX_GIGAMON_SSL_CER_SERIAL_NUMBER	Gigamon SSL certificate serial number	20	Enabled by **SSL**
26866	108	IPFIX_GIGAMON_SSL_CER_SIG_ALG	Gigamon SSL certificate signature algorithm	9	Enabled by **SSL**
26866	109	IPFIX_GIGAMON_SSL_CER_SUBJECT_PUB_ALG	Gigamon SSL certificate subject public algorithm	9	Enabled by **SSL**
26866	110	IPFIX_GIGAMON_SSL_CER_SUBJECT_PUB_KEY_SIZE	Gigamon SSL certificate subject public algorithm size	2	Enabled by **SSL**
26866	111	IPFIX_GIGAMON_SSL_CER_SUBJECT_ALT_NAMES	Gigamon SSL certificate subject alternate names	Variable	Enabled by **SSL**
26866	112	IPFIX_GIGAMON_SSL_SERVER_NAME_INDICATION	Gigamon SSL server name indication	Variable	Enabled by **SSL**
26866	113	IPFIX_GIGAMON_SSL_SERVER_VERSION	Gigamon SSL server version	2	Enabled by **SSL**
26866	114	IPFIX_GIGAMON_SSL_SERVER_CIPHER	Gigamon SSL server cipher	2	Enabled by **SSL**
26866	115	IPFIX_GIGAMON_SSL_SERVER_COMPRESSION_METHOD	Gigamon SSL server compression method	1	Enabled by **SSL**
26866	116	IPFIX_GIGAMON_SSL_SERVER_SESSION_ID	Gigamon SSL server session ID	32	Enabled by **SSL**
26866	1608	IPFIX_GIGAMON_RADIUS_CALLED_STATION_ID	Gigamon RADIUS Called station ID	Variable	Enabled by **RADIUS**
26866	1637	IPFIX_GIGAMON_RADIUS_CALLING_STATION_ID	Gigamon RADIUS Calling station ID	Variable	Enabled by **RADIUS**
26866	1647	IPFIX_GIGAMON_RADIUS_LOGIN	Gigamon RADIUS Username	Variable	Enabled by **RADIUS**
26866	10002	IPFIX_GIGAMON_RADIUS_NAT_ADDRESS	Gigamon RADIUS NAT IP address	4	Enabled by **RADIUS**
26866	10003	IPFIX_GIGAMON_RADIUS_PORT_START	Gigamon RADIUS NAT port start	4	Enabled by **RADIUS**
26866	10004	IPFIX_GIGAMON_RADIUS_PORT_END	Gigamon RADIUS NAT port end	4	Enabled by **RADIUS**

3054	183	IPFIX_IXIA_HTTP_HOSTNAME	IXIA HTTP Hostname	32	Enabled by **HTTP Host and URL**
3054	184	IPFIX_IXIA_HTTP_URI	IXIA HTTP URI	64	Enabled by **HTTP Host and URL**
637	93	IPFIX_ALU_NAT_SUB_STRING	A string that represents the NAT subscriber construct. The original IP source address, before NAT is performed is included in this string. For example: LSN-Host@10.10.10.101	Variable	Enabled by **NEL Common block, Mapped to an existing field storing SRC NAT IP**

sFlow v2/4/5

Record	Description	Collector Configuration
input	SNMP index of input interface.	Collected by default
output	SNMP index of output interface.	Collected by default
src_as	Autonomous system number of the source.	Collected by default
dst_as	Autonomous system number of the destination.	Collected by default
src_vlan	Source Virtual LAN.	Enabled by **SRC/DST VLAN ID labels**
dst_vlan	Destination Virtual LAN.	Enabled by **SRC/DST VLAN ID labels**
dst_tos	Destination type of service.	Enabled by **SRC/DST mask, (dst) TOS, Direction**
dir	Direction / Directory.	Enabled by **SRC/DST mask, (dst) TOS, Direction**
src_mask	Source address prefix mask bits.	Enabled by **SRC/DST mask, (dst) TOS, Direction**
dst_mask	Destination address prefix mask bits.	Enabled by **SRC/DST mask, (dst) TOS, Direction**
in_src_mac	Input source MAC address.	Enabled by **In SRC/out DST MAC address**
out_dst_mac	Output destination MAC address.	Enabled by **In SRC/out DST MAC address**
nexthop	IP address of next hop router.	Enabled by **Next HOP IP address**
bgp_nexthop	BGP Next hop IP address.	Enabled by **BGP next HOP IP address**
first	SysUptime at start of flow.	Collected by default
last	SysUptime at the time the last packet of the flow was received.	Collected by default
fwd_status	(sFlow data stored per record)	Collected by default
tcp_flags	Cumulative OR of TCP flags	Collected by default
prot	IP protocol type	Collected by default
tos	IP type of service (TOS)	Collected by default
srcport	TCP/UDP source port number or equivalent	Collected by default

dstport	TCP/UDP destination port number of equivalent	Collected by default
scr_ipaddr	Source IP address	Collected by default
dst_ipaddr	Destination IP address	Collected by default

jFlow

Supported – compatible with NetFlow v5/7/9.

> **Warning:**
> jFlow works with NetFlowStart attributes in a different way than traditional flow vendors. The original timestamp of the first packet is preserved even when the flow is exported due to active timeout expiration. As a result, there can be a timeframe of 16:10-16:15 which contains NetFlowStart 14:37. Flowmon can cope with the described approach, but the user needs to be aware of this behavior.

NetFlow Lite

Supported – compatible with NetFlow v5/7/9.

NetStream

Supported – compatible with [NetFlow v5/7](#)/[9](#).

cflowd

Supported – compatible with [NetFlow v5/7](#)/[9](#).

AWS VPC Flow Logs

Support is enabled via **Flowmon Configuration Center – FMC Configuration – AWS Flow Logs**. Logs are periodically downloaded, converted to IPFIX and sent to the specified listening port of the local collector. The table below specifies how AWS VPC Flow Logs fields are mapped to supported IPFIX fields.

Field	Maps to, in IPFIX	Collector Configuration
version	**No mapping**	**Not supported**
account-id	**No mapping**	**Not supported**
interface-id	**No mapping, replaced by a generated value for IPFIX_ingressInterface**	Collected by default
srcaddr	IPFIX_SourceIPv4Address IPFIX_SourceIPv6Address	Collected by default
dstaddr	IPFIX_DestinationIPv4Address IPFIX_DestinationIPv6Address	Collected by default
srcport	IPFIX_SourceTransportPort	Collected by default
dstport	IPFIX_DestinationTransportPort	Collected by default
protocol	IPFIX_protocolIdentifier	Collected by default
packets	IPFIX_packetTotalCount	Collected by default
bytes	IPFIX_octetTotalCount	Collected by default
start	IPFIX_flowStartMilliseconds	Collected by default
end	IPFIX_flowEndMilliseconds	Collected by default
action	IPFIX_firewallEvent	**Not supported**
log-status	**No mapping**	**Not supported**

Flowmon Collector - Extra Fields

Note: The following fields are automatically added by Flowmon Collector irrespective of the flow standard used. Fields containing user identity are populated only when parsing of the user identity information is enabled in **Flowmon Configuration Center – System – System Settings – Syslog Server**. For details, please refer to Flowmon User Guide.

Field	Description	Collector Configuration
Source country	Identification of source country	Collected by default
Destination country	Identification of destination country	Collected by default
Source AS number	Autonomous system source number	Collected by default
Destination AS number	Autonomous system destination number	Collected by default
Time packet received	Time packet received	Enabled by **Time stamp flow received by collector**
User Identity Source Username	User name provided from source identity	**Added to each flow from Active Directory logs**
User Identity Destination Username	User name provided from destination identity	**Added to each flow from Active Directory logs**

History

Date	Description
November 20, 2020	Added the AWS VPC Flow Logs section.
November 13, 2020	- Added the Overview section and the **Probe Configuration** column in NetFlow v5/7/9 and IPFIX. - Renamed the **Database Field Names** column to **Collector Configuration**. - Added **Nokia (ALU)** and field **IPFIX_ALU_NAT_SUB_STRING** to IPFIX - Other Vendors.