Network Management

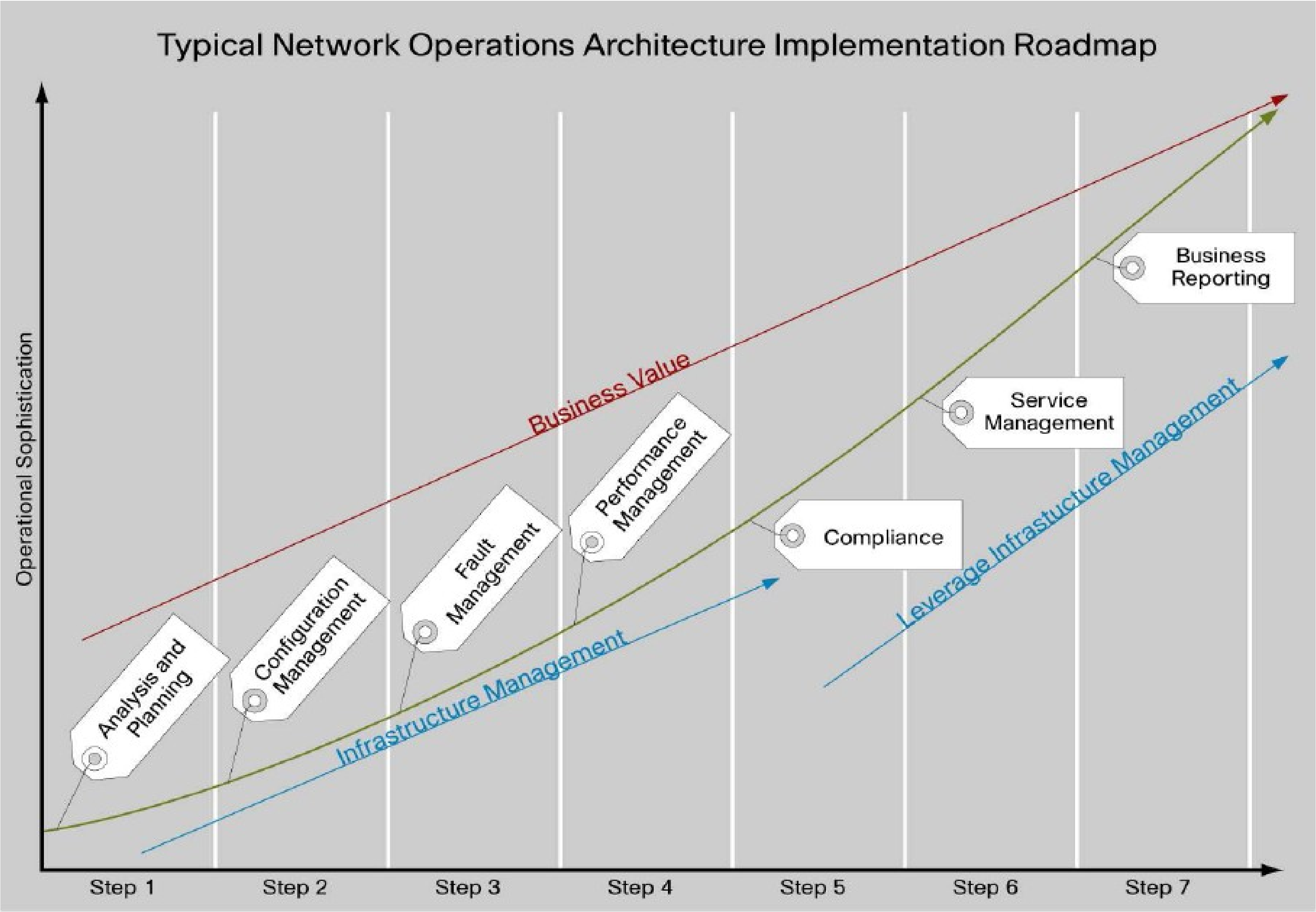
Arquitetura de Redes

Mestrado Integrado
Engenharia de Computadores e Telemática
DETI-UA

universidade de aveiro

deti.ua.pt

Network Implementation and Management



Network Management (1)

- Documentation and Diagrams
  - Network documentation and diagrams are critical in a production environment
    - Provide information when troubleshooting network outages; they are, however, static
    - In a dynamic network environment, purely static documentation is not suitable
  - An effective configuration management capability <u>should provide up-to-date and dynamically updated information</u>
    - When combined with static documentation and diagrams, it provides more relevant information to support network operations
- Compliance
  - Compliance is about meeting regulations imposed by government or industry
  - It is not however a matter of buying a product and being compliant;<u> it is about building capabilities to support compliance over time</u>

Network Management (2)

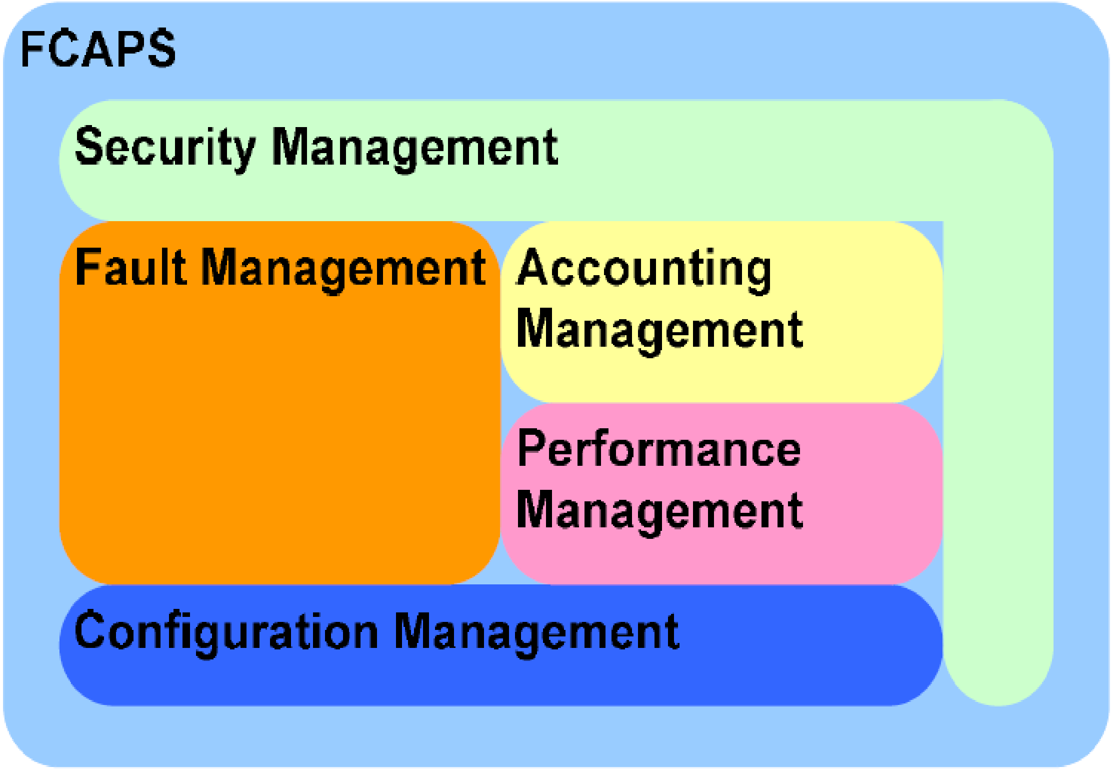
- Managing Risk
  - A key issue with network management is the rapid increase in the number and heterogeneity of network elements
    - The ability to understand risk exposure becomes more difficult
  - The ability to understand exposure is no longer possible without new capabilities in auditing and reporting
  - Requires appropriate supporting processes and operational methodologies so that the risk can be understood and expediently mitigated
- Time to Resolve
  - A key measure in many service levels is incident time to resolution
  - An incident will result from a network outage, and in simple terms, an outage to a production network that is considered stable is caused by one of the following:
    - Layer 1 network failure (leased line, fiber cut, and so on),
    - Physical infrastructure failure, power, air conditioning
    - Hardware failure, power supply, chassis, or module
    - Software failure, due to memory leak or bug
    - Security exploit, causing DOS or software failure
    - A change in configuration, either logical (being a new feature) or physical (being new hardware or connections)
  - Configuration management assists with time to resolution by providing the necessary information to support troubleshooting and decision making. If a network outage is caused by a configuration change, this needs to be eliminated as the root cause in the first instance.

Network Management Models

- FCAPS (Fault, Configuration, Accounting, Performance and Security)
  - Popular conceptual framework for network management
  - Sponsored by ISO
  - For all networks
  - Functions: Fault, Configuration, Accounting, Performance and Security
- TMN (Telecommunications Management Network)
  - Conceptual framework for many Service Provider's network management systems
  - Sponsored by ITU-T
  - For telecommunications networks
  - Functions: Business Management, Service Management, Network Management and Element Management
- OAM&P (Operation, Administration, Maintenance and Provisioning)
  - Widely adopted by large Service Providers in their network management systems
  - Sponsored by service providers
  - For telecommunication networks
  - Functions: Operation, Administration, Maintenance and Provisioning
- TOM (Telecoms Operations Map) and eTOM (enhanced Telecom Operations Map)
  - Designed to replace the OAM&P.
  - Sponsored by TeleManagement Forum
  - For service provider's networks
  - Functions: Network and Systems management, Service development and Operations, Customer care

FCAPS

- Each of the functions interacts with each of the others
- Security has to touch all the functions to be effective
- Configuration is the function that holds the important data for all the functions



FCAPS - Fault Management

- Set of functions that enable the detection, isolation, and correction of abnormal operation of the telecommunication network
- Consists of the following functions
  - Reliability, Availability, and Survivability (RAS) quality assurance
    - Establishes the reliability criteria that guide the design policy for redundant equipment
  - Alarm surveillance
    - Describes the capability to monitor network element failures in near-real time
  - Fault localization
  - Fault correction
  - Testing
    - A network element analyzes equipment functions
    - Active testing of external device components, such as circuits, links, and neighbor devices
  - Trouble administration
    - Transfers trouble reports originated by customers and trouble tickets originated by proactive failure-detection checks

FCAPS - Configuration Management

- Provides functions to identify, collect configuration data from, exercise control over, and provide configuration data to network elements.
- Configuration management supports the following functions
  - Installing the physical equipment and logical configurations
  - Service planning and negotiation
    - Planning for the introduction of new services, changing deployed service features, and disconnecting existing services
  - Provisioning
    - Consists of necessary procedures to bring equipment into service but does not include installation
  - Status and control
    - Provides the capability to monitor and control certain aspects of the network elements
  - Network planning and engineering
    - Functions associated with determining the need for growth in capacity and the introduction of new technologies

FCAPS - Accounting Management

- Provides the procedures to measure the use of network services and determine costs to the service provider and charges to the customer for such use
- Includes the following functions:
- Usage measurement
  - Planning and management of the usage measurement process
  - Network and service usage aggregation, correlation, and validation
  - Usage distribution
  - Usage surveillance
  - Usage testing and error correction
  - Measurement rules identification
  - Usage short-term and long-term storage
  - Usage accumulation and validation
  - Administration of usage data collection
  - Usage generation
- Tariffing and pricing
  - A tariff is used to determine the amount of payment for services usage.
- Collections and finance
  - Functionality for administering customer accounts, informing customers of balances and payment dates, and receiving payments.
- Enterprise control
  - This group supports the enterprise's financial responsibilities, such as budgeting, auditing, and profitability analysis.

FCAPS - Performance Management

- Provides functions to evaluate and report on the behavior of telecommunication equipment and the effectiveness of the network or network element
- Collect and analyze statistical data for the purpose of monitoring and correcting the behavior and effectiveness of the network, network elements, or other equipment, and to aid in planning, provisioning, maintenance, and quality measurement.
- Includes the following functions:
  - Performance quality assurance
    - Includes quality measurements, such as performance goals and assessment functions
  - Performance monitoring
    - Continuous collection of data concerning the performance of the network element.
    - May also be designed to detect characteristic patterns of impairment before the quality has dropped below an acceptable level
  - Performance management control
    - Includes the setting of thresholds and data analysis algorithms and the collection of performance data
    - It has no direct effect on the managed network
    - Includes functions that affect the routing and processing of traffic
  - Performance analysis
    - Collected performance records may require additional processing and analysis to evaluate the entity's performance level
    - Includes functions: Recommendations for performance improvement, Exception threshold policy, Traffic forecasting (trending), Performance summaries (per network and service, and traffic-specific), Exception analysis (per network and service, and traffic-specific), Capacity analysis (per network and service, and traffic-specific) and Performance characterization

FCAPS - Security Management

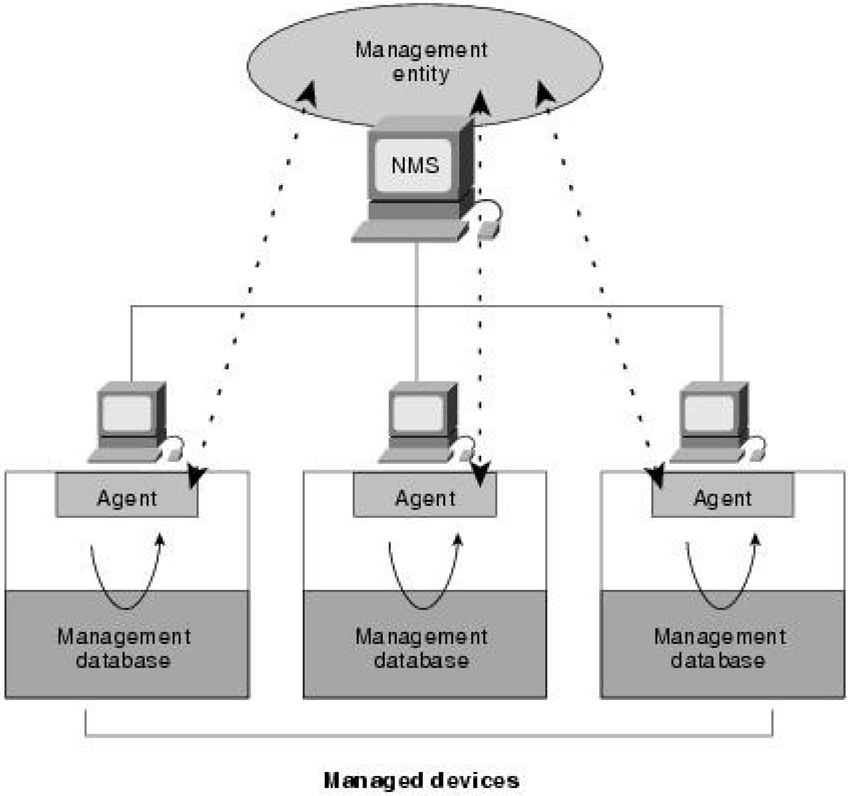
- Security is required for all functional areas.
- Security management consists in providing
  - Services for communications provide authentication, access control, data confidentiality, data integrity, and non-repudiation.
  - Security event detection and reporting reports activities that may be construed as a security violation (unauthorized user, physical tampering with equipment) on higher layers of security applications
- Security management includes the following functions:
  - Prevention
  - Detection
  - Containment and recovery
  - Security administration

SNMP

Simple Network Management Protocol

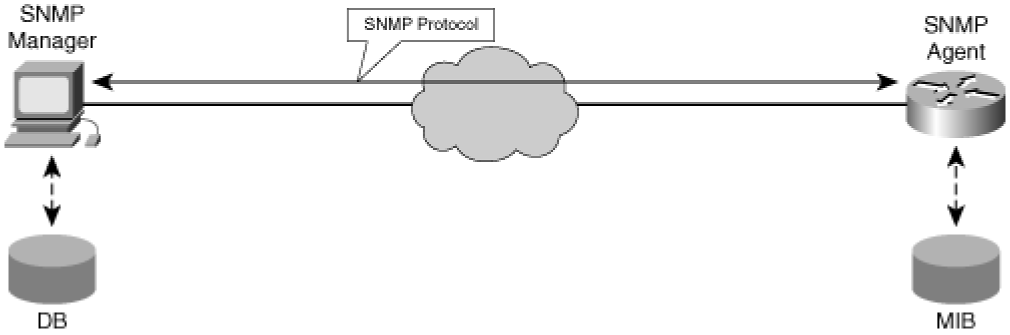
SNMP Basic Components

- An SNMP-managed network consists of three key components:
- Managed devices
  - Network node that contains an SNMP agent.
  - Collect and store management information and make this information available using SNMP.
  - Can be routers and access servers, switches and bridges, hubs, computer hosts, or printers.
- Agents
  - Network-management software module that resides in a managed device.
- Network-management systems (NMSs)
  - Executes applications that monitor and control managed devices.
  - Provide the bulk of the processing and memory resources required for network management.
  - One or more NMSs must exist on any managed network.



Data Collection Protocols: SNMP, SMI, and MIB

- SNMP is an Internet protocol developed by the IETF
- It is designed to facilitate the exchange of management information between network elements



- SNMP agent
  - A software module that resides in network elements; it collects and stores management information specified in the supported MIB modules. The SNMP agent responds to SNMP requests from an NMS station for information and actions. The SNMP agent can send fault notifications pro-actively to the SNMP manager.
- Managed object
  - A representation of something that can be managed.
  - Managed objects differ from variables, which are particular object instances.
- Management Information Base (MIB)
  - A collection of managed objects residing in a virtual information store.
  - A collection of related managed objects is defined in a specific MIB module.
  - A MIB can be considered a local data store at the network element.
- Syntax notation
  - A language used to describe managed objects in a machine-independent format
  - SNMP-based management systems use a subset of the International Organization for Standardization's (ISO) Open System Interconnection (OSI) Abstract Syntax Notation 1 (ASN.1, International Telecommunication Union Recommendation X.208) to define both the packets exchanged by the management protocol and the objects that are to be managed.
- Structure of Management Information (SMI)
  - Defines the rules for describing management information (the MIB). The SMI is defined using ASN.1.

SNMP Basic Commands

- Managed devices are monitored and controlled using four basic SNMP commands: read, write, trap, and traversal operations.
  - The **read** command is used by an NMS to monitor managed devices. The NMS examines different variables that are maintained by managed devices.
  - The **write** command is used by an NMS to control managed devices. The NMS changes the values of variables stored within managed devices.
  - The **trap** command is used by managed devices to asynchronously report events to the NMS. When certain types of events occur, a managed device sends a trap to the NMS.
  - Traversal operations are used by the NMS to determine which variables a managed device supports and to sequentially gather information in variable tables, such as a routing table.

SNMP: Polling

- Manager periodically asks the agent for new information

☺ Advantage: Manager completely controls the equipment, and knows all network details

☹ Disadvantage: Delay between event and its entry in the system, and unnecessary communication overhead:

  - Slow polling, slow answer to the events
  - Quick polling, quick reaction, but large bandwidth wastage

SNMP: Traps

- There is an event → trap is sent
- Trap contains appropriate information
  - equipment name, time instant of event, type of event

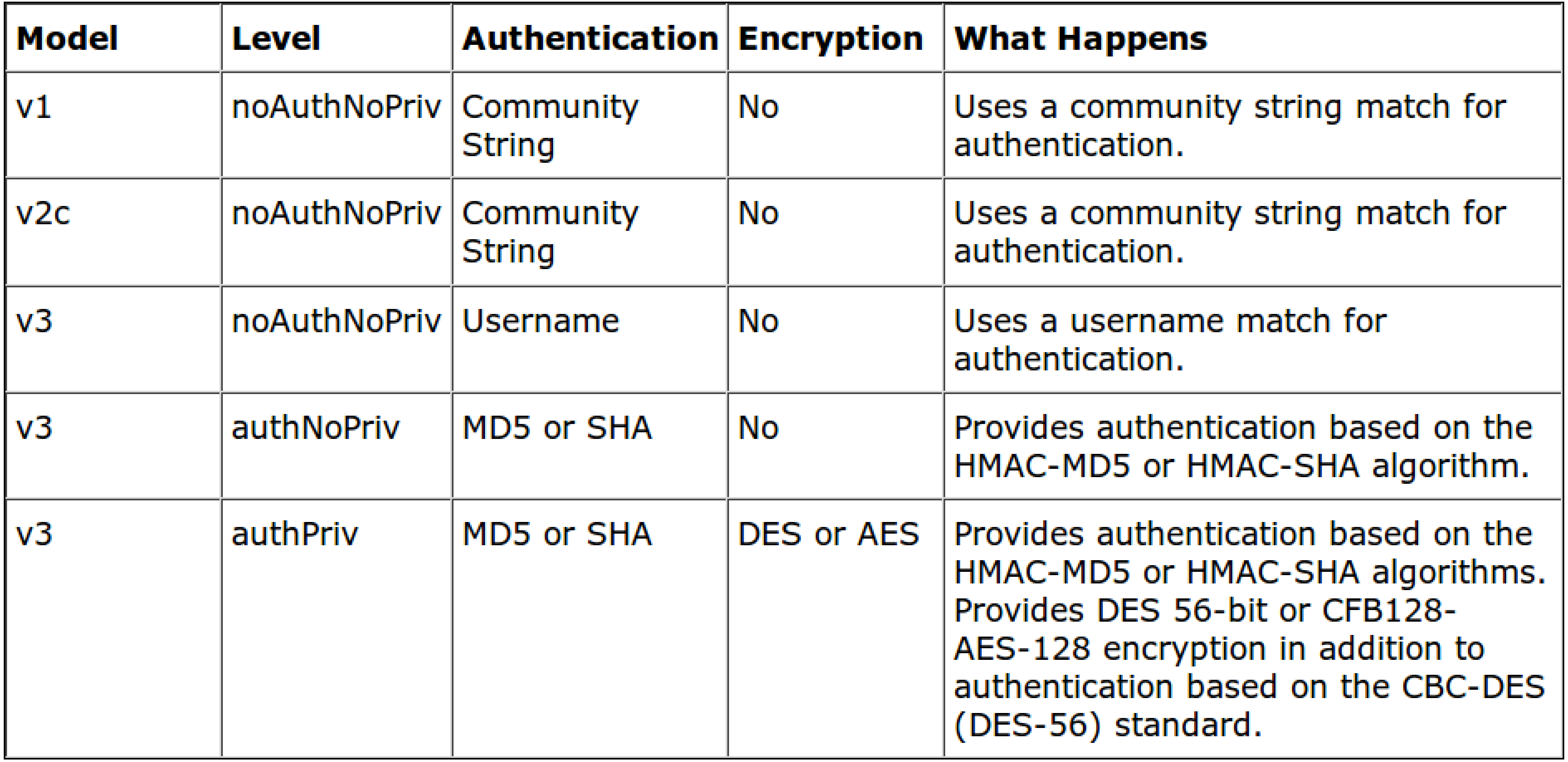
☺ Advantage: information only generated when required

☹ Disadvantage:

- ☹ More resources required in the managed equipment
- ☹ Traps can be useless
  - If many events occur, bandwitdh can be wasted with all traps (thresholds can solve)
  - Since the agent has only a limited scope of the network, NMS may already know about the events.

- Traps&Polling
  - Event occurs → trap is sent
  - Manager performs polling to obtain the rest of information
  - Manager also performs periodic polling, as backup

SNMP Versions



Model	Level	Authentication	Encryption	What Happens
v1	noAuthNoPriv	Community String	No	Uses a community string match for authentication.
v2c	noAuthNoPriv	Community String	No	Uses a community string match for authentication.
v3	noAuthNoPriv	Username	No	Uses a username match for authentication.
v3	authNoPriv	MD5 or SHA	No	Provides authentication based on the HMAC-MD5 or HMAC-SHA algorithm.
v3	authPriv	MD5 or SHA	DES or AES	Provides authentication based on the HMAC-MD5 or HMAC-SHA algorithms. Provides DES 56-bit or CFB128-AES-128 encryption in addition to authentication based on the CBC-DES (DES-56) standard.

SNMPv1: security and authentication

- In its initial version, authorization and authentication were based on the notion of “SNMP community string”
- The “words of community” identify the permissions of the machine accessing the agent: read-only ou read-write
- By default, all systems are configured with the community strings:
  - public (read-only)
  - private (read-write)
- The words are case sensitive.

SNMPv2c and SNMPv3 versions

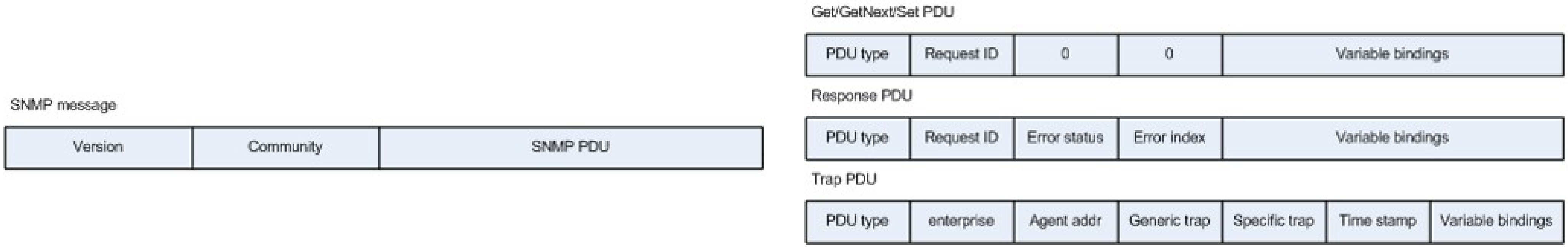
- SNMPv2 extensions
  - Structure of management information (SMI)
  - Manager-Manager capacity
  - New protocol operations
- SNMPv3 extensions
  - New message format
  - Message security
  - Access control

SNMPv3: Security

- Notion of “access control dependent on the user”
  - The agent mantains access rights information (policies) to different users in a data base
- Authentication: shared secret key
  - MD5 or SHA authentication passphrase hashes
- Privacy
  - Packet data may now be DES encrypted (future use allows additional encryption)
  - Passphrase defaults to authentication passphrase
  - Allows for unique Privacy passphrase
- Protection against replays: resort to nounces

SNMPv1 Message

- Version: SNMP version.
- Community: Community name, used for the authentication between an agent and the NMS.
  - In Get or GetNext operations, read community name is used for authentication;
  - In Set operation, write community name is used for authentication.
- Request ID: It is used to match a response to a request.
  - SNMP assigns a unique ID to each request.
- Error status: It is used in a response to indicate the errors when the agent processes the request
  - noError, tooBig, noSuchName, badValue, readOnly, and genErr.
- Error index: Provides the information of the variables that caused the error when an error occurs.
- Variable bindings: It is composed of a variable name and value.
- Enterprise: Type of the device that generates traps.
- Agent addr: Address of the device that generates traps.
- Generic trap: It includes coldStart, warmStart, linkDown, linkup, authenticationFailure, egpNeighborLoss and enterpriseSpecific.
- Specific trap: Specific trap information of a vendor.
- Time stamp: The amount of time between the time when the SNMP entity sending this message reinitialized and the time when traps were generated, that is, the value of sysUpTime.



SNMP message

Version	Community	SNMP PDU

Get/GetNext/Set PDU

PDU type	Request ID	0	0	Variable bindings

Response PDU

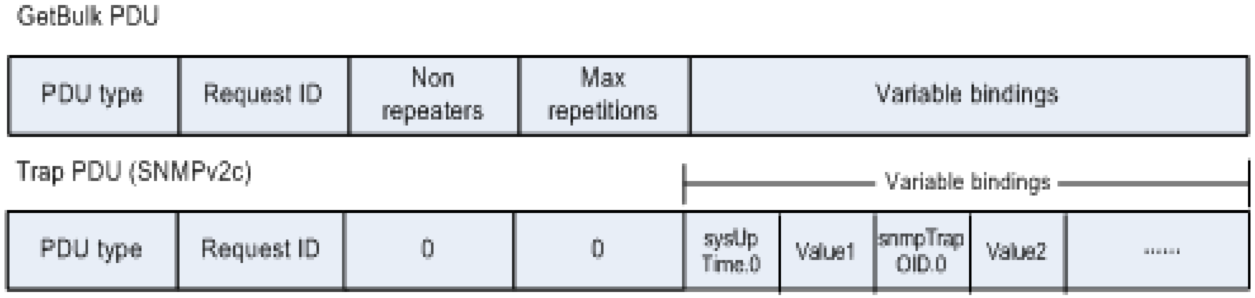
PDU type	Request ID	Error status	Error index	Variable bindings

Trap PDU

PDU type	enterprise	Agent addr	Generic trap	Specific trap	Time stamp	Variable bindings

SNMPv2c Message

- Compared with SNMPv1, GetBulk packets are added in SNMPv2c.
  - GetBulk operation corresponds to GetNext operation.
  - In a GetBulk operation, the setting of Non repeaters and Max repetitions parameters enables NMS to obtain data of many managed objects from an agent.
- In SNMPv2c, trap message format is different from that in SNMPv1.
  - SNMPv2c trap PDU adopts the format of SNMPv1 Get/GetNext/Set PDU, and sysUpTime and snmpTrapOID are used as variables in variable bindings to create a packet.



GetBulk PDU

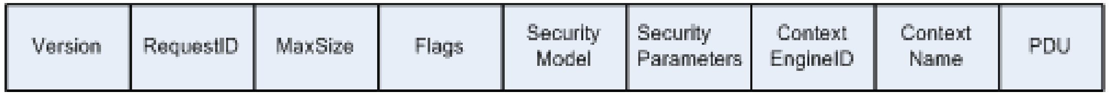
PDU type	Request ID	Non repeaters	Max repetitions	Variable bindings

Trap PDU (SNMPv2c)

				Variable bindings				
PDU type	Request ID	0	0	sysUp Time.0	Value1	snmpTrap OID.0	Value2	......

SNMPv3 Message

SNMPv3 message

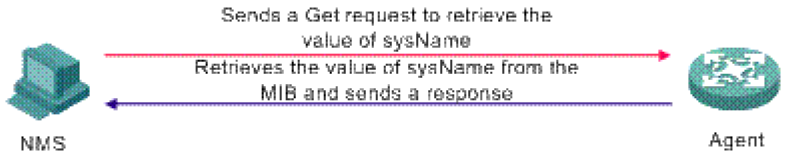


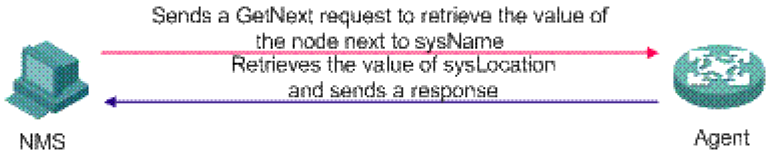
Version	RequestID	MaxSize	Flags	Security Model	Security Parameters	Context EngineID	Context Name	PDU

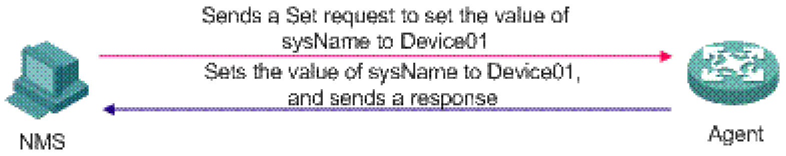
- SNMPv3 message format is modified, but the PDU format is the same as that in SNMPv2c.
- The entire SNMPv3 message can be authenticated, and EngineID, ContextName, and PDU are encrypted.
- RequestID, MaxSize, Flags, SecurityModel and SecurityParameters form the SNMPv3 message header.
- Fields:
  - RequestID
  - MaxSize: The maximum size of the message that the sender of the message can receive.
  - Flags: Message flag which occupies one byte. Only the lowest three bytes are valid. 0x0 indicates no authentication no privacy, 0x1 indicates authentication without privacy, 0x3 indicates authentication with privacy, and 0x4 indicates to send a report PDU.
  - SecurityModel: Message security model, in the range 0 to 3. 0 indicates any model, 1 indicates SNMPv1 security model, 2 indicates SNMPv2c security model, and 3 indicates SNMPv3 security model.
  - SecurityParameters includes the following fields:
    - AuthoritativeEngineID: Specifies the snmpEngineID of the authoritative SNMP engine involved in the exchange of the message, used for identification, authentication and encryption for an SNMP entity. This field refers to the source for a trap, response, or report, and to the destination for a Get, GetNext, GetBulk, or Set operation.
    - AuthoritativeEngineBoots: Specifies the snmpEngineBoots value at the authoritative SNMP engine involved in the exchange of the message. It indicates the number of times that this SNMP engine has initialized or reinitialized itself since its initial configuration.
    - AuthoritativeEngineTime: Specifies the snmpEngineTime value at the authoritative SNMP engine involved in the exchange of the message. It is used for time window check.
    - UserName: Specifies the user (principal) on whose behalf the message is being exchanged. Usernames configured on NMS and Agent must be the same.
    - AuthenticationParameters: A key used in authentication calculation. If no authentication is performed, this field is null.
    - PrivacyParameters: A parameter used in privacy calculation.
  - ContextEngineID: Uniquely identifies an SNMP entity. For a message received, this field decides how this message will be processed; for a message sent, this field is provided by the sender.
  - ContextName: Identifies a context. Must be unique within an SNMP entity.

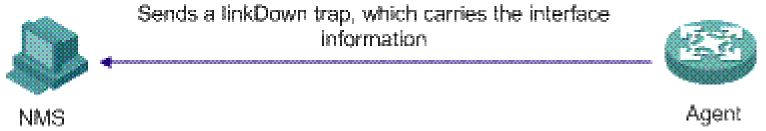
SNMP Operations

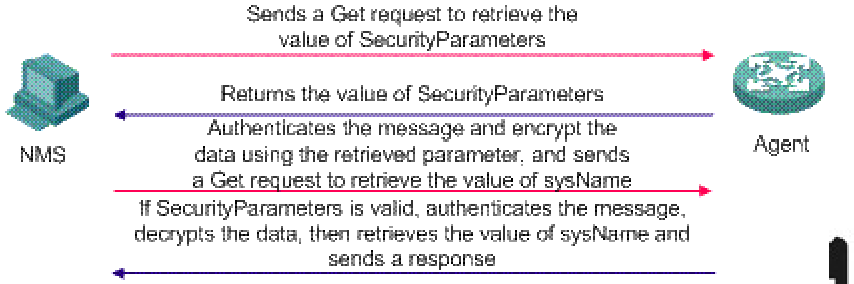
- SNMP provides the following five basic operations:
  - Get operation
    - Request sent by the NMS to the agent to retrieve one or more values from the agent.
  - GetNext operation
    - Request sent by the NMS to retrieve the value of the next OID in the tree.
  - Set operation
    - Request sent by the NMS to the agent to set one or more values of the agent.
  - Response operation
    - Response sent by the agent to the NMS.
  - Trap operation
    - Unsolicited response sent by the agent to notify the NMS of the events occurred.
- In SNMPv3 get operations are performed using authentication and encryption.


Sends a Get request to retrieve the value of sysName
Retrieves the value of sysName from the MIB and sends a response
NMS
Agent


Sends a GetNext request to retrieve the value of the node next to sysName
Retrieves the value of sysLocation and sends a response
NMS
Agent

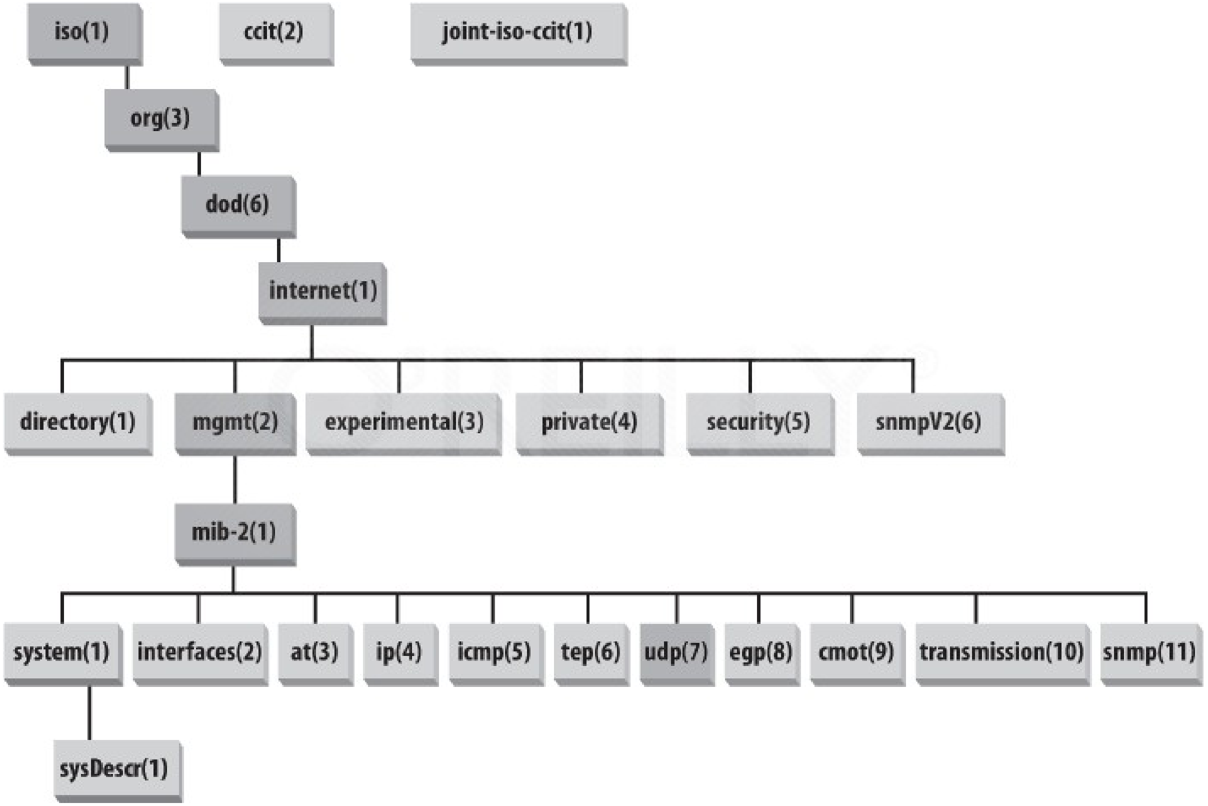

Sends a Set request to set the value of sysName to Device01
Sets the value of sysName to Device01, and sends a response
NMS
Agent


Sends a linkDown trap, which carries the interface information
NMS
Agent


Sends a Get request to retrieve the value of SecurityParameters
Returns the value of SecurityParameters
Authenticates the message and encrypt the data using the retrieved parameter, and sends a Get request to retrieve the value of sysName
If SecurityParameters is valid, authenticates the message, decrypts the data, then retrieves the value of sysName and sends a response
NMS
Agent

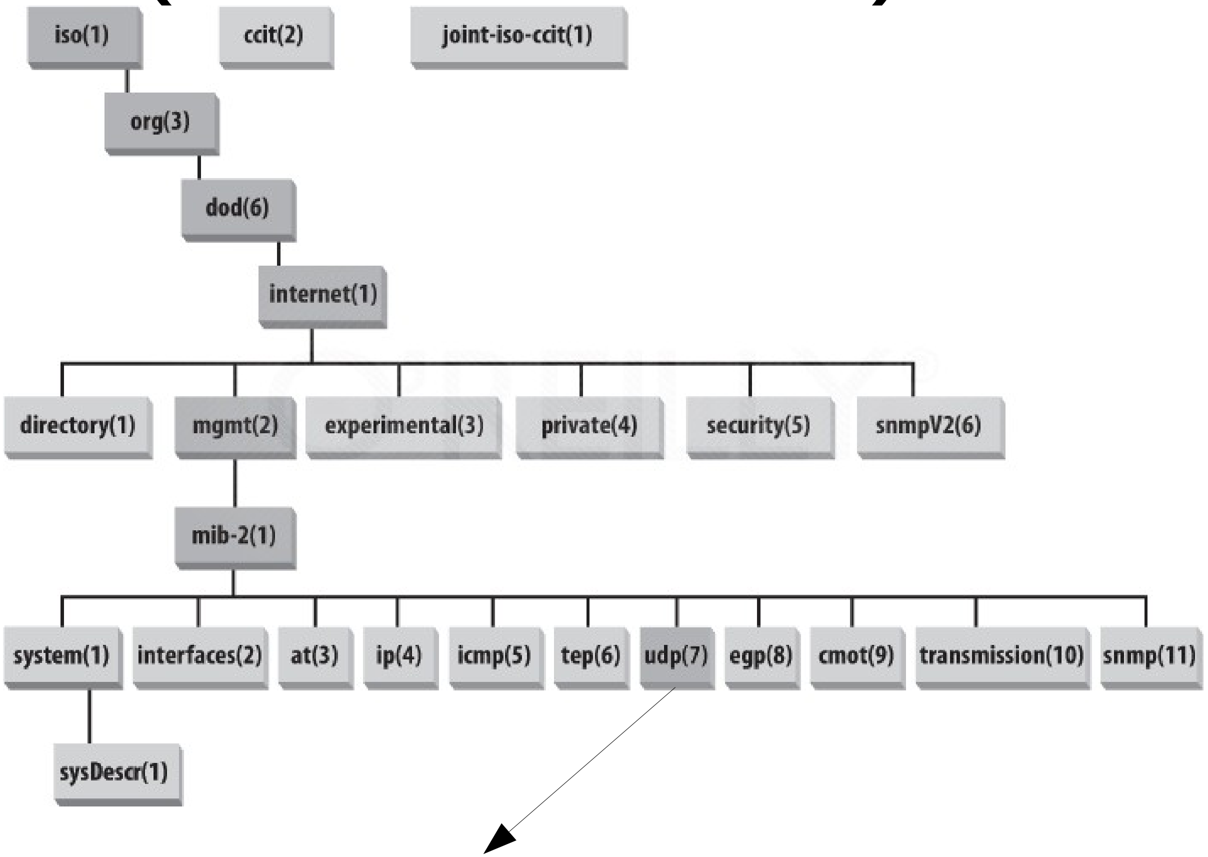
MIB Modules and Object Identifiers

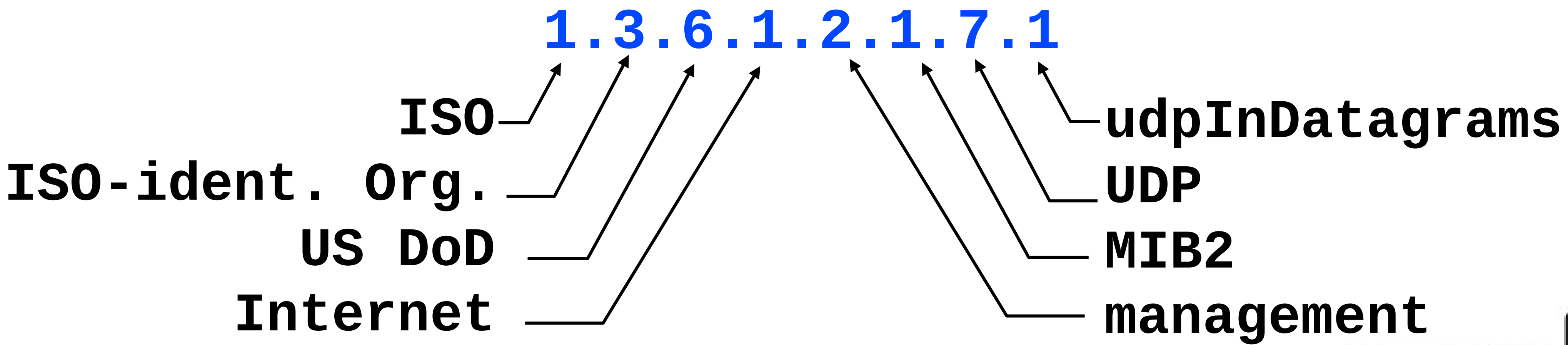
- An SNMP MIB module is a specification of management information on a device
- The SMI represents the MIB database structure in a tree form with conceptual tables, where each managed resource is represented by an object
- Object Identifiers (OIDs) uniquely identify or name MIB variables in the tree
  - Ordered sequence of nonnegative integers written left to right, containing at least two elements
  - For easier human interaction, string-valued names also identify the OIDs
    - MIB-II (object ID 1.3.6.1.2.1)
    - Cisco private MIB (object ID 1.3.6.1.4.1.9)
- The MIB tree is extensible with new standard MIB modules or by experimental and private branches
  - Vendors can define their own private branches to include instances of their own products



SNMP Names (numbers/OID)

- To nominate all possible objects (protocols, data, etc.) it is used an ISO Object Identifier (OID) tree:
  - Hierarchic nomenclature of objects
  - Each leaf of the tree has a name and number





1.3.6.1.2.1.7.1

ISO

ISO-ident. Org.

US DoD

Internet

udpInDatagrams

UDP

MIB2

management

SNMP MIBs

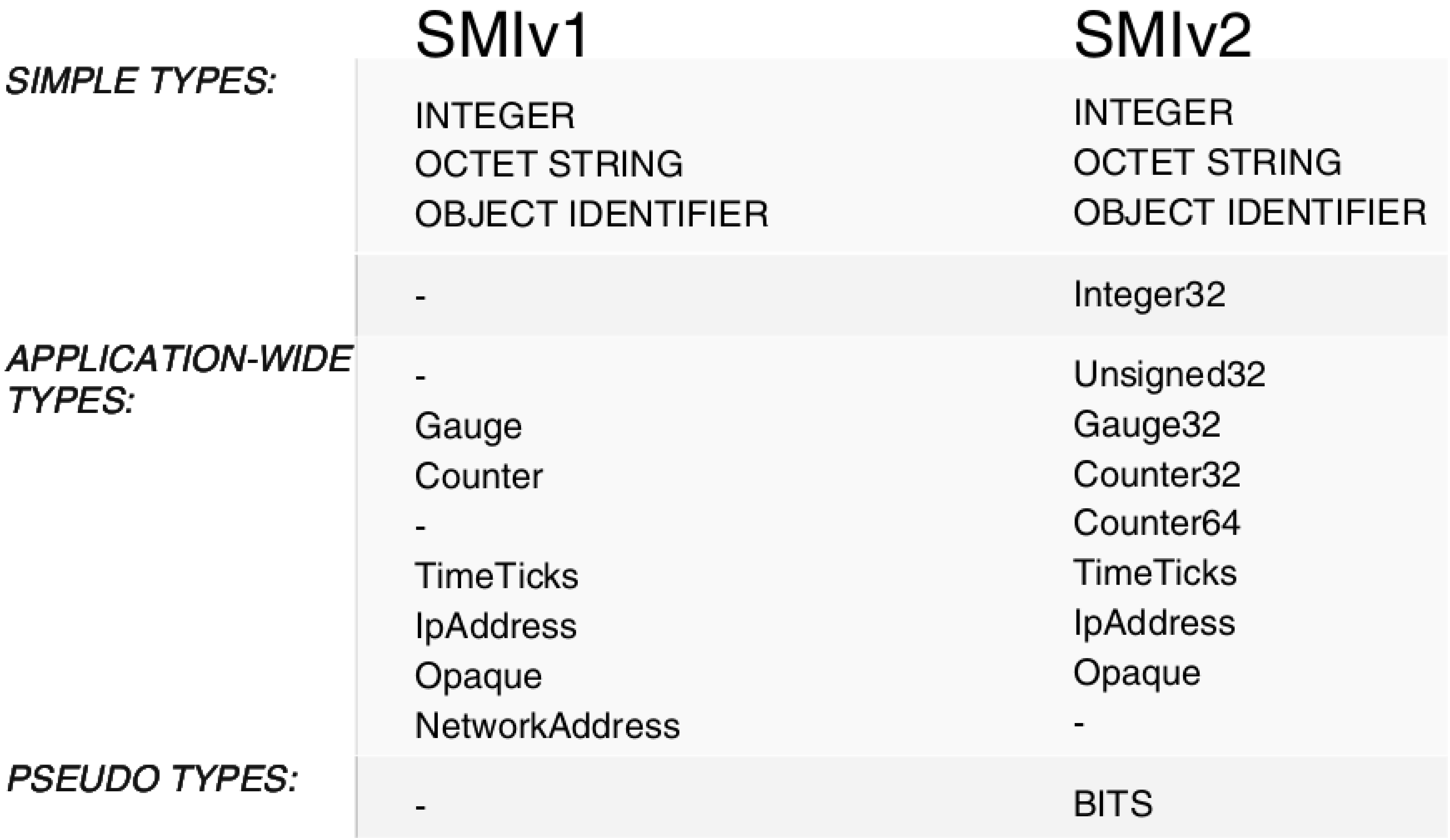
- Management Information Base (MIB): set of managed objects, used to define information from equipments, and created by the manufacturer
- Example: UDP module

Object ID	Name	Type	Comments
1.3.6.1.2.1.7.1	UDPInDatagrams	Counter32	Number of UDP datagrams delivered to users.
1.3.6.1.2.1.7.2	UDPNoPorts	Counter32	Number of received UDP datagrams for which there was no application at the destination port.
1.3.6.1.2.1.7.3	UDPInErrors	Counter32	The number of received UDP datagrams that could not be delivered for reasons other than the lack of an application at the destination port.
1.3.6.1.2.1.7.4	UDPOutDatagrams	Counter32	The total number of UDP datagrams sent from this entity.

SMI: Data language definition

- Well-defined sintax and semantics of management information
  - Type of basic data
    - INTEGER, Integer32, Unsigned32, OCTET, STRING, OBJECT IDENTIFIED, IPaddress, Counter32, Counter64, Guage32, Tie Ticks,Opaque...
  - Type of object
    - Type of data, status, semantic of the managed object
  - Module identification
    - Collection of objects inter-related in the MIB

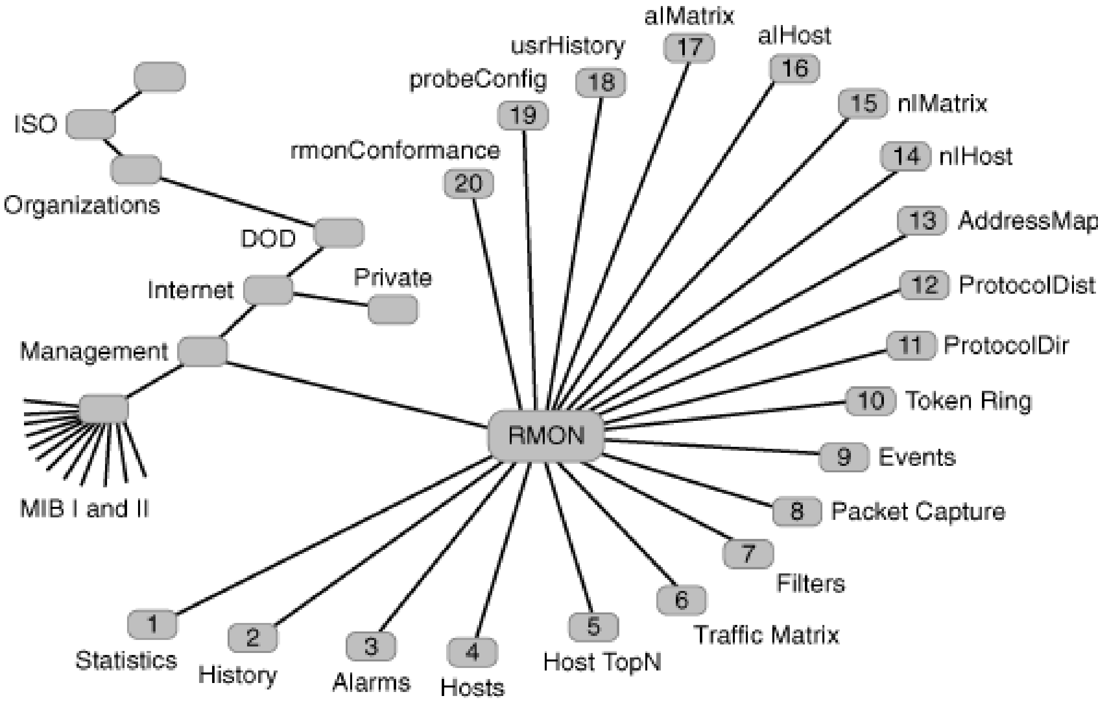
SMI: Data Types for Scalars



	SMIv1	SMIv2
SIMPLE TYPES:	INTEGER	INTEGER
	OCTET STRING	OCTET STRING
	OBJECT IDENTIFIER	OBJECT IDENTIFIER
	-	Integer32
APPLICATION-WIDE TYPES:	-	Unsigned32
	Gauge	Gauge32
	Counter	Counter32
	-	Counter64
	TimeTicks	TimeTicks
	IpAddress	IpAddress
	Opaque	Opaque
	NetworkAddress	-
PSEUDO TYPES:	-	BITS

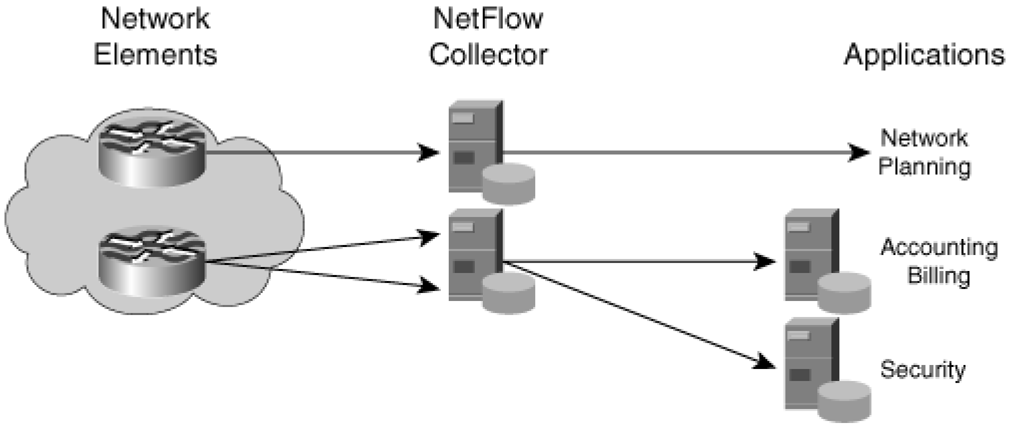
RMON

- RMON is a set of standardized MIB variables that monitor networks
  - All previously defined MIBs monitored only nodes
- RMON has 9 groups
  - Statistics, History, Alarm, Host, HostTopN, Matrix, Filter, Packet Capture, and Event
- The term RMON now is often used to refer to the concept of remote monitoring and to the entire series of RMON MIB extensions
- The main RMON MIB extensions are:
  - RMON 1 and RMON 2 MIBs - Remote Monitoring MIB versions 1 and 2
  - DSMON MIB - Remote Monitoring MIB Extensions for Differentiated Services
  - SMON MIB - Remote Network Monitoring MIB Extensions for Switched Networks
  - APM MIB - Application Performance Measurement MIB



Data Collection Protocols: NetFlow and IPFIX Export Protocols

- Cisco IOS NetFlow services give network administrators access to information about IP flows within their networks
  - An IP flow is defined as a unidirectional sequence of packets between given source and destination endpoints
  - IP flows are highly granular; flow endpoints are identified by IP address and by transport layer application port numbers
- NetFlow flow records are exported to an external device, a NetFlow collector
  - Can be used for a variety of purposes, including network management and planning, enterprise accounting, departmental chargeback, ISP billing, data warehousing, user monitoring and profiling, combating denial of service (DoS) attacks, and data mining for marketing purposes



- The IETF IPFIX stands for "IP Flow Information eXport," is an IETF effort to standardize an export protocol similar to NetFlow
  - A protocol that exports flow-related information
  - IPFIX protocol specifications are largely based on the NetFlow version 9 export protocol

NetFlow versions 1 and 5

- NetFlow v1/v5 packets are UDP/IP packets with a NetFlow header and one or more NetFlow data Records

IP header	UDP header	NetFlow header	NetFlow record	...	NetFlow record

Header format

byte 3	byte 2	byte 1	byte 0
version		count	
system uptime			
UNIX seconds			
UNIX nanoseconds			

byte 3	byte 2	byte 1	byte 0
version		count	
system uptime			
UNIX seconds			
UNIX nanoseconds			
flow sequence number			
engine type	engine ID	reserved	

Record format

byte 3	byte 2	byte 1	byte 0
source IP address			
destination IP address			
next-hop IP address			
input interface index		output interface index	
packets			
bytes			
start time of flow			
end time of flow			
source port		destination port	
pad		IP protocol	TOS
TCP flags	padding		
reserved			

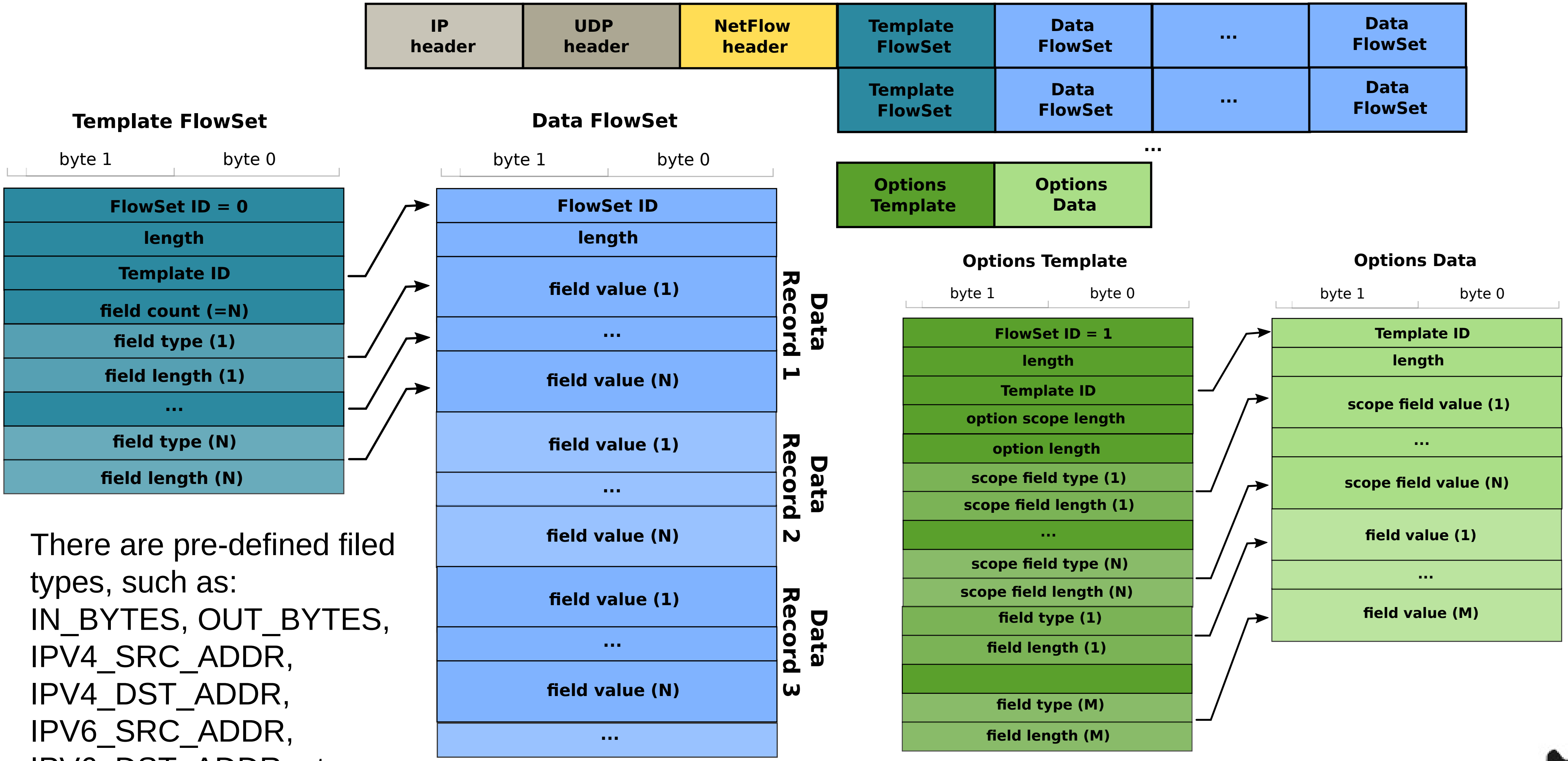
byte 3	byte 2	byte 1	byte 0
source IP address			
destination IP address			
next-hop IP address			
input interface index		output interface index	
packets			
bytes			
start time of flow			
end time of flow			
source port		destination port	
pad	TCP flags	IP protocol	TOS
source AS		destination AS	
src netmask length	dst netmask length	pad	

Version 1

Version 5

NetFlow version 9

- NetFlow v9 packets are UDP/IP packets with a NetFlow header, one or more Template FlowSets (may be suppressed, if sent previously), one or more Data FlowSets, and, optionally, an Options Template and Data Record.



IP header | UDP header | NetFlow header | Template FlowSet | Data FlowSet | ... | Data FlowSet

Template FlowSet | Data FlowSet | ... | Data FlowSet

...

Options Template | Options Data

Template FlowSet

byte 1 | byte 0

- FlowSet ID = 0
- length
- Template ID
- field count (=N)
- field type (1)
- field length (1)
- ...
- field type (N)
- field length (N)

Data FlowSet

byte 1 | byte 0

- FlowSet ID
- length
- Data Record 1: field value (1), ..., field value (N)
- Data Record 2: field value (1), ..., field value (N)
- Data Record 3: field value (1), ..., field value (N)
- ...

Options Template

byte 1 | byte 0

- FlowSet ID = 1
- length
- Template ID
- option scope length
- option length
- scope field type (1)
- scope field length (1)
- ...
- scope field type (N)
- scope field length (N)
- field type (1)
- field length (1)
- field type (M)
- field length (M)

Options Data

byte 1 | byte 0

- Template ID
- length
- scope field value (1)
- ...
- scope field value (N)
- field value (1)
- ...
- field value (M)

There are pre-defined filed types, such as: IN_BYTES, OUT_BYTES, IPV4_SRC_ADDR, IPV4_DST_ADDR, IPV6_SRC_ADDR, IPV6_DST_ADDR, etc...

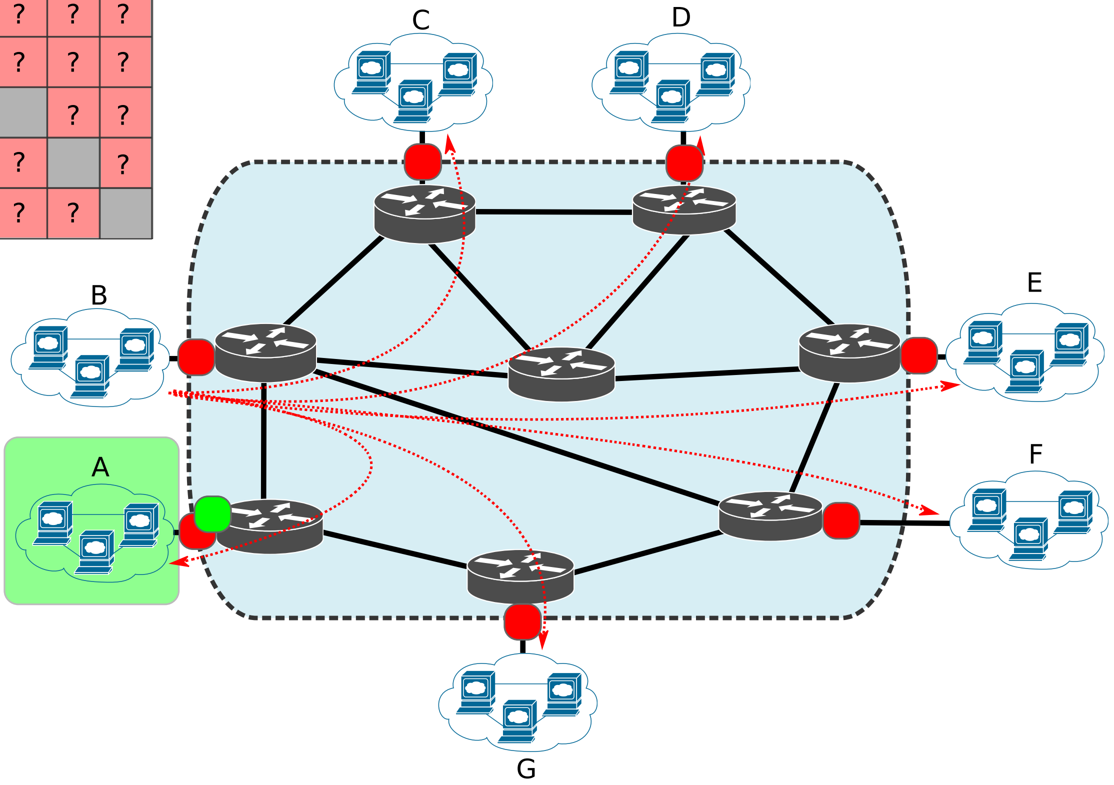
NetFlow Usage

- Used to characterize users/services in terms of amount of traffic.
  - Users/Groups (overall or per-app) → Applied in (V)LAN interfaces.
  - Services → Applied to data-center interfaces
- Used to characterize traffic destinations (to egress points) from a specific ingress point in a network: traffic matrices.
  - Ingress/Egress points may be:
    - Network access links (distribution layer L3SW, Internet access routers, user VPN server links),
    - Network core border links (core border routers),
    - BGP peering links (AS Border routers).
- Used to characterize "in network" routing.
  - Complex to implement and process.

NetFlow Deplyment

- Interfaces to monitor depend on objective:
  - Traffic matrix inference – all core border interfaces.
  - User/group flow generation inference - access interface from user/group.
- Egress vs. Ingress monitoring:
  - Traffic matrix inference – ingress OR egress.
  - User/group flow generation inference – both directions.

	A	B	C	D	E	F	G
A		?	?	?	?	?	?
B	?		?	?	?	?	?
C	?	?		?	?	?	?
D	?	?	?		?	?	?
E	?	?	?	?		?	?
F	?	?	?	?	?		?
G	?	?	?	?	?	?	



IPFIX (v10) and Flexible NetFlow

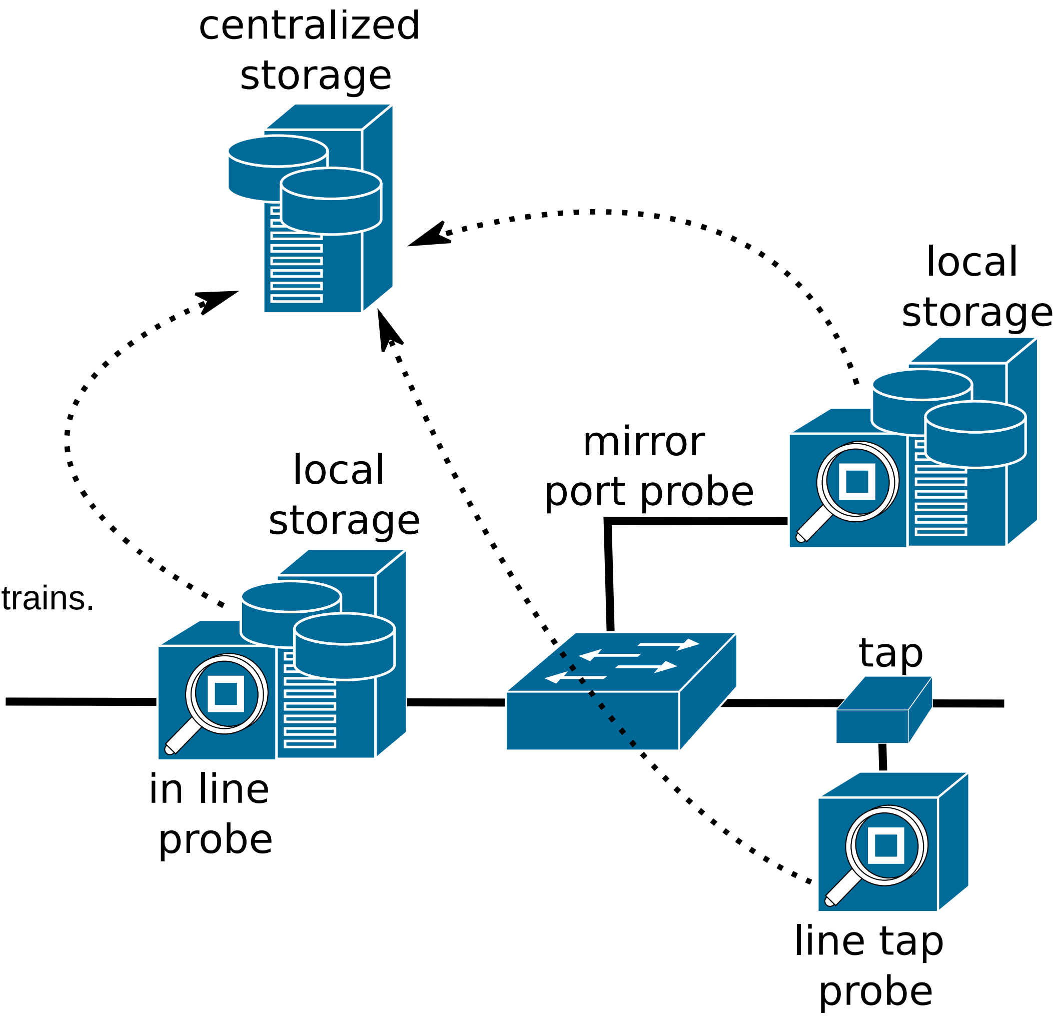
- IPFIX is very similar to NetFlow v9
  - Uses version 10 in a similar header.
  - Also has Templates and Data Records.
  - Also has Options Templates and Options Data Records.
- IPFIX made provisions for NetFlow v9 and added support for it.
  - IPFIX lists an overview of the “Information Element identifiers” that are compatible with the “field types” used by NetFlow v9.
- IPFIX has more filed types than the ones defined for NetFlow v9.
  - Also allows a vendor ID to be specified which a vendor can use to export proprietary/generic information.
- IPFIX allows for variable length fields.
  - Useful to export variable size strings (e.g., URLs).
- NetFlow v9 extension “Flexible NetFlow” aims to be equally flexible as IPFIX.

sFlow and jFlow

- sFlow
  - Uses sampling techniques designed for providing continuous site-wide (and enterprise-wide) traffic monitoring of high speed switched and routed networks.
  - Allow monitoring network traffic at Gigabit speeds and higher.
  - Allow to scale the monitoring of tens of thousands of agents from a single sFlow collector.
  - Supported by multiple vendors.
    - Including Cisco in
- jFlow is used in Juniper equipments.
  - Similar to NetFlow, however version 9 it also allows the usage of flow sampling techniques

Network Passive Probing
Packet Capturing

- User for:
  - Specific and detailed data inference,
  - Infer small and medium timescale dynamics.
- Probe types
  - Switch mirror port,
  - In-line,
  - Network tap.
- Filtering/sampled by
  - User/terminal address/VLAN/access port,
  - Group address/VLAN/access port,
  - Protocols (UDP/TCP),
  - Upper layer protocols,
    - Hard to identify due to encryption and legal/privacy constrains.
  - UDP/TCP port number/range.
- Data processing
  - Packet/byte count,
  - Flow count,
  - IP addresses and port distribution,
  - App/service statistics and distribution.
- Local vs. Centralized storage and processing.
  - Data upload to centralized point should not have impact on measurements.
  - Local storage/processing requires probes with more resources.

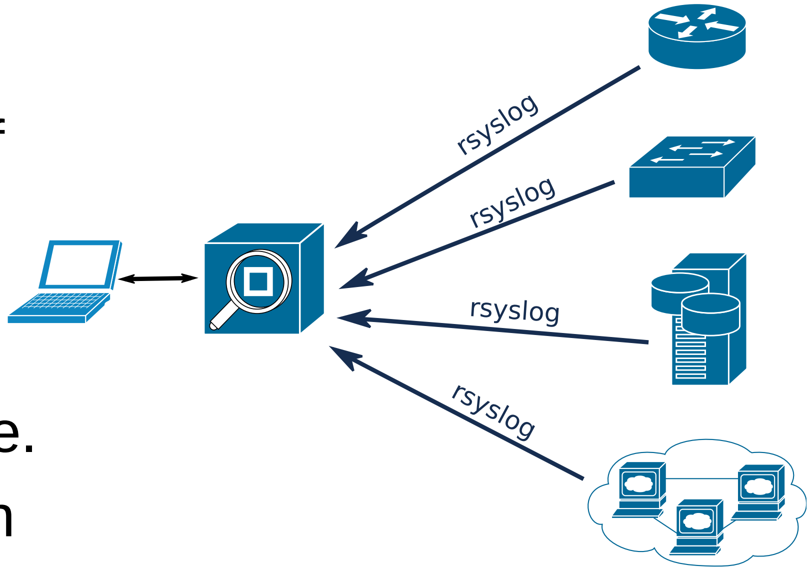


Data Collection Protocols: PSAMP

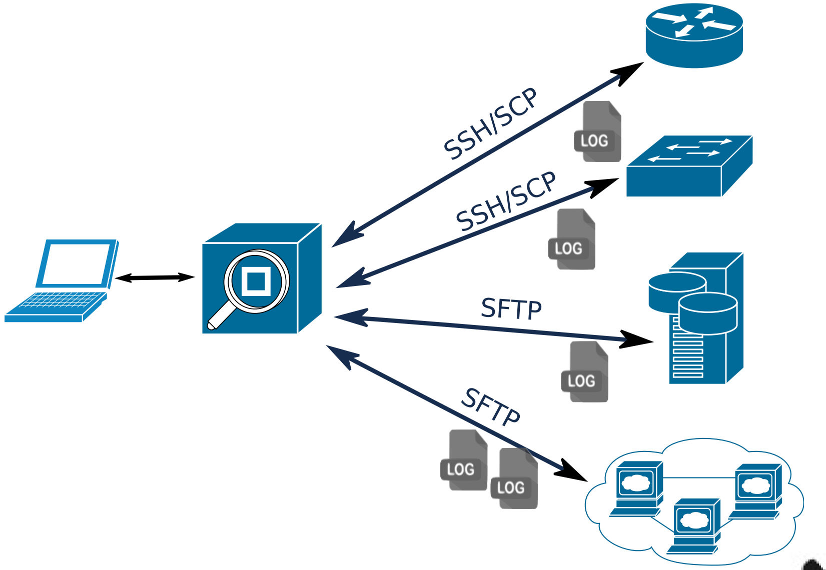
- The Packet Sampling (PSAMP) working group history started soon after the IPFIX working group creation:
- There was a clear need to define a standard set of capabilities for network elements to sample subsets of packets by statistical and other methods
  - Specifically, on the high-end routers where monitoring every packet was practically impossible
- The focus of the working group was to
  - Specify a set of selection operations by which packets are sampled
  - Specify the information that is to be made available for reporting on sampled packets
  - Describe protocols by which information on sampled packets is reported to applications
  - Describe protocols by which packet selection and reporting are configured
- For export of PSAMP packet information, the IPFIX protocol is used

Log Files Access

- rsyslog
  - Able to accept inputs from a wide variety of services, transform them, and output the results to diverse network destinations.
    - Over TCP and/or SSL/TLS.
  - Timing controlled by monitored node/device.
  - Many post- and cross-processing tasks can be made on the monitored node/device.

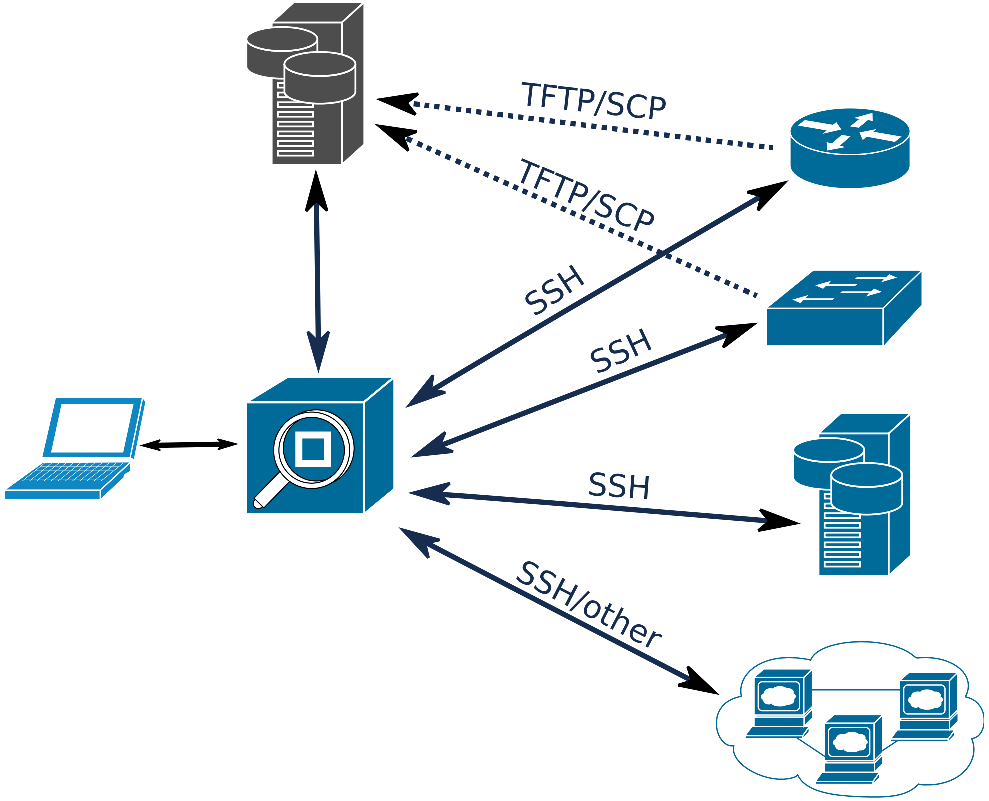


- Direct access to log files
  - Using any remote access to remote files.
    - Requires special permissions.
  - SSH/SCP, SFTP, etc...
  - Timing controlled by central point.
  - Requires all heavy post- and cross-processing in a central point.



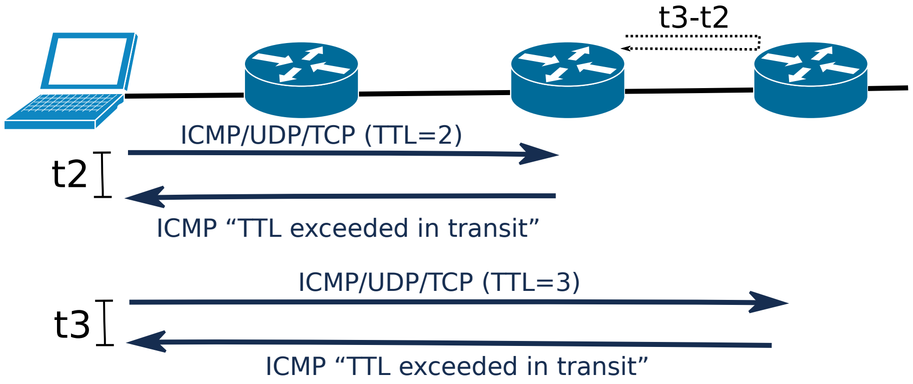
Remote CLI Access

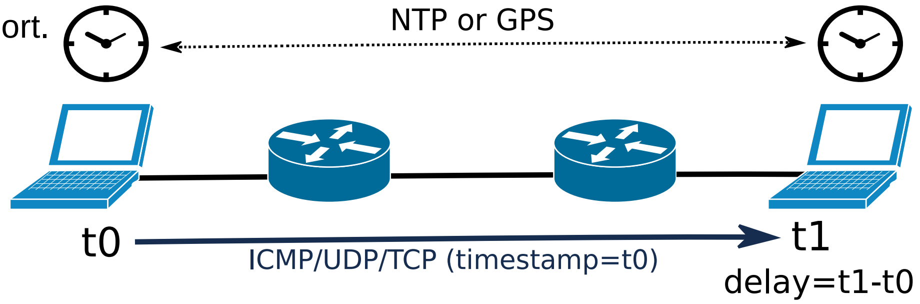
- Using a remote console to devices,
  - Using SSH, telnet (insecure), or proprietary protocols,
  - Retrieve configurations and device's processes status.
  - Devices can also upload configurations to a central point.
    - Using TFTP (insecure) or SFTP/SCP (many devices do not support it).
- Send “show” like CLI commands, retrieve output, parse information.

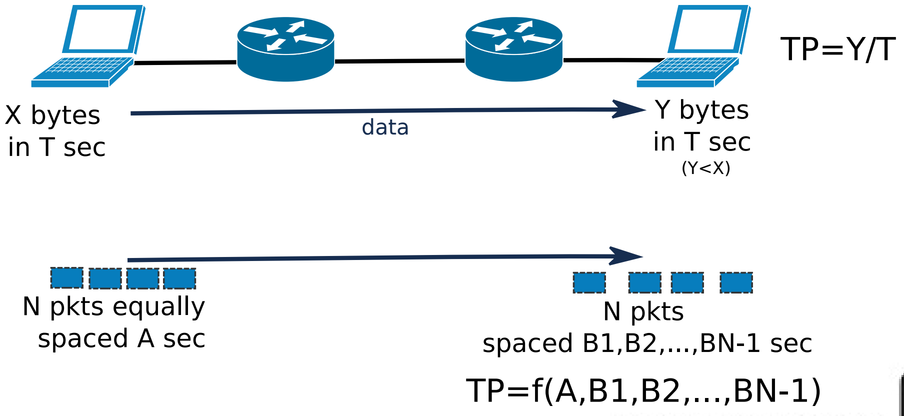


Active Measurements

- Two-way delay/jitter
  - End-to-end and middle hop.
  - Requires the control of only one end.
  - Ping and traceroute like solutions.
    - Requires that middle nodes respond to probes.
      - ICMP “TTL exceeded in transit” message.
    - ICMP, UDP or TCP.
      - UDP/TCP allows to test QoS (DiffServ) by IP/Port.
- One-way delay/jitter
  - End-to-end.
  - Requires control of both ends and clock synchronization.
    - May be complex/impossible for close nodes (low delay).
- End-to-end throughput
  - Requires control of both ends.
  - Directly sending/receiving large amounts of data.
  - Indirectly using packet train techniques.
    - Prone to errors.


t3-t2
ICMP/UDP/TCP (TTL=2)
t2
ICMP “TTL exceeded in transit”
ICMP/UDP/TCP (TTL=3)
t3
ICMP “TTL exceeded in transit”



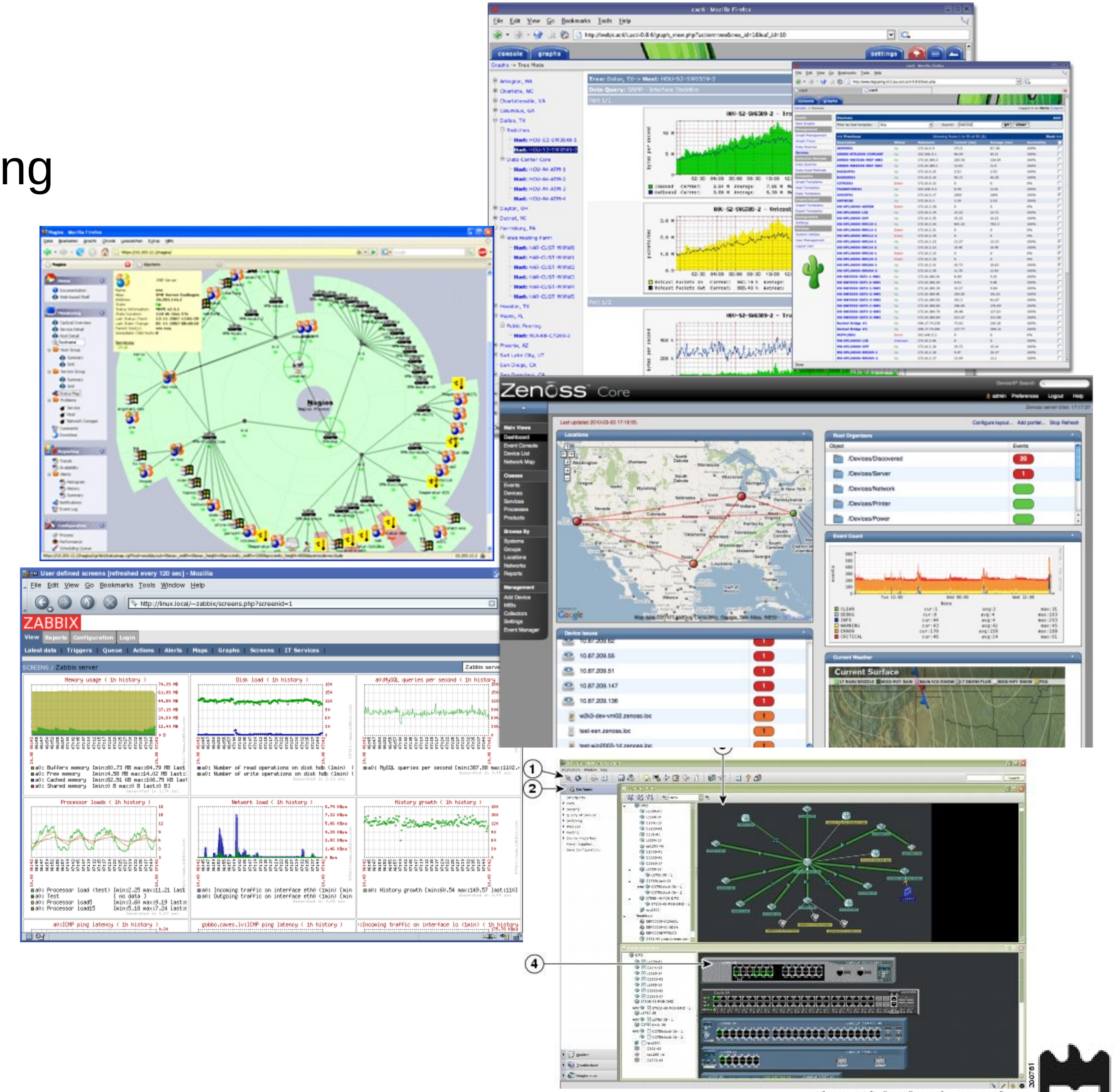


Monitoring and Management Tools

- Integrated Monitoring & Management
  - CiscoWorks
    - SNMP and Cisco Discovery Protocol (CDP) based
  - Allows equipment reconfiguration and firmware update
- Monitoring (and Alert)
  - Link usage, flow analysis, traffic matrices, protocol usage
  - SNMP and/or pcap lib based tools
    - Multiple proprietary tools
    - Open source/freeware
      - Cacti, Nagios, NTOP, OpenNMS
  - Netflow based tools
- Management
  - Console/scripting based tools
- In Research (advanced traffic capture and processing capabilities)
  - Pcap lib based tools
    - TCPdump, Wireshark, Tshark, etc...
  - DAG technology (cards+software)

Open Source/Commercial Tools

- Cacti and Cricket
  - SNMP + RRDtool graphing
- Nagios
  - SNMP + HTTP + SSH + DB + other
  - Plugins
- Zenoss
- Zabbix
- Cisco Network Assistant
- Etc...

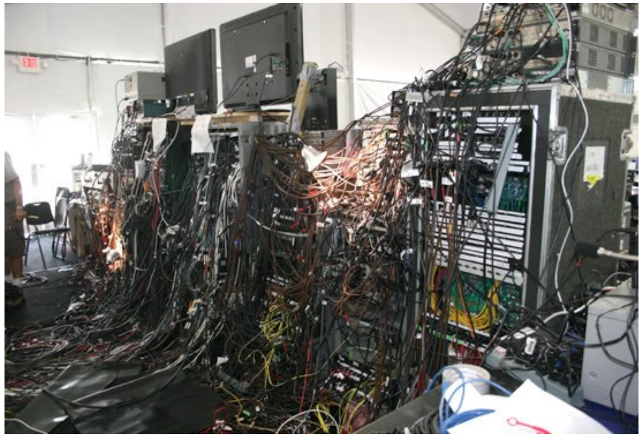


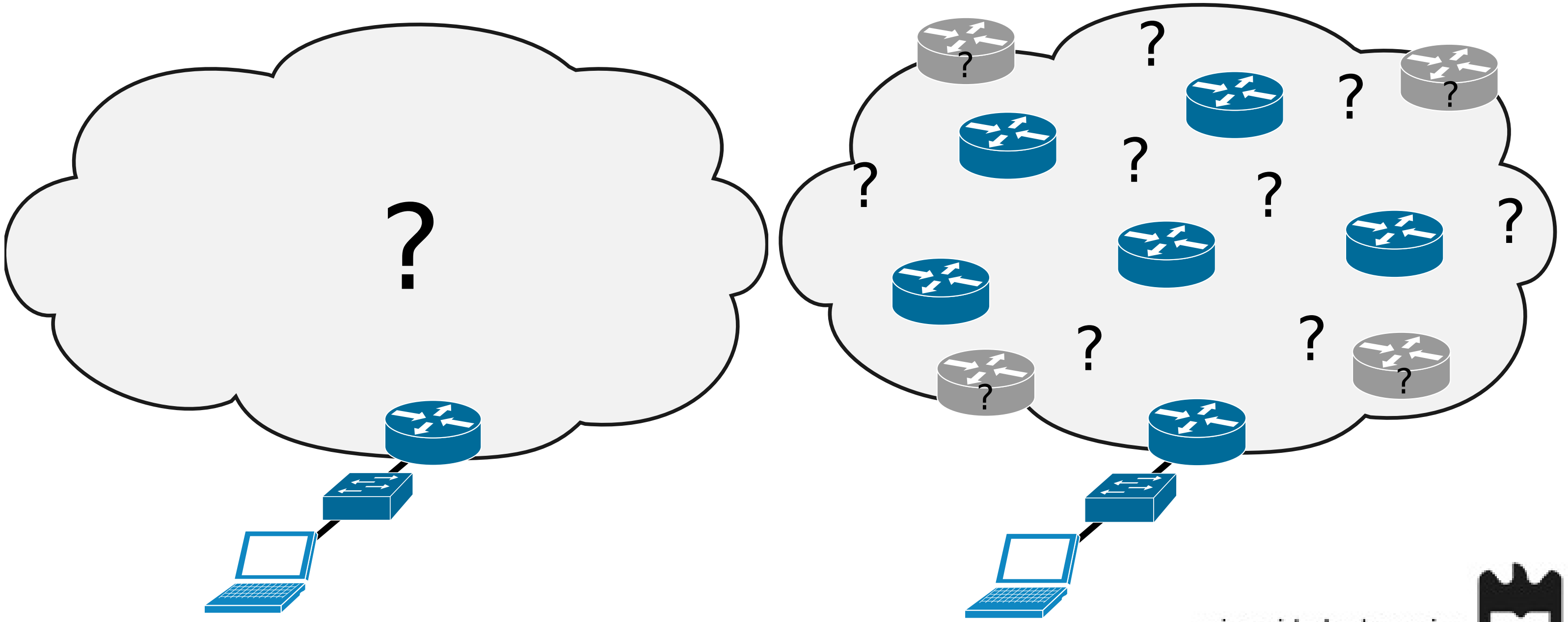
Challenges

1. Nodes discover and network links identification.
2. Infer link utilization and packet losses in nodes.
3. Estimate end-to-end two-way delay/jitter and identify delay/jitter inducing nodes.
4. Infer traffic matrix in ISP core.
5. Infer traffic matrix in Corporate Network core.
6. Detailed core routing.
7. User traffic profile
   - Usage over time profile (bytes, packets, flows, ports, etc...).
   - Destinations (per IP, per UDP/TCP ports).
8. User groups traffic profile
   - Same as for a single user, with IP address, VLAN, and/or access port (physical or virtual) aggregation.
9. Application/service traffic profile
   - Server side: known server/service addresses, UDP/TCP ports, and/or access ports.
   - Client side: requires DPI or known UDP/TCP ports mapping.

Challenge 1

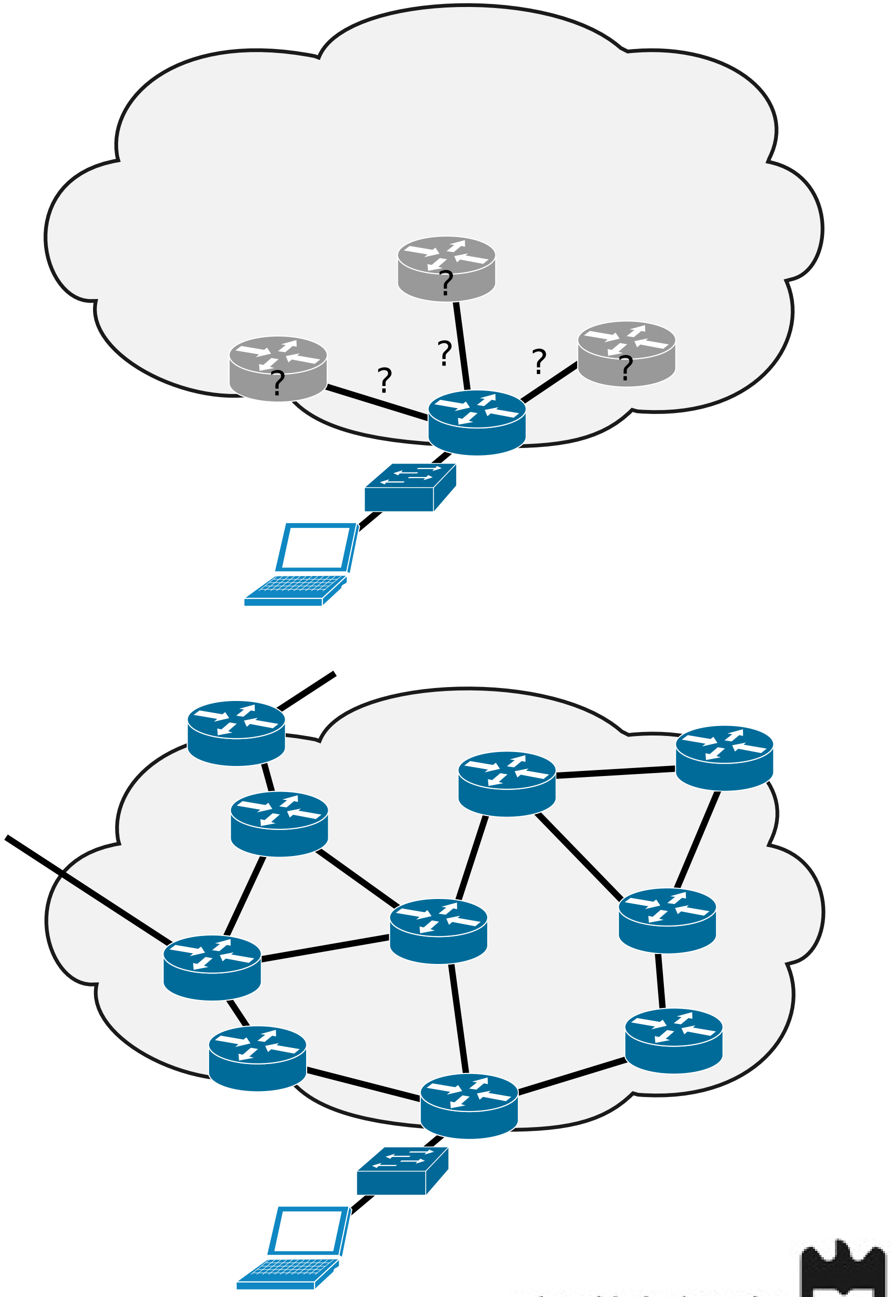
- Nodes discover and network links identification.
  - Two possible stating points:
    - Complete unknown in terms of network nodes and links.
    - Some network nodes may be know.
      - Unknown nodes may be active and interacting with known devices.
  - Physically inspecting/discovering may be hard in terms of complexity and/or geographic distance.





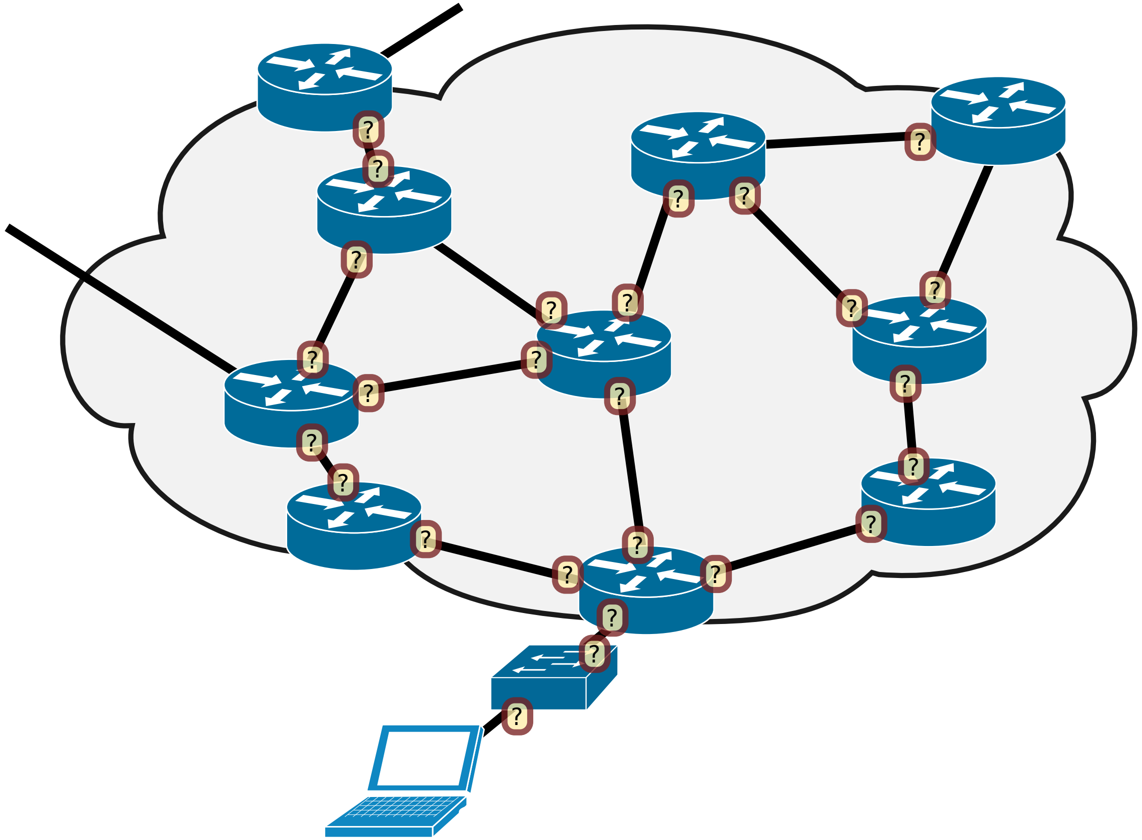
Challenge 1 (possible solution)

- Minimal knowledge about nodes must include IP addresses and remote access and/or console port credentials (when physical access is possible).
  - If SNMP community and credentials are unknown, using CLI, configure SNMP (or recover configuration) in device(s).
- Starting from known node(s):
  - Using SNMP, identify active interfaces, respective addresses and interconnection IP networks (prefixes and masks).
  - Using SNMP, analyze ARP and Neighbor tables to discover MAC addresses from unknown active nodes.
  - Using SNMP, analyze routing processes and routing/forwarding tables to discover unknown active nodes.
    - Proprietary protocols can also be used, e.g., Cisco CDP.
  - Access (remotely or physically) newly discovered nodes.
    - Credentials may be know even for “lost” unknown nodes!
    - If not, a password reset must be made using the console port!
      - Requires physical access.
  - Correlate information to build network graph.
  - Repeat process until no new information is discovered.

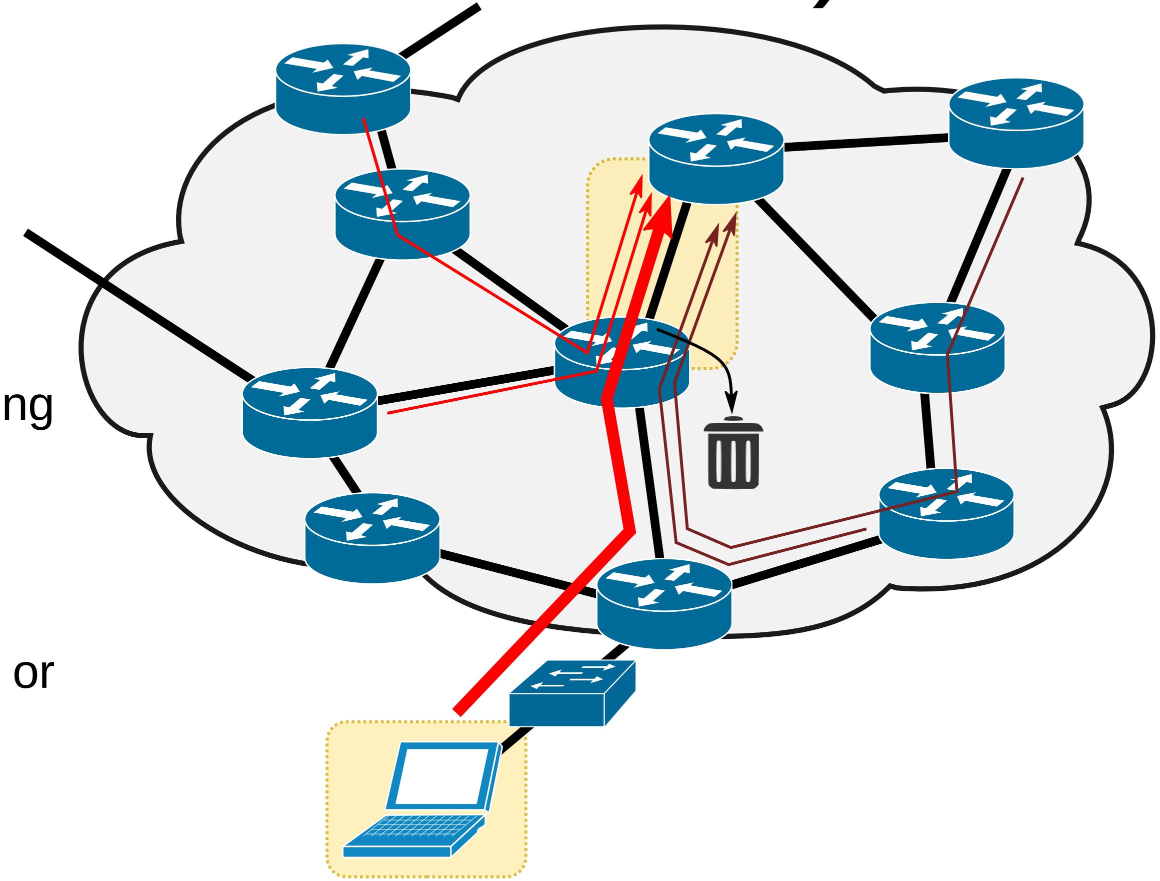


Challenge 2

- Infer link utilization and packet losses in nodes.
  - To identify links and node cards requiring upgrade.
  - To identify routing anomalies.
  - To identify sources with anomalous behavior.



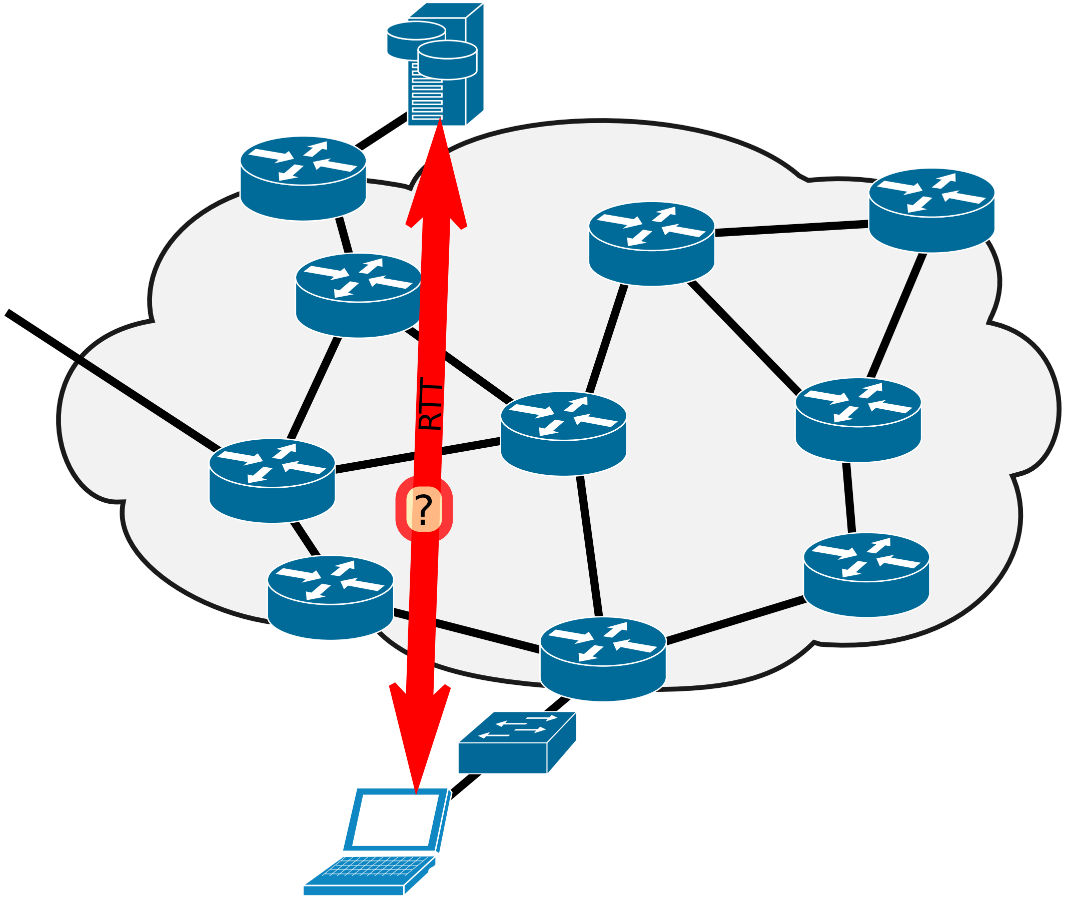
Challenge 2 (possible solution)



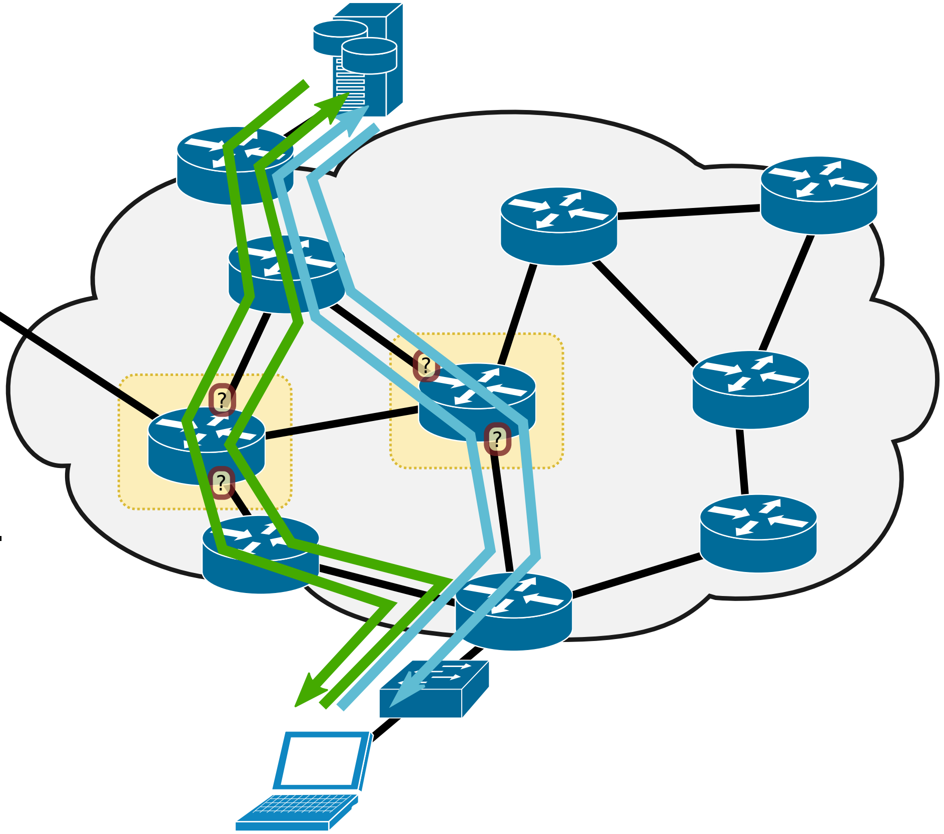
- Using SNMP in all know nodes/interfaces,
  - Infer link utilization over time,
  - Infer packet losses over time,
  - Recover (if necessary) routing/forwarding tables.
- Observable very high link utilization and/or high packet losses.
  - Cause 1: Low capacity or defected link or node card.
    - Symptom: A single link with very high utilization.
    - Actions: Upgrade card/link.
  - Cause 2: Poor routing decisions
    - Symptom: Multiple links with very high and very low utilization within the network.
    - Actions: Confirm poor routing analyzing routing/forwarding tables. → Correct routing mechanisms.
  - Cause 3: High traffic generation (due to DoS attack or negligence of usage)
    - Symptom: High utilization within a small group of links.
    - Actions: Identify traffic path with SNMP link utilization data or using NetFlow. → Locate source of the traffic. → Block traffic.

Challenge 3

- Estimate end-to-end two-way delay/jitter.
- Identify delay/jitter inducing nodes.
  - Between two point the RTT is higher than expected.
  - Identify which node/link is responsible.



Challenge 3 (possible solutions)

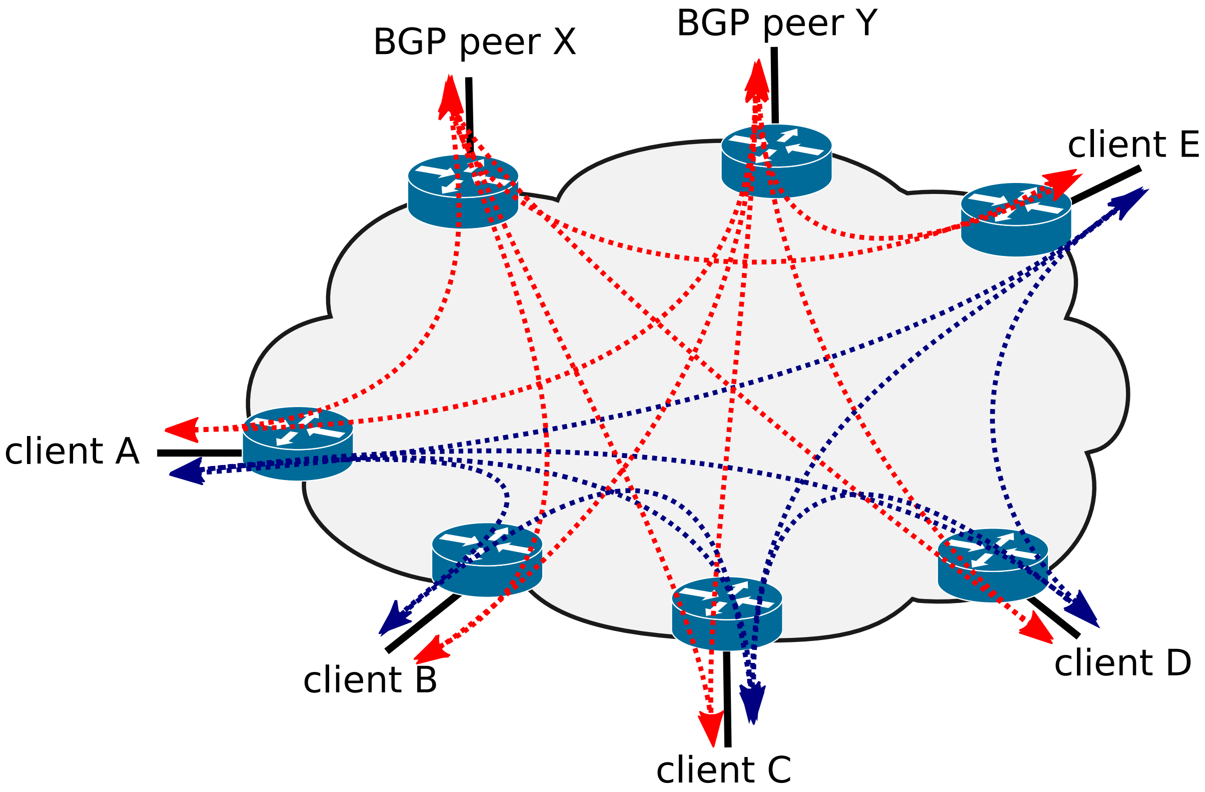


- RTT stability inference.
  - RTT may be constantly high, imposed by a constant routing path using problematic nodes.
  - RTT may me occasionally high, caused by an occasional routing decision (instable routing) and/or node problem.
- Perform multiple trace routes from an end-point or end-point gateway, to the other end-point, and vice-versa.
  - Verify routing path constancy in both directions.
  - Identify hops with high RTT increment.
  - Identify problematic nodes addresses.
- Using SNMP, analyze the size of output interface queue(s) for all nodes in the path (or paths) or just the ones previously identified as problematic.
  - To confirm the exact node and node problem (card, link, etc...).

- Actions
  - Upgrade/fix problematic nodes/cards/links.
  - Fix instable routing.

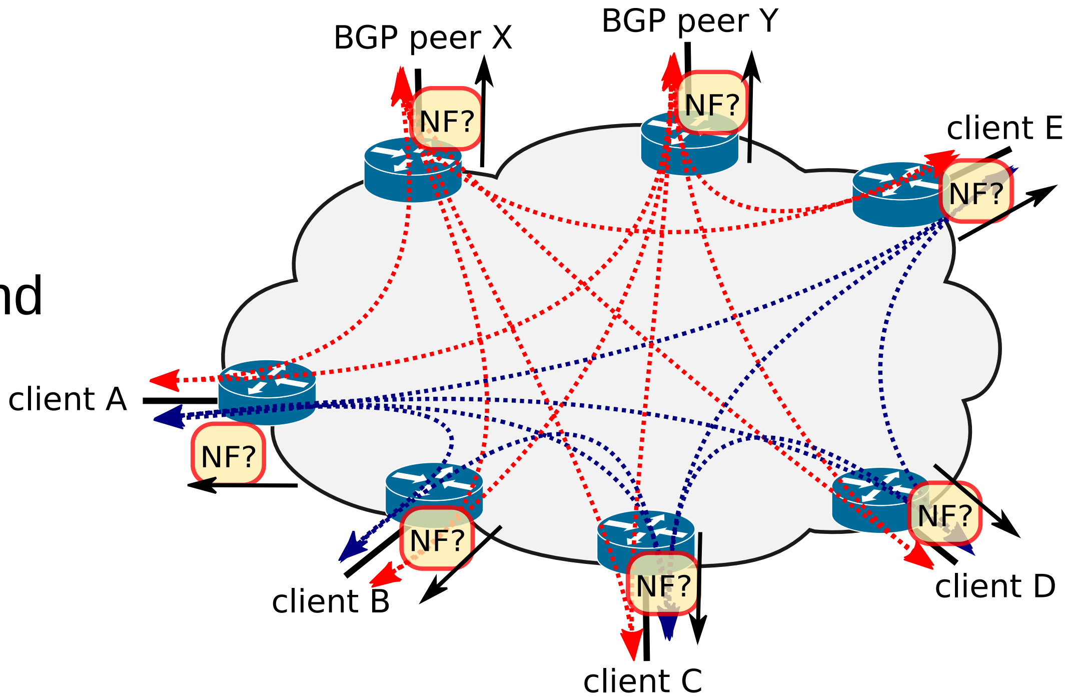
Challenge 4

- Infer traffic matrix in ISP core.
  - Amount of traffic between clients and external BGP peers (Internet traffic).
  - Amount of traffic between clients (internal traffic).
    - Bytes, packets, flows, sessions, etc...
    - Total, per UDP/TCP port, over time profile, etc...



Challenge 4 (possible solution)

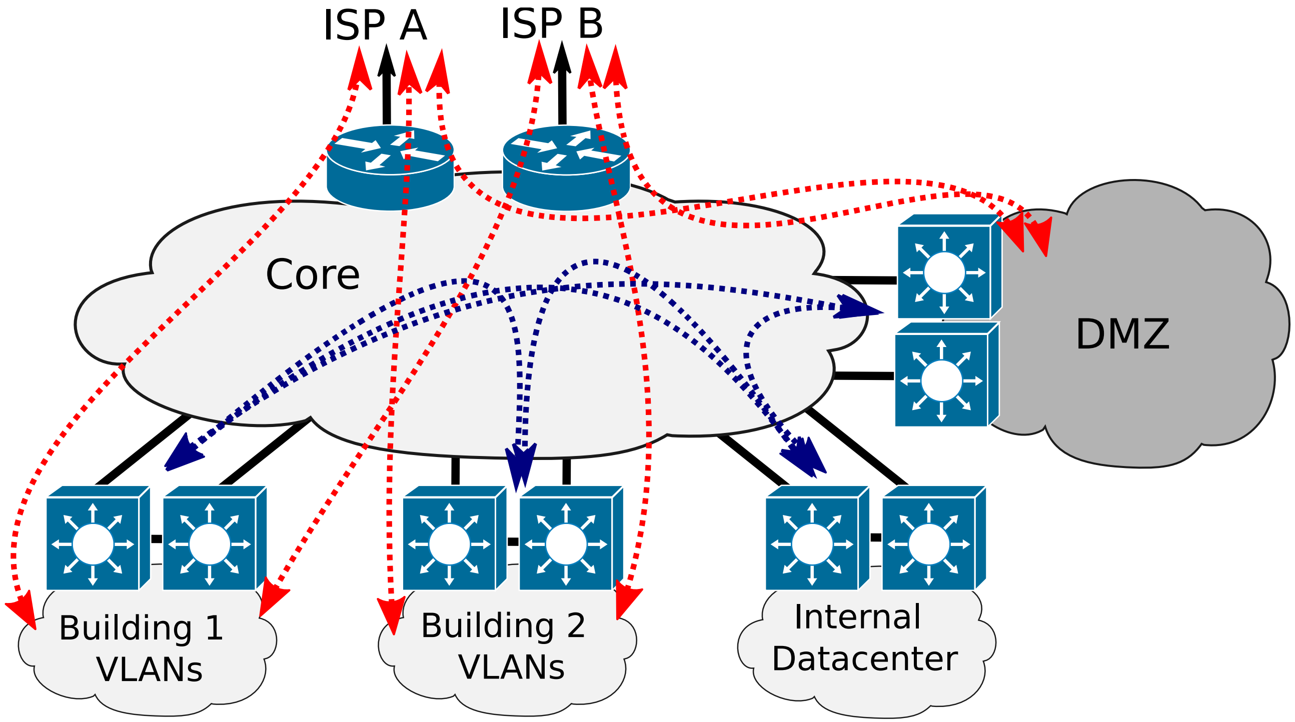
- Identify flows ingressing or egressing the core using NetFlow.
  - Best approach is to choose only one direction to avoid repeated reports and post-filtering.
  - Client terminated flows are easily identifiable/differentiated based on known public IP addresses from clients and or reporter ID (router, interfaces indexes).
  - Internet (BGP peers) terminated must be differentiated based on reporter ID (router, interfaces indexes).
  - May be required the usage of BGP information (version 5/9) and MPLS (version 9) information, when private AS exist, and when MPLS and/or MPLS/VPN connections exist.



- Complex scenarios (private AS, MPLS, MPLS-VPN) requires complex data processing of NetFlow exports.

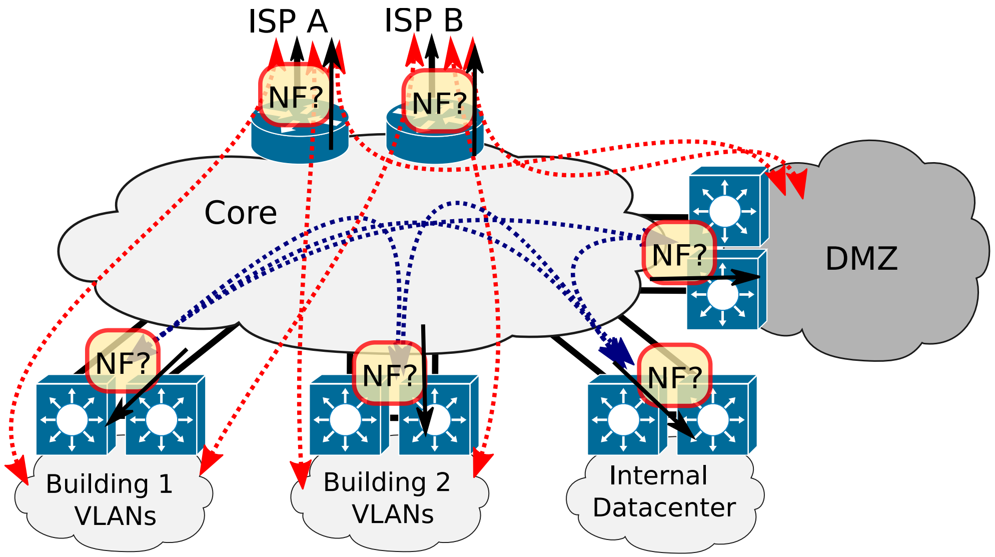
Challenge 5

- Infer traffic matrix in Corporate Network core.
    - Amount of traffic between (V)LAN and Internet accesses (users external traffic).
    - Amount of traffic between DMZ and Internet accesses (services external traffic).
    - Amount of traffic between (V)LAN and DMZ (internal traffic).



Challenge 5 (possible solution)

- Identify flows ingressing or egressing the network core using NetFlow.
  - Monitor VLAN and/or physical interfaces on core edge.
  - (V)LAN terminated flows are easily identifiable/differentiated based on known public IP addresses from (V)LAN and or reporter ID (router, interfaces indexes).
  - Internet accesses (Internet) must be differentiated based on reporter ID (ISP access routers, interfaces indexes).



Challenges 6, 7 and 8

- Infer traffic profiles over time,
  - Counts of Bytes, packets, errors, flows, data-streams (multiple flows), calls, etc...,
    - In consecutive time windows.
  - For individual users:
    - SNMP obtained counters in access device (switch, modem, AP) port.
    - Passive probing (packet capture) for user device addresses.
  - For user groups:
    - SNMP obtained counters in distribution device (switch) with access to group.
    - Passive probing (packet capture) for group device addresses (e.g, VLAN network prefix).
  - For applications/services
    - SNMP obtained counters in control device (e.g., firewall) with port/address/service filter.
    - Passive probing (packet capture) for specific client/server addresses and/or ports.
      - – May require DPI if port numbers are unknown.