



Cflowd

# Installing cflowd

- ◆ download the latest cflowd from [www.caida.org](http://www.caida.org)
- ◆ also download the arts + + package (arts + + package is installed by default if you are using KDE / X11).
- ◆ Compile cflowd
  - ./configure
  - make
  - make install
  - make clean

# Documentation

- ◆ refer to CAIDA documentation for details

<http://www.caida.org/tools/measurement/cflowd/configuration/configuration.htm>

- ◆ Example configuration files are included in the cflowd package in the etc/ subdirectory. They're named cflowd.conf.example and cfdcollect.conf.example.

# cflowd

- ◆ cflowd, cflowdmux and the local utilities (cfdases, cfdnets, et. al.) all read configuration information from cflowd.conf.
- ◆ In a standard installation, cflowd.conf will be located in the /usr/local/arts/etc/ directory.
- ◆ cflowd.conf contains three types of stanzas:
  - an OPTIONS stanza specifying system-wide configuration values,
  - CISCOEXPORTER stanzas specifying configuration values for each Cisco from which we're collecting data, and
  - COLLECTOR stanzas specifying hosts from which we permit cfdcollect connections.

# options

◆ The OPTIONS stanza in cflowd.conf is used to set system-wide configuration values for cflowd, cflowdmux and local clients. There should be a single OPTIONS stanza in cflowd.conf, and it should be the first stanza. Following are descriptions of each of the settings in an OPTIONS stanza.

- ◆ LOGFACILITY *(optional)*
- ◆ TCPCOLLECTPORT *(optional)*
- ◆ PKTBUFSIZE *(optional)*
- ◆ TABLESOCKFILE
- ◆ FLOWDIR
- ◆ FLOWFILELEN *(optional)*
- ◆ NUMFLOWFILES *(optional)*
- ◆ MINLOGMISSED *(optional)*

# options example

```
OPTIONS {  
  LOGFACILITY:      mgmtserver  
  TCPCOLLECTPORT:   2056  
  TABLESOCKFILE:   usr/local/arts/etc/cflowdtable.socket  
  FLOWDIR:           /usr/local/arts/data/cflowd  
  FLOWFILELEN:       1000000  
  NUMFLOWFILES:     10  
  MINLOGMISSED:     300  
}
```

# CISCOEXPORTER

- ◆ The CISCOEXPORTER stanza is used to specify configuration values for a single Cisco router. There may be more than one CISCOEXPORTER stanza in `cflowd.conf`, with each corresponding to a Cisco router from which we would like to collect data.

# CISCOEXPORTER example

```
CISCOEXPORTER {  
  HOST: 204.212.46.1          # IP address of Cisco sending data.  
  ADDRESSES: { 204.212.46.1,   # Addresses of interfaces on Cisco  
               204.212.45.14 } # sending data.  
  CFDATAPORT: 2055           # Port on which to listen for data.  
  SNMPCOMM: 'public'  
  LOCALAS: 195               # Local AS of Cisco sending data.  
  COLLECT: { protocol, ifmatrix, portmatrix, netmatrix, nexthop, asmatrix,  
            tos, flows }  
}
```



# COLLECTOR

- ◆ The COLLECTOR stanza is used to hold configuration values for a host running cfdcollect. In a standard configuration, there will only be one or two of these, since a single cfdcollect host is normally used to collect data from all instances of cflowd. However, there may be more than one entry (for example, you may have a hot backup host on which you will run cfdcollect when the primary cfdcollect host is down).

# COLLECTOR Example

```
COLLECTOR {  
  HOST:      195.83.243.2    # IP address of host running cfdcollect  
  ADDRESSES: { 195.83.243.2, 195.83.241.9 } # other addresses of host  
  AUTH:      none  
}
```

# cfddcollect configuration

- ◆ cfddcollect uses a simple configuration file, typically named cfddcollect.conf and located in the /usr/local/arts/etc/ directory. cfddcollect.conf should contain two types of stanzas: a single 'system' stanza specifying system-wide values for cfddcollect, and one or more cflowd' stanzas (one for each instance of cflowd).

```
system {  
    logFacility:      mgmtserver          # Syslog to mgmtserver facility.  
    dataDirectory:    /usr/local/arts/data/cflowd  
    filePrefix:        arts  
    pidFile:          /usr/local/arts/etc/cfddcollect.pid  
}  
  
cflowd {  
    host:              mgmtserver.conference.apricot.net  
    tcpCollectPort:    2056  
    minPollInterval:   300  
}
```

# starting cflowdmux

- ◆ cflowdmux should be started before cflowd. It can be started with no arguments, in which case it will use a compiled-in default as the configuration file name (typically `/usr/local/arts/etc/cflowd.conf`). It will also accept an explicit configuration file name as the first argument. Examples:
  - ◆ `# cflowdmux` would start cflowdmux using the compiled-in default configuration file.
  - ◆ `# cflowdmux /etc/cflowd.conf` would start cflowdmux using `/etc/cflowd.conf` as the configuration file.

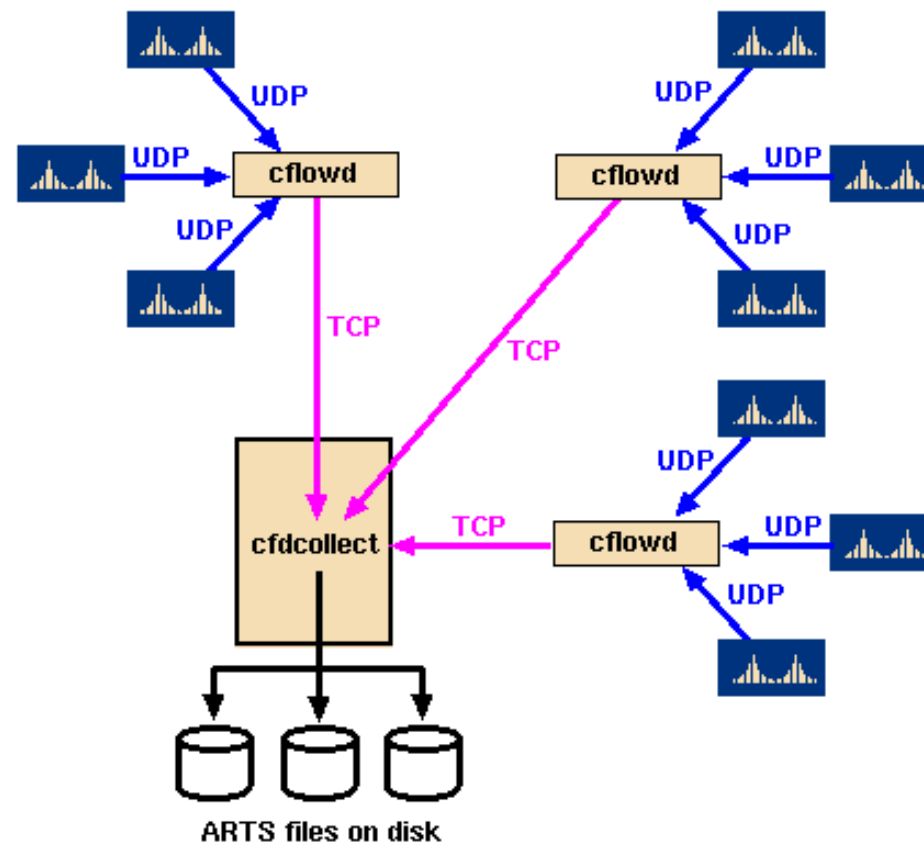
# starting cflowd

- ◆ After starting cflowdmux, you should start cflowd. Like cflowdmux, cflowd will use a compiled-in default configuration file (typically `/usr/local/arts/etc/cflowd.conf`) if given no arguments. It will also accept an explicit configuration file name as the first argument.

Examples:

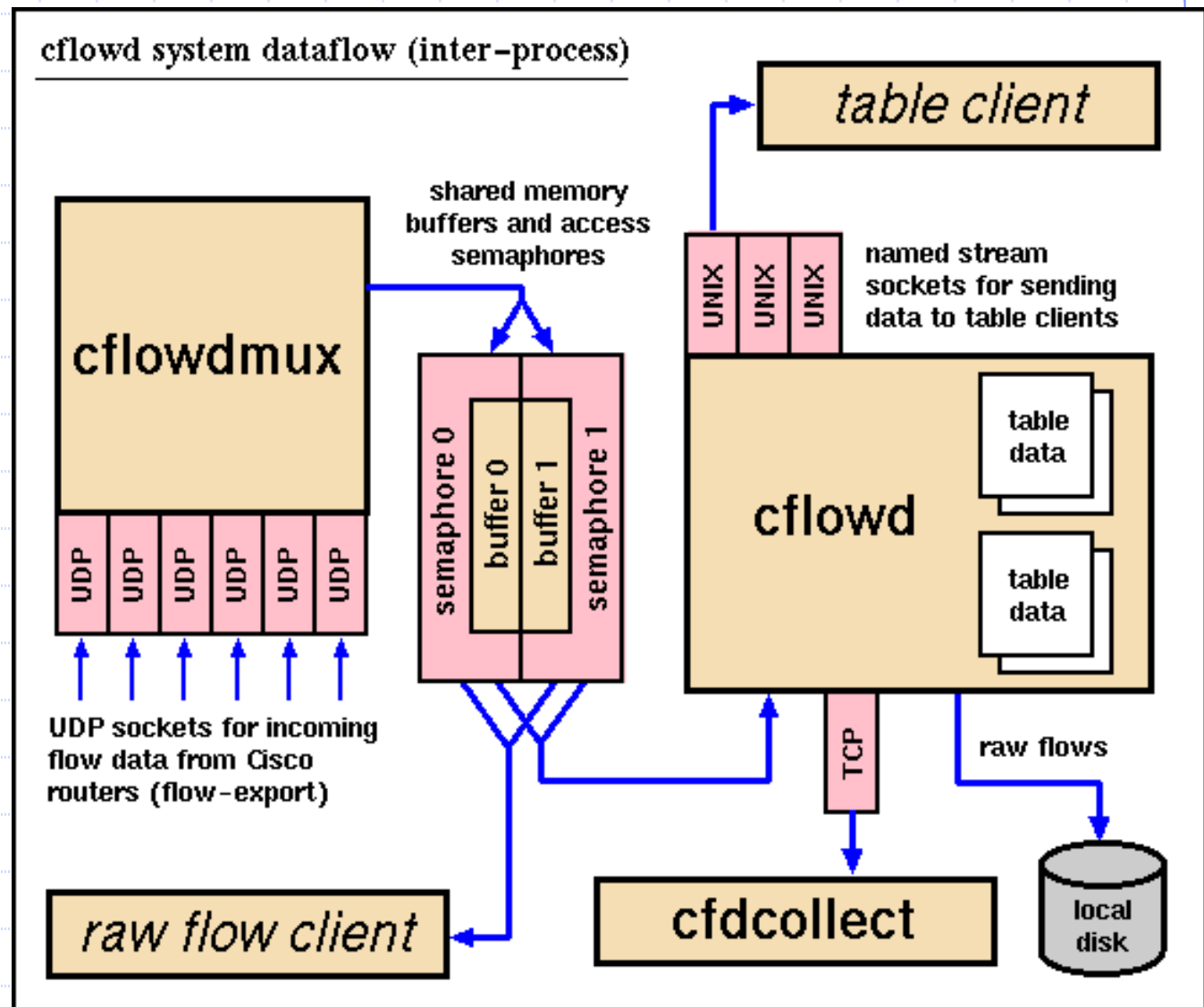
- ◆ `% cflowd` would start cflowd using the compiled-in default configuration file.
- ◆ `% cflowd /etc/cflowd.conf` would start cflowd using `/etc/cflowd.conf` as the configuration file.

# cflowd

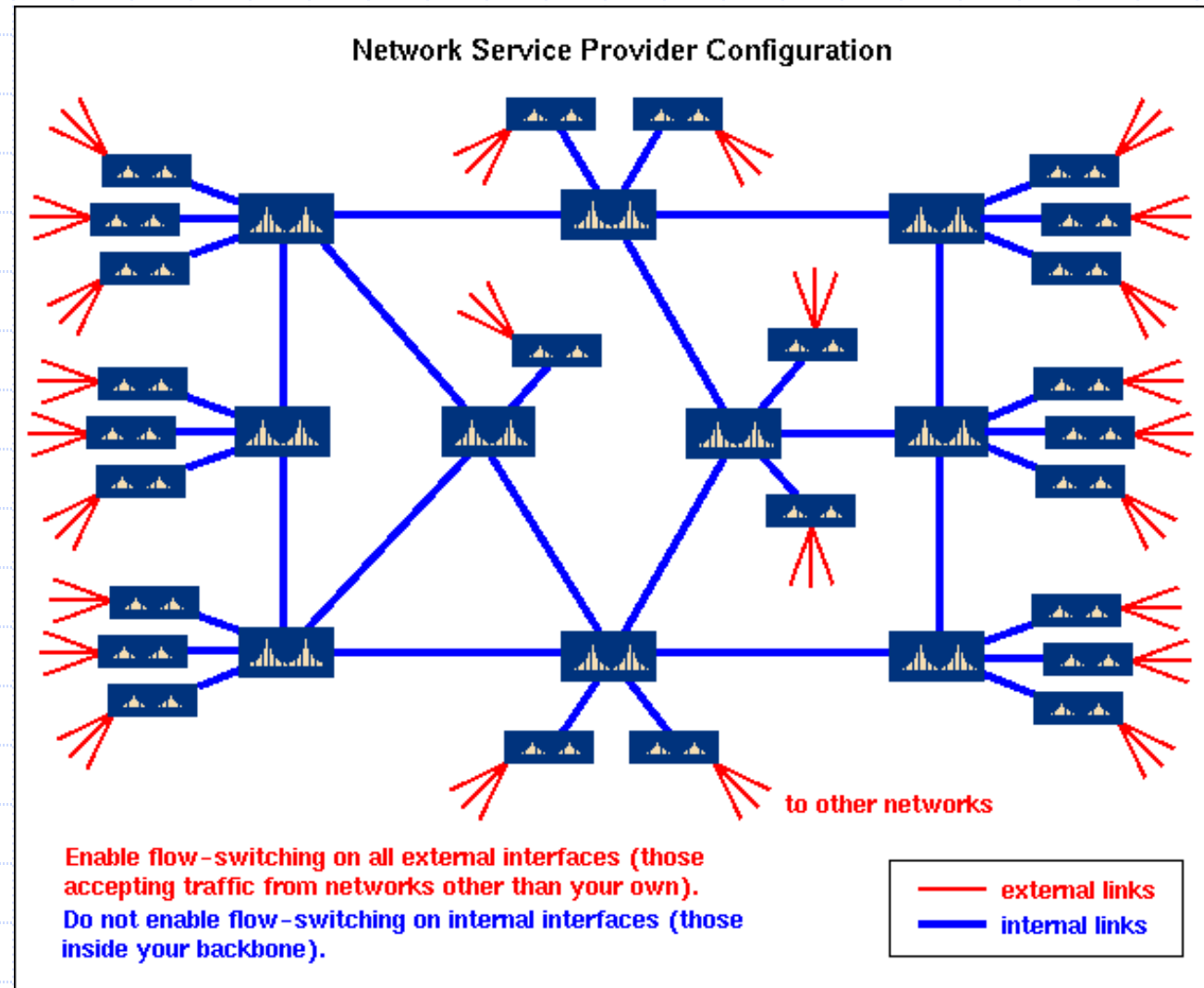


Data flow in the cflowd system. Each Cisco sends UDP flow-export packets to a host running cflowd. cflowd creates tabular summaries which are picked up by cfdcollect at regular intervals. cfdcollect stores the tabular data in ARTS files, which can be processed with the arts++ package.

# dataflow



# deployment





# Enabling Netflow in Cisco Routers

- ◆ These are the global configuration mode commands:
  - ip flow-export version 5 peer-as
  - ip flow-export source-interface xxx
- ◆ Choose the interface closest to your collector. This simply ensures that there is no confusion as to the source address that will be listed in the flows.
  - ip flow-export destination x.x.xx y
- ◆ x.x.xx is the collector's ip address, y is the port you will specify in the flow-capture command line. You may choose any port, just remember what it is and avoid the obvious registered ports like 80. (The flow packets are UDP.)
  - ip flow-cache timeout active 1
- ◆ This syntax is for IOS 12.2 and later. If you are running an 11.x or 12.0/12.1 code, the syntax would be: "ip flow-cache active-timeout 1". This command ensures the timely delivery of flows to the collector.
- ◆ In the interface configuration mode of **each major** interface: (major as opposed to sub-interface)
  - ip route-cache flow



```
export PATH=$PATH:/usr/local/rrdtool/bin
```



```
export PATH=$PATH:/usr/local/netflow/bin
```

# Flowscan

- ◆ FlowScan analyzes and reports on Internet Protocol (IP) flow data exported by routers. Consisting of Perl scripts and modules, FlowScan binds together
  - a flow collection engine (a patched version of [cflowd](#)),
  - a high performance database (Round Robin Database - RRD), and
  - a visualization tool ([RRDtool](#)). FlowScan produces graph images that provide a continuous, near real-time view of the network border traffic.