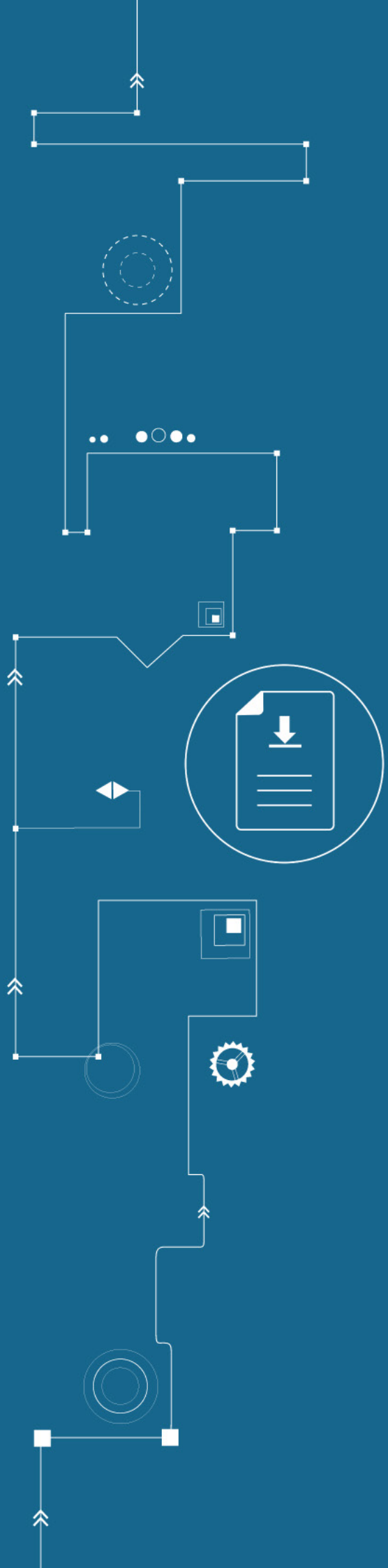
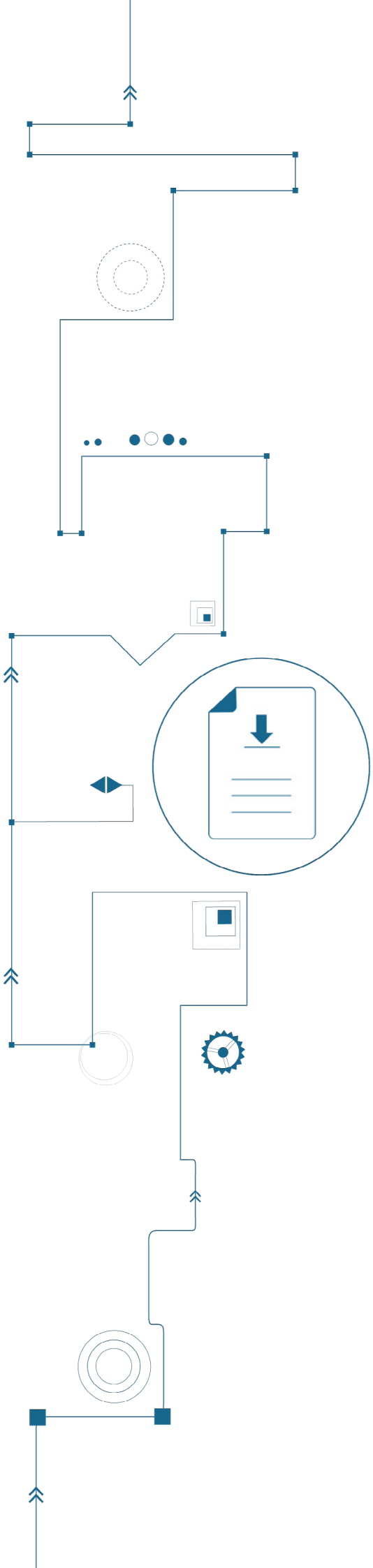
CUBRO
NETWORK VISIBILITY



FLOW VS TIME WINDOW BASED MONITORING

WHITE PAPER

MAY 2021



CUBRO
NETWORK VISIBILITY

TABLE OF CONTENTS

Introduction

NetFlow is a widely used tool in network monitoring which has strengths and weaknesses. This white paper discusses time window based monitoring and compares it to NetFlow.

What is Flow and NetFlow

In packet switching networks, traffic flow, packet flow or network flow is a sequence of packets from a source computer to a destination, which may be another host, a multicast group, or a broadcast domain.

RFC 2722 defines traffic flow as "an artificial logical equivalent to a call or connection." RFC 3697 defines traffic flow as "a sequence of packets sent from a particular source to a particular unicast, anycast, or multicast destination that the source desires to label as a flow." In more technical terms, a flow is defined by its 5-tuple.

Flow identifies a communication channel, and all packets sharing the same 5-tuple fields belong to the same flow.

NetFlow is a feature that was introduced on Cisco routers around 1996 that provides the ability to collect IP network traffic as it enters or exits an interface. NetFlow comprises functionality built into network devices that collects measurements for each flow and exports them to another system for analysis. For example, NetFlow captures the timestamp of a flow's first and last packets (and hence its duration), the total number of bytes and packets exchanged, a summary of the flags used in TCP connections, and other details.

By collecting and analyzing this flow data, we can learn details about how the network is being used. For example, flow analysis is helpful in data center design and operations.

NetFlow Pros & Cons

NetFlow based monitoring is commonly used and there are lots of tools and expertise.

NetFlow is used in network monitoring, network planning and security analysis.

NetFlow is designed for Transport Layer troubleshooting. For example to visualize traffic patterns throughout the entire network, when and how frequently users access an application in the network and to monitor and profile a user's utilization of network and application resources to detect any potential security or policy violations.

NetFlow is useful for tracking and anticipating network growth. It can help to plan upgrades to increase the number of ports, routing devices or higher-bandwidth interfaces needed to meet growing demand. Netflow can detect changes in network behavior to identify anomalies indicative of a security breach.

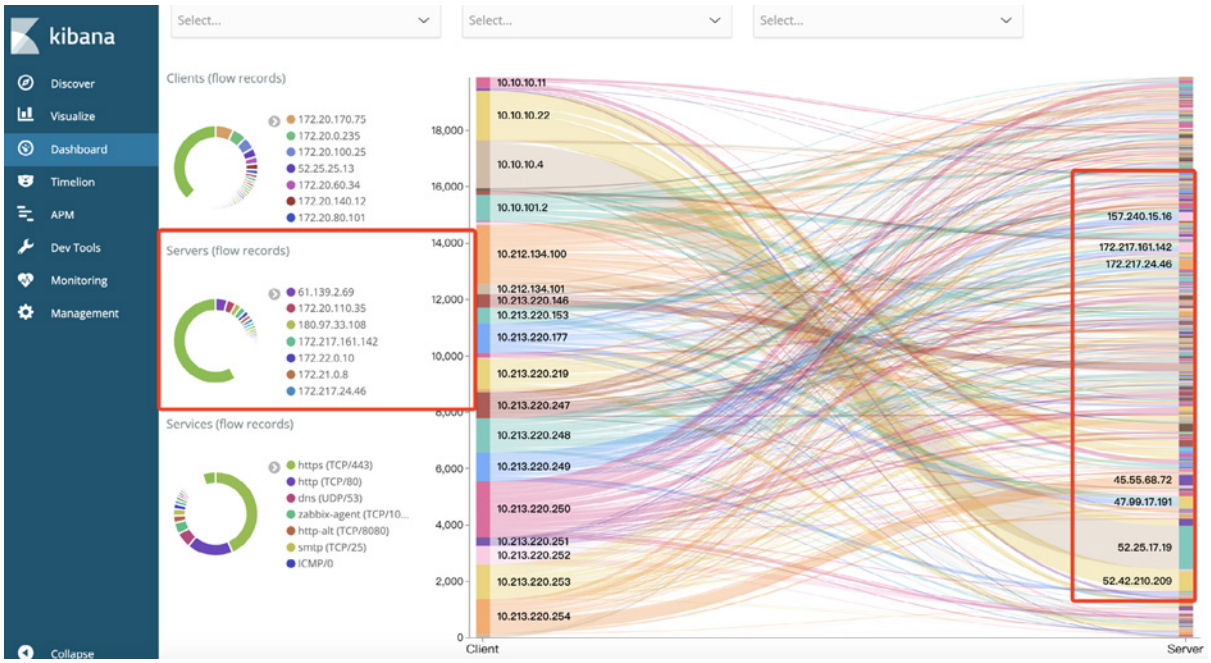
NetFlow is not practical with user plane monitoring, because it uses a lot of bandwidth and has a performance impact on the devices where it is implemented with and therefore packet sampling is used. However, sampling reduces network visibility and could prevent Operations teams from uncovering critical security threats or performance issues. NetFlow has limited coverage in terms of what you can monitor and what fields are available in NetFlow.

NetFlow Comparison

NetFlow has inspired other technologies such as IPFIX. IPFIX has NetFlow functionality with additional flexibility and extensions. NetFlow and IPFIX are flow export protocols that aim at aggregating packets into flows. sFlow is an alternative approach which is a packet-sampling technology rather than a "flow-sampling" technology. It has no notion of flows or packet aggregation but provides full L2-L7 visibility.

Cubro's Time Window based solution collects data for uploads, downloads and internal traffic as well as DPI information from a device/user perspective over time.

User Plane monitoring with NetFlow



Flow based Monitoring is not efficient for user-generated traffic.

Millions of flows

Web 2.0 allows users to interact and collaborate with each other through social media as creators of user-generated content. It is very common to see a list of 50+ open sessions.

NetFlow records use a lot of bandwidth

Many web pages establish dozens of TCP sessions with each TCP session generating a flow. Some of these sessions only transfer a few bytes thus NetFlow record can be bigger and generate more traffic than the original network traffic.

Flow based Monitoring Challenges

Stateless communication

The session is established when the device is started, but the session flow can remain open forever.

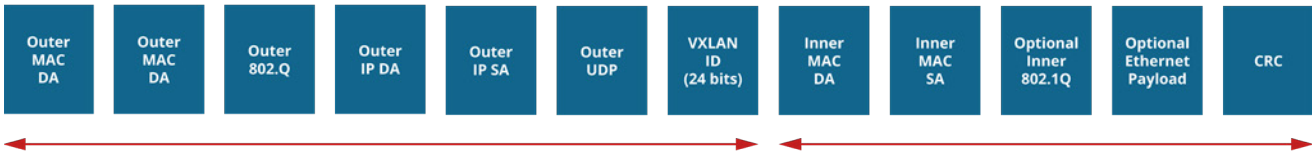
Sessions not terminated

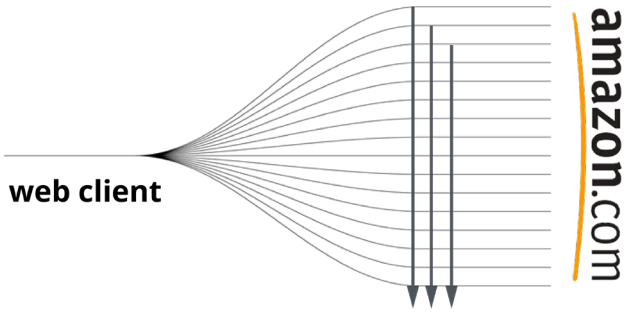
In an ideal network the flow monitoring tool sees the “fin” message in the TCP flow and knows the session has ended. When the session has ended the probe writes all data to disc, stops supporting this flow, and relocates used memory.

In real life most sessions will not close regularly, and are terminated only by shutting off the device or rebooting the device.

Tunneled Traffic: VXLAN

Typically, IPFIX and NetFlow don’t support tunnels. NetFlow will show only the outer tunnel, and the more relevant inner tunnel information is lost.





Context missing

A flow is produced for each 5 tuple connection. In the example above, many of the flows cannot be detected as Amazon related, because the external domain cannot be resolved.

High resource requirements

When the flows expire they are exported to a Netflow Collector for analysis and archiving, often requiring significant processing, storage and retrieval resources.

Flow monitoring traffic volume

Traffic volume depends on the number of sessions

The amount of traffic that is produced by a NetFlow solution does not depend on the bandwidth, but on the amount of sessions/flows.

Example of flow traffic volume

Assumptions	: 10Gbps traffic, one flow
Result	: 1kbs
Rule of Thumb	: 1-3% of the input traffic

Example 2

Assumptions	: 900Gbps traffic, one flow
Result	: 18Gbps using 2% of the input
Retention time	: 30 days
Total storage	: 18G/8*60*60*24*30 = 5,800 TB

Example 3

Assumptions	: CSP with 5M subscribers, every user on avg. 50 sessions
Active users	: 30%
Open sessions	: 40
Total flows	: Total flows needed 30% * 5M * 40 = 60M flows

High number of sessions, continuously open sessions and the volume produced by flow monitoring unnecessarily loads the monitoring and analytic systems leading to high investments.

Time Window based Monitoring

Flow based user plane monitoring

Flow based computation on user plane requires lots of resources and leads to high investments. In addition, important data cannot be extracted from single flows.

Time Window based user plane monitoring

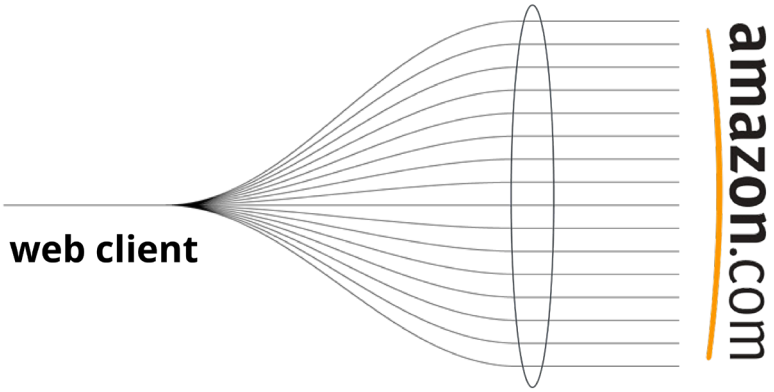
Aggregating metadata within a time window solves the challenges and allows more effective use of resources without losing any important data.

Metadata is collected for uploads, downloads and internal traffic, as well as DPI information from a device/user perspective over time.

Cubro's Time Window based solution extracts the essential information out of large data streams with the benefit of efficiently using less resources and storage.

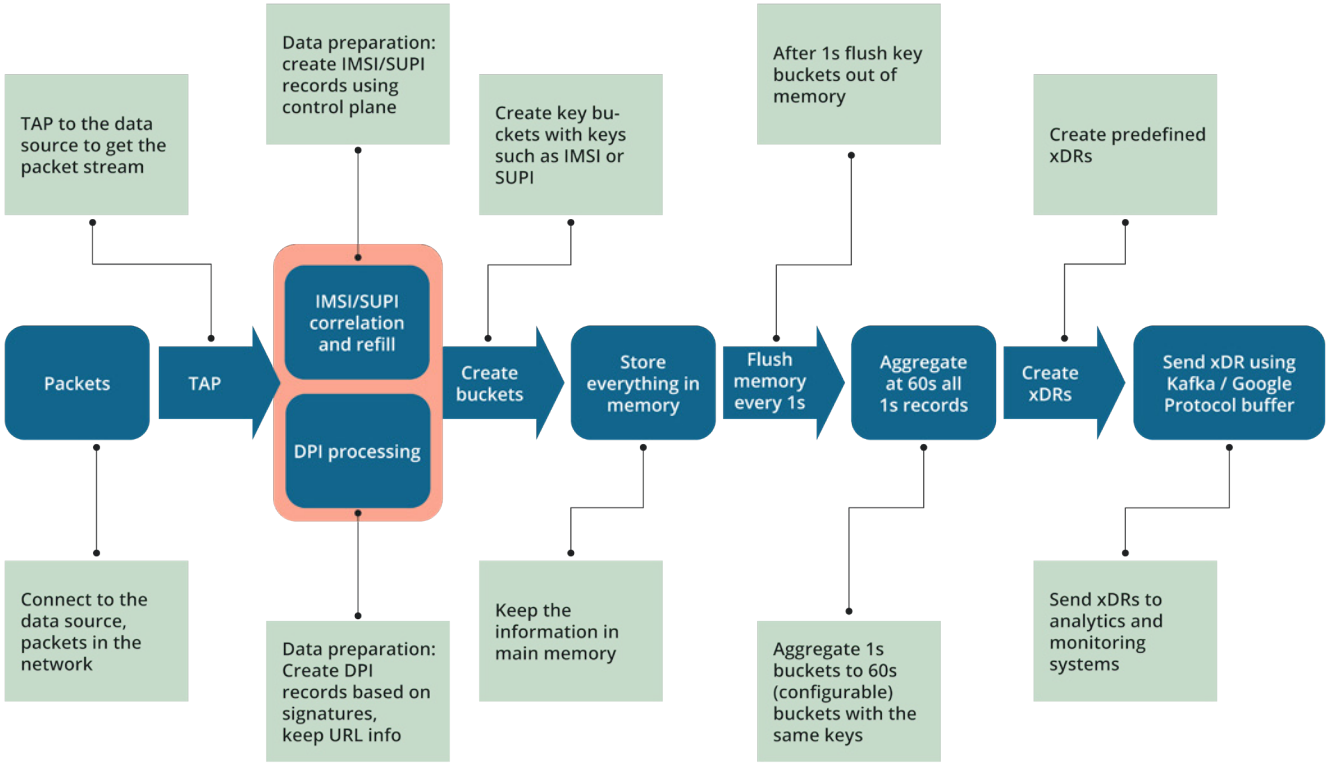
The time window based approach supports various views such as service, client/device and server perspectives:

- Service: The amount of traffic, usage frequency over time, traffic volumes uploaded or downloaded
- Client/Device: The usage of the network by upload and download per user, services and locations used, frequency of the use over time.



Every client has a bucket for each application. All packets are collected and counted for a defined time window (configurable). When the time window is closed an XDR is produced/enriched and sent out using significantly reduced volume.

Time window based Monitoring Data flow



Data Structure

Cubro's time-window based aggregation produces data structure easily understood by humans.

```
message TimeWindow {
  int64 timestamp = 1;
  int64 bps = 2;
  int64 incomingPkts = 3;
  int64 outgoingPkts = 4;
  int64 internalPkts = 5;
  int64 incomingBytes = 6;
  int64 outgoingBytes = 7;
  int64 internalBytes = 8;
  int32 connections = 9;
  repeated ServiceData services = 10;
  repeated ClientData clients = 11;
  repeated IpData servers = 12;
  repeated TypeBytesData l3 = 13;
  repeated TypeBytesData l4 = 14;
  repeated PortData ports = 15;
  repeated DnsData domains = 16;
}

message ServiceData {
  ServiceEntry service = 1;
  int64 incomingPkts = 2;
  int64 outgoingPkts = 3;
  int64 internalPkts = 4;
  int64 incomingBytes = 5;
  int64 outgoingBytes = 6;
  int64 internalBytes = 7;
  repeated IpData clients = 8;
  repeated IpData servers = 9;
}
```

Note: Partial representation of dataset.

Summary

There are several definitions for Network Management and Network Monitoring and the descriptions depend on the particular use case or situation.

For example, network optimization requires monitoring as means of determining if the network is running optimally.

Using this view highlights the importance of having visibility in the network of connected devices, early insights into future infrastructure needs and the ability to identify security threats faster.

Network Management looks different from a telecom perspective compared to IT management, and security, application management and subscriber centric monitoring also have their own special requirements. In recent years in addition to detecting an incident, response and automation have become increasingly important.

In this wide range of requirements NetFlow has been used to understand and visualize traffic patterns. Time Window based monitoring on the other hand has its focus on subscriber behavior and is effective in analyzing the user plane traffic.