1 Introduction to ntop

Network management is becoming an increasingly complex task due to the variety of network types and the integration of different network media. As networks become larger, more complex, and more heterogeneous, the costs of network management rise. In this scenario, automated tools are needed to support human effort, gathering information about the status and behaviour of networked elements. According to [14], network monitoring is the most fundamental aspect of automated network management.

This text provides information on the use of `ntop` by network managers or operators. `ntop` [3] is a simple, free and portable traffic measurement and monitoring tool, initially conceived by Luca Deri and Stefano Suin for tackling performance problems on the campus network of the University of Pisa, Italy.

Similar to the Unix top tool that reports processes CPU usage, the authors needed a simple tool able to report the network top users (hence the term `ntop`) for quickly identifying those hosts that were currently using most of the available network resources. `ntop` then evolved into a more flexible and powerful tool [6], [8], [5]. The current version of `ntop` features command line and web-based user interfaces, and is available on both UNIX and Win32 platforms. It is currently developed using the concept of open source software [13]. `ntop` focuses on:

- traffic measurement,
- traffic monitoring,
- network optimization and planning, and
- detection of network security violations.

This text is structured as follows: Section 2 presents the features mentioned above in further detail, Section 3 describes the installation procedures, Section 4 provides an example of the use of `ntop`, and Section 5 discusses alternative approaches to monitoring.

2 Functions

This section presents in further detail `ntop`'s main functions: *traffic measurement*, *traffic monitoring*, *network optimization and planning*, and *detection of network security violations*.

2.1 Traffic Measurement

Traffic measurement consists in measuring the usage of relevant traffic activities. `ntop` tracks network usage, generating a series of statistics for each host in the local subnet and for the subnet as a whole. The needed information is collected by the host running `ntop` by simply observing the traffic on the network. This arrangement off-loads the processing requirements from operational nodes to the `ntop` host. All packets in the subnet are captured and associated with a sender/receiver pair. In this way, it is possible to track all traffic activities of a particular host.

The following table shows the information registered by `ntop` for each host connected to the (broadcast) network:

Data Sent / Received	The total traffic (volume and packets) generated or received by the host. Classified according to network protocol (IP, IPX, AppleTalk, etc.) and IP protocol (FTP, HTTP, NFS, etc.)
Used Bandwidth	Actual, average and peak bandwidth usage.
IP Multicast	Total amount of multicast traffic generated or received by the host.
TCP Sessions History	Currently active TCP sessions established/accepted by the host and associated traffic statistics.
UDP Traffic	Total amount of UDP traffic sorted by port.
TCP/UDP Used Services	For each protocol (e.g. HTTP), a list of the last 5 clients that interacted with the host using the protocol.
Traffic Distribution	Local traffic, local to remote traffic, remote to local traffic (a host is local if it belongs to either the specified network card subnet or to the subnet(s) specified in the initialization [5]).
IP Traffic Distribution	UDP vs. TCP traffic, relative distribution of the IP protocols according to the host name.

Table 1: Information recorded by `ntop` for each host

`ntop` also reports global traffic statistics, including:

Traffic Distribution	Local (subnet) traffic, local vs. remote (outside specified/local subnet), remote vs. local.
Packets Distribution	Total number of packets sorted by packet size, unicast vs. broadcast vs. multicast and IP vs. non-IP traffic.
Used Bandwidth	Actual, average and peak bandwidth usage.
Protocol Utilization and Distribution	Distribution of the observed traffic according to both protocol and source/destination (local vs. remote).
Local Subnet Traffic Matrix	Monitored traffic between each pair of hosts in the subnet.
Network Flows	Traffic statistics for user-defined flows (traffic of particular interest to the user)

Table 2: Global statistics recorded by ntop

In addition to the information provided above, the current version allows the installation of *plug-ins* to provide detailed statistics about particular protocols not present in the standard version. Examples of these are the NFS and NetBIOS plug-ins. `ntop` will also generate statistics about the host on which it is running, listing open sockets, data sent/received, and contacted peers for each process.

2.2 Traffic Monitoring

Traffic monitoring is the ability to identify those situations where network traffic does not comply with specified policies or when it exceeds some defined thresholds. In general, network administrators specify policies that apply to the behaviour of elements in the managed networked. Nevertheless, it is possible that some hosts will not comply with the policies prescribed. Typical causes of misbehaviour are related to misconfiguration of operating systems, network interfaces, software applications and others [6].

ntop provides support for detecting some network configuration problems including:

- Use of *duplicate IP addresses*.
- *Identification of local hosts in "promiscuous mode"*.
- *Misconfiguration of software applications*, by analysing protocol traffic data.
- *Service misuse detection*
  Identification of hosts that do not make use of specified proxies.
- *Protocol misuse*
  Identification of hosts that use unnecessary protocols.
- *Identification of subnet routers*
  Detection of misconfigured workstations acting as routers.
- *Excessive network bandwidth utilization*

2.3 Network Optimization and Planning

Sub-optimal configuration of hosts might influence negatively the overall performance of a network. ntop allows the administrator to identify potential sources of unproductive bandwidth usage, particularly the use of *unnecessary protocols* and *sub-optimal routing* problems. Indirectly, through *traffic characterization and distribution*, it is possible to revise policies for the network to promote *wiser bandwidth usage*.

2.4 Detection of Network Security Violations

In networks, most of the security attacks come from the network itself. For this reason ntop provides the users support for both tracking ongoing attacks and identifying potential security holes including *IP spoofing*, *network cards in promiscuous mode, denial of service attacks, trojan horses* (that use well known ports) and *portscan attacks*.

When a security violation or a network misconfiguration is identified, ntop offers facilities to *generate alarms for the network operator* (via e-mail, SNMP traps or Short Messaging Systems) and to *perform specific actions* (when applicable) in order to block the attack. As it is also possible to keep traffic information stored into a database, the records can be used to understand the attack and prevent further similar occurrences. Further information on the use of ntop for security purposes is available on [7].

It is important to note that ntop, as well as other monitoring tools, might pose security threats if not installed and configured properly. Free access to ntop's web interface will allow any user with web access to read all the information provided by ntop, gaining knowledge about the network that would not be disclosed otherwise.

3 Installation

ntop is currently available on version 1.3. It is distributed under the GNU General Public License [9], and can be down-loaded free of charge from ntop's official homepage [3] and other mirrors on the Internet. It supports the platforms, media and protocols shown in the table below.

Platforms	UNIX, Win32
Media	Ethernet, Token Ring, PPP, FDDI, Raw IP, Loopback
Protocols	IP, IPX, NetBIOS, OSI, AppleTalk, DecNet, DLC
IP Protocols	Fully user configurable (NFS, HTTP, X11, DNS, FTP, SMTP, POP, IMAP, SNMP, Telnet, etc.)

Table 3: Platforms, Media and Protocols supported by `ntop`

Before down-loading the software, it is important to select the station which will host `ntop`. This host should have an interface to the network to be monitored, since only the traffic captured through this interface can be analysed. In switched networks (or bridged networks), when selecting `ntop`'s host station, it is important to consider that only the segment where `ntop`'s host is installed will be monitored. Nevertheless, modern switches (switching hubs) allow global network traffic (or virtual LANs) to be mirrored to a specified switch port. Therefore, `ntop` can be activated on a host that is attached to such a port. Unfortunately, this is not possible in case of different LANs interconnected via routers, for instance, in an IP inter-network.

After having selected which station will host `ntop`, a proper down-load format should be chosen. Available formats include:

- *source code* (which should compile virtually on any UNIX and Win32 platform)
- *application binary or binary package for different UNIX flavours* (Linux, IRIX 6.2, Solaris 2.7 i386/SPARC, HP-UX 11.X, FreeBSD 3.X, AIX 4.1), and
- *binary demo for Windows 95/98/NT* (limited to 1,000 packets capture).

Both UNIX and Win32 versions are developed under a single source-code tree, using a system-independent interface for user-level packet capture called `libpcap`. This library is available for most UNIX flavours, and has been ported to the Win32 platform by the authors of `ntop`. This Win32 port can also be down-loaded from the official homepage.

In the supported UNIX platforms, after having down-loaded `ntop`'s source code and installed `libcap`, `ntop` should be compiled and installed:

```
# cd /ntops-directory/ntop-1.3
# sh ./configure
# make
# make install
# exit
```

If `ntop` has been down-loaded in binary format, the installation process depends on the package manager being used.

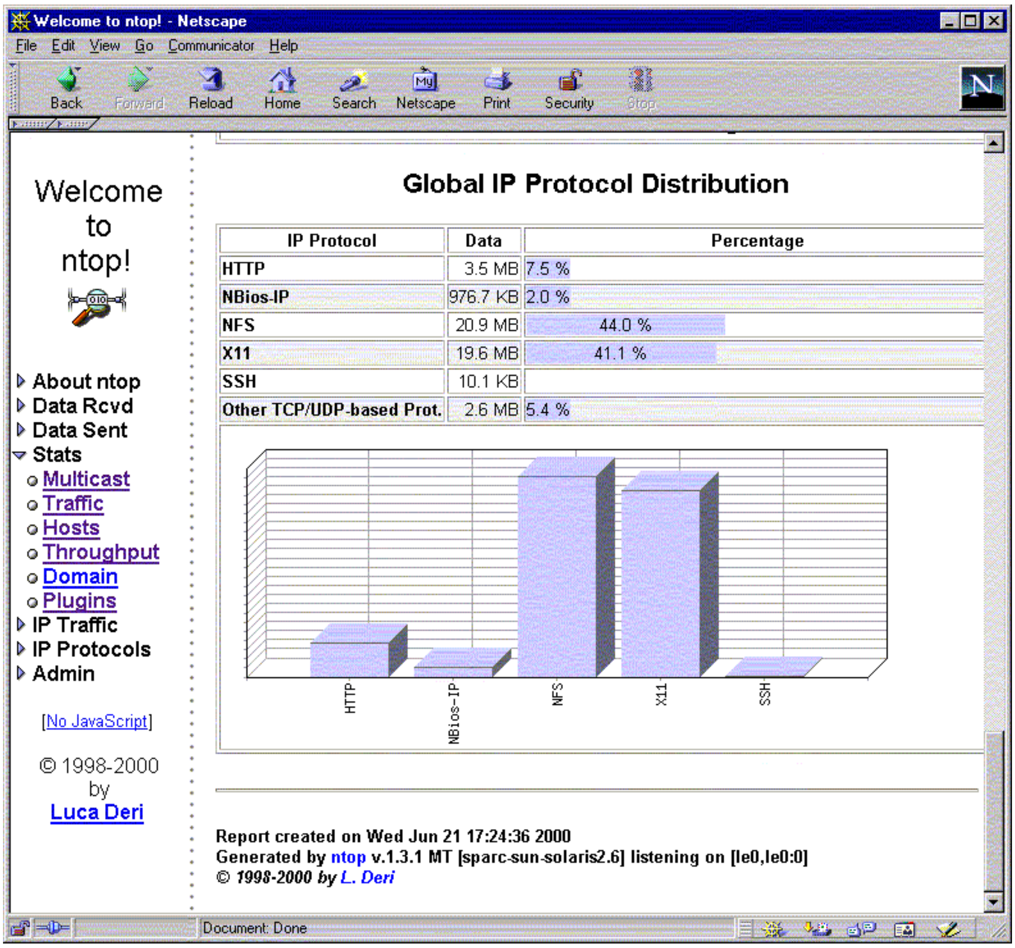
As mentioned before, `ntop`'s full source code is available free of charge on the Internet, through the URL ftp://ftp.ntop.org/pub/local/ntop/snapshots/. In order to fund the project partially, if a Windows user is not willing or is not able to compile `ntop`, the full version in binary form is distributed under the payment of a US$ 49.95 fee.

After the installation, `ntop` should be executed (by a user with super-user access), and will start capturing packets from the network. When activated in web-based mode, `ntop` features its own internal web server (set to a specific port on the start-up). Therefore, it will be possible to access the software with a web browser through the URL `http://hostname:portnumber/`

The current version of `ntop` supports *plug-ins*, as an extension mechanism. The administrator is allowed to extend `ntop`'s functionality with extra-features. Examples of plug-ins are ICMP, ARP/RARP and WAP plug-ins. Those can be installed optionally, and started-up selectively during `ntop` initialization.

4 Utilization Examples

In this section, some examples of capabilities are presented. The examples show screenshots of `ntop`'s web-based mode.



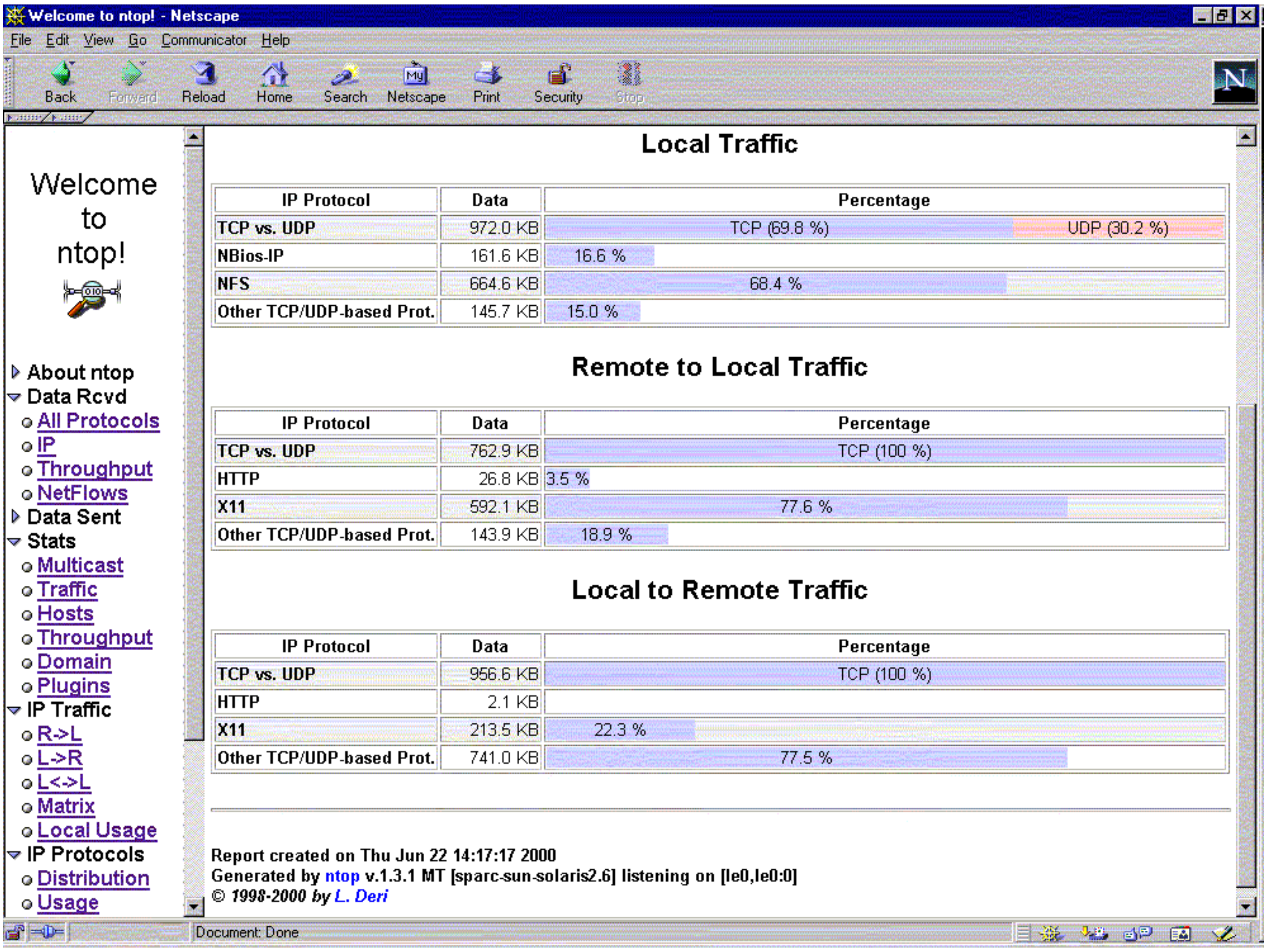
Global IP Protocol Distribution

IP Protocol	Data	Percentage
HTTP	3.5 MB	7.5 %
NBios-IP	976.7 KB	2.0 %
NFS	20.9 MB	44.0 %
X11	19.6 MB	41.1 %
SSH	10.1 KB	
Other TCP/UDP-based Prot.	2.6 MB	5.4 %

Report created on Wed Jun 21 17:24:36 2000
Generated by ntop v.1.3.1 MT [sparc-sun-solaris2.6] listening on [le0,le0:0]
© 1998-2000 by L. Deri

Figure 1: Global IP Protocol Distribution

The traffic statistics report general information about the observed traffic. The traffic is considered from a global perspective, with no host-specific information. In Figure 1, it is possible to view the *Global IP Protocol Distribution* table and graph. The data collected by `ntop` shows that NFS and X11 are the highest bandwidth consuming protocols currently present in the network. Together they account for 85.1% of the network usage. This sort of statistics is important for the administrator to understand the traffic, associating it to specific applications. In this way, it will be possible to manage the available bandwidth appropriately.



Local Traffic

IP Protocol	Data	Percentage
TCP vs. UDP	972.0 KB	TCP (69.8 %) UDP (30.2 %)
NBios-IP	161.6 KB	16.6 %
NFS	664.6 KB	68.4 %
Other TCP/UDP-based Prot.	145.7 KB	15.0 %

Remote to Local Traffic

IP Protocol	Data	Percentage
TCP vs. UDP	762.9 KB	TCP (100 %)
HTTP	26.8 KB	3.5 %
X11	592.1 KB	77.6 %
Other TCP/UDP-based Prot.	143.9 KB	18.9 %

Local to Remote Traffic

IP Protocol	Data	Percentage
TCP vs. UDP	956.6 KB	TCP (100 %)
HTTP	2.1 KB	
X11	213.5 KB	22.3 %
Other TCP/UDP-based Prot.	741.0 KB	77.5 %

Report created on Thu Jun 22 14:17:17 2000
Generated by ntop v.1.3.1 MT [sparc-sun-solaris2.6] listening on [le0,le0:0]
© *1998-2000 by L. Deri*

Figure 2: Local, Remote to Local and Local to Remote Traffic

The tables in Figure 2 show statistics on *local*, *remote to local and local to remote traffic*. A host is considered local if attached to the local broadcast network, or remote otherwise [5]. The *local traffic table* shows information on exchanged traffic between local hosts. In the example, it is possible to verify that NFS accounts for 68.4% of the local traffic. The *remote to local traffic table* shows the incoming traffic generated from remote (non-local) hosts. In this example, local X11 servers are being used by hosts outside the network segment. With access to this sort of information, the administrator is able to revise policies on acceptable remote X-Windows usage. As could be expected, the *local to remote traffic table* relates to the traffic leaving the local network boundaries.

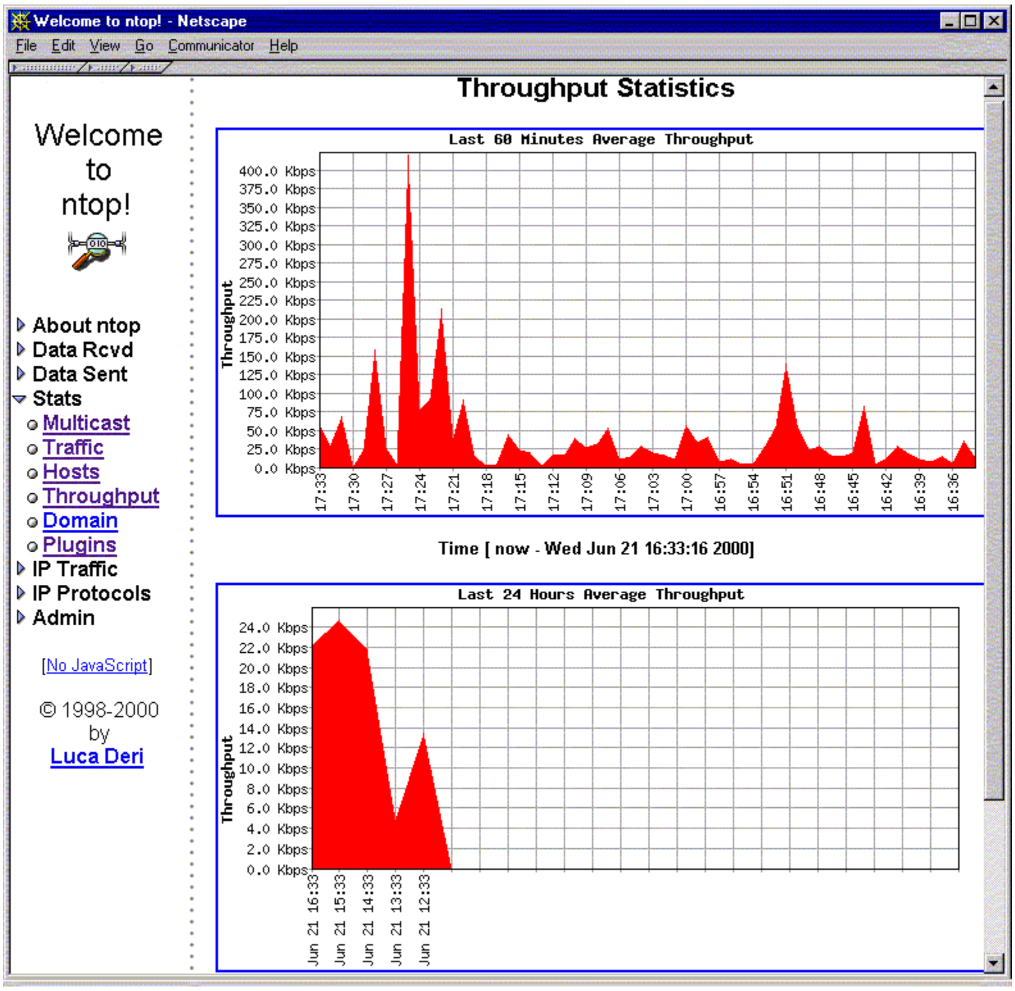
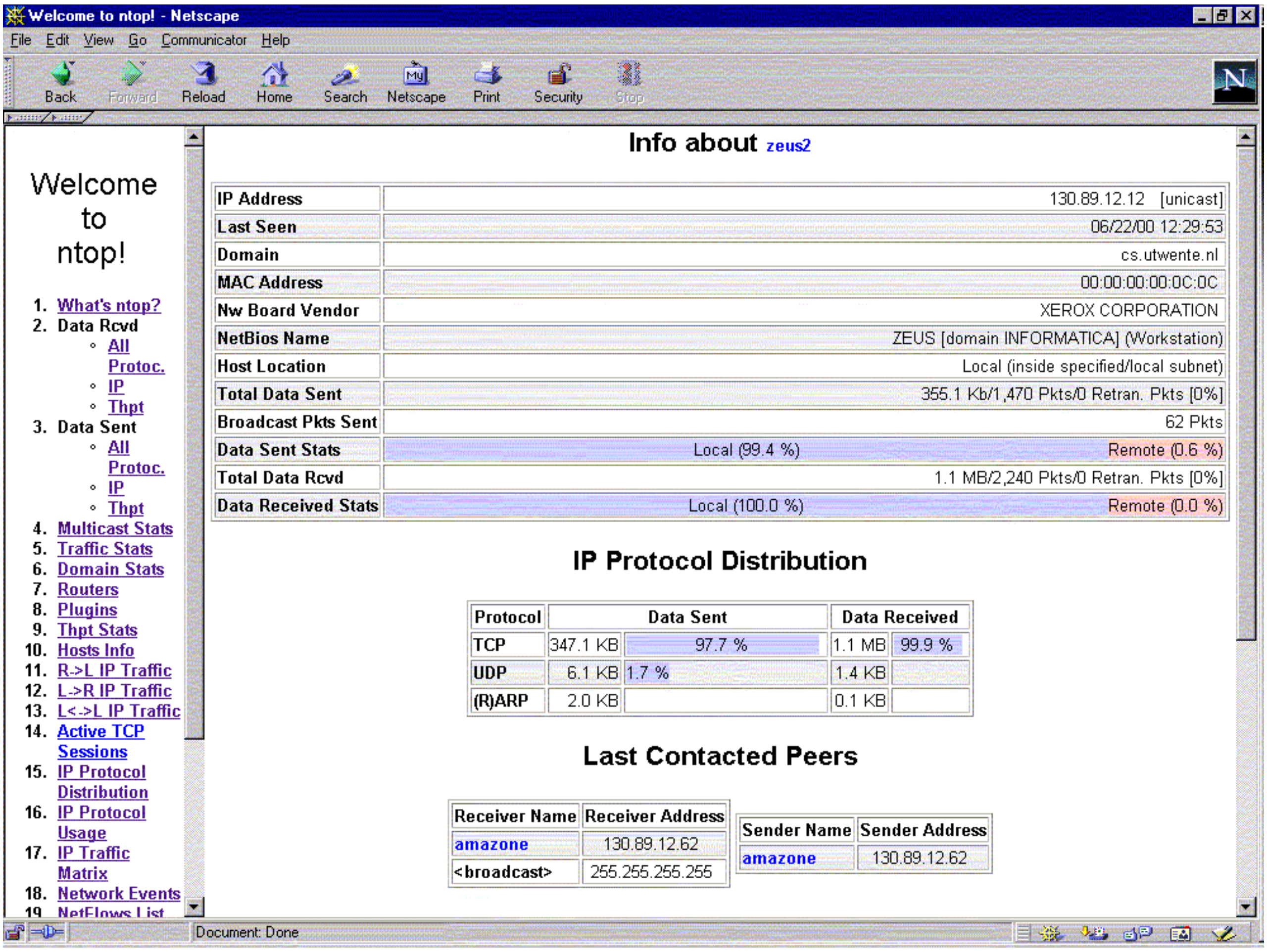


Figure 3: Throughput Statistics

Figure 3 shows another global traffic statistics, displayed in *throughput graphs*. Those are graphs that show the evolution of the total throughput observed in the network. They are presented in different time scales, showing the throughput in the last 60 minutes and in the last 24 hours. This sort of statistics is valuable to determine peek and low usage periods. In this way the administrator will be able to better schedule traffic intensive or network disruptive activities (physical network maintenance, switch configuration, data traffic with low priority, etc.). It might also be interesting to detect unexpected throughput peeks, which could indicate excessive use of the network resources by a user or group of users, or other non-standard behaviour.

The previous examples showed the use of `ntop` for global traffic information. Figure 4 shows some information provided by `ntop` for a *specific host*.


Welcome to ntop! - Netscape
File Edit View Go Communicator Help
Back Forward Reload Home Search Netscape Print Security Stop
Document: Done

Welcome to ntop!

1. What's ntop?
2. Data Rcvd
   - All Protoc.
   - IP
   - Thpt
3. Data Sent
   - All Protoc.
   - IP
   - Thpt
4. Multicast Stats
5. Traffic Stats
6. Domain Stats
7. Routers
8. Plugins
9. Thpt Stats
10. Hosts Info
11. R->L IP Traffic
12. L->R IP Traffic
13. L<->L IP Traffic
14. Active TCP Sessions
15. IP Protocol Distribution
16. IP Protocol Usage
17. IP Traffic Matrix
18. Network Events
19. NetFlows List

Info about zeus2

IP Address		130.89.12.12 [unicast]
Last Seen		06/22/00 12:29:53
Domain		cs.utwente.nl
MAC Address		00:00:00:00:0C:0C
Nw Board Vendor		XEROX CORPORATION
NetBios Name		ZEUS [domain INFORMATICA] (Workstation)
Host Location		Local (inside specified/local subnet)
Total Data Sent		355.1 Kb/1,470 Pkts/0 Retran. Pkts [0%]
Broadcast Pkts Sent		62 Pkts
Data Sent Stats	Local (99.4 %)	Remote (0.6 %)
Total Data Rcvd		1.1 MB/2,240 Pkts/0 Retran. Pkts [0%]
Data Received Stats	Local (100.0 %)	Remote (0.0 %)

IP Protocol Distribution

Protocol	Data Sent		Data Received	
TCP	347.1 KB	97.7 %	1.1 MB	99.9 %
UDP	6.1 KB	1.7 %	1.4 KB	
(R)ARP	2.0 KB		0.1 KB	

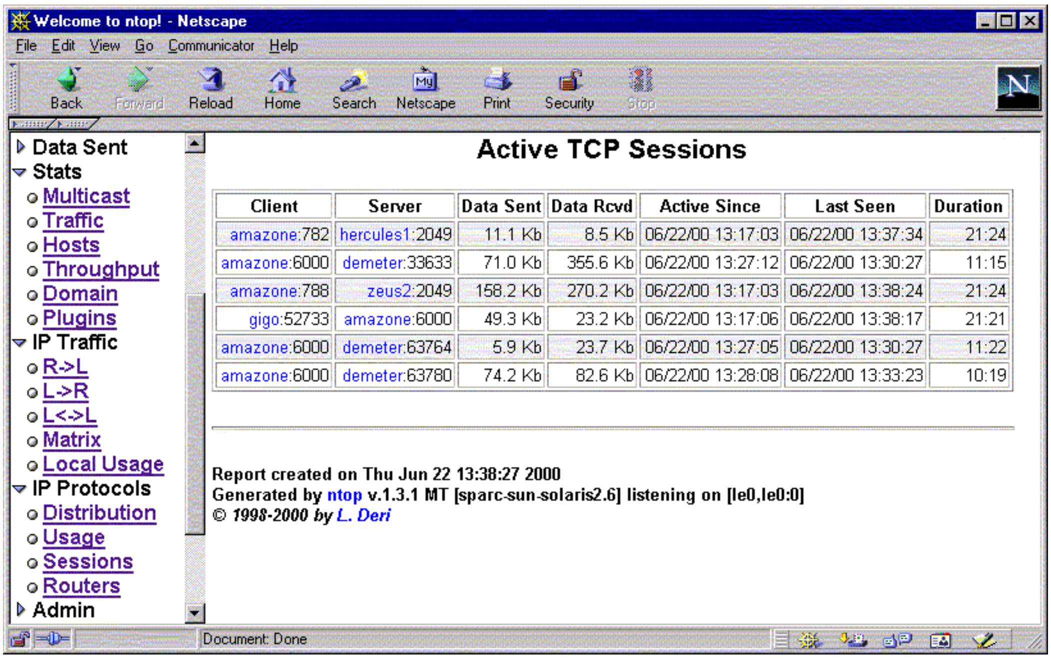
Last Contacted Peers

Receiver Name	Receiver Address
amazone	130.89.12.62
<broadcast>	255.255.255.255

Sender Name	Sender Address
amazone	130.89.12.62

Figure 4: Host Information

The listing includes IP address, MAC address and board vendor (only for local hosts), total data sent/received statistics (local vs. remote traffic), broadcast packets sent, etc. The *IP protocol distribution table* provides information about protocol distribution, dividing IP traffic according to known higher-level protocols. The *last contact peers table* shows the last hosts that exchanged data with the host being inspected. Host information will support network operators in the proper configuration and maintenance of individual elements in the network. Moreover, hosts might be associated with specific users. These statistics could be representative of their behaviour.



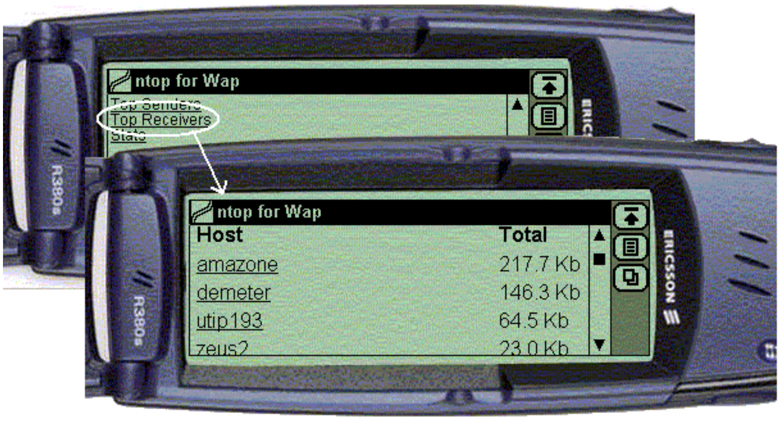
Active TCP Sessions

Client	Server	Data Sent	Data Rcvd	Active Since	Last Seen	Duration
amazone:782	hercules1:2049	11.1 Kb	8.5 Kb	06/22/00 13:17:03	06/22/00 13:37:34	21:24
amazone:6000	demeter:33633	71.0 Kb	355.6 Kb	06/22/00 13:27:12	06/22/00 13:30:27	11:15
amazone:788	zeus2:2049	158.2 Kb	270.2 Kb	06/22/00 13:17:03	06/22/00 13:38:24	21:24
gigo:52733	amazone:6000	49.3 Kb	23.2 Kb	06/22/00 13:17:06	06/22/00 13:38:17	21:21
amazone:6000	demeter:63764	5.9 Kb	23.7 Kb	06/22/00 13:27:05	06/22/00 13:30:27	11:22
amazone:6000	demeter:63780	74.2 Kb	82.6 Kb	06/22/00 13:28:08	06/22/00 13:33:23	10:19

Report created on Thu Jun 22 13:38:27 2000
Generated by ntop v.1.3.1 MT [sparc-sun-solaris2.6] listening on [le0,le0:0]
© 1998-2000 by L. Deri

Figure 5: Active TCP Sessions

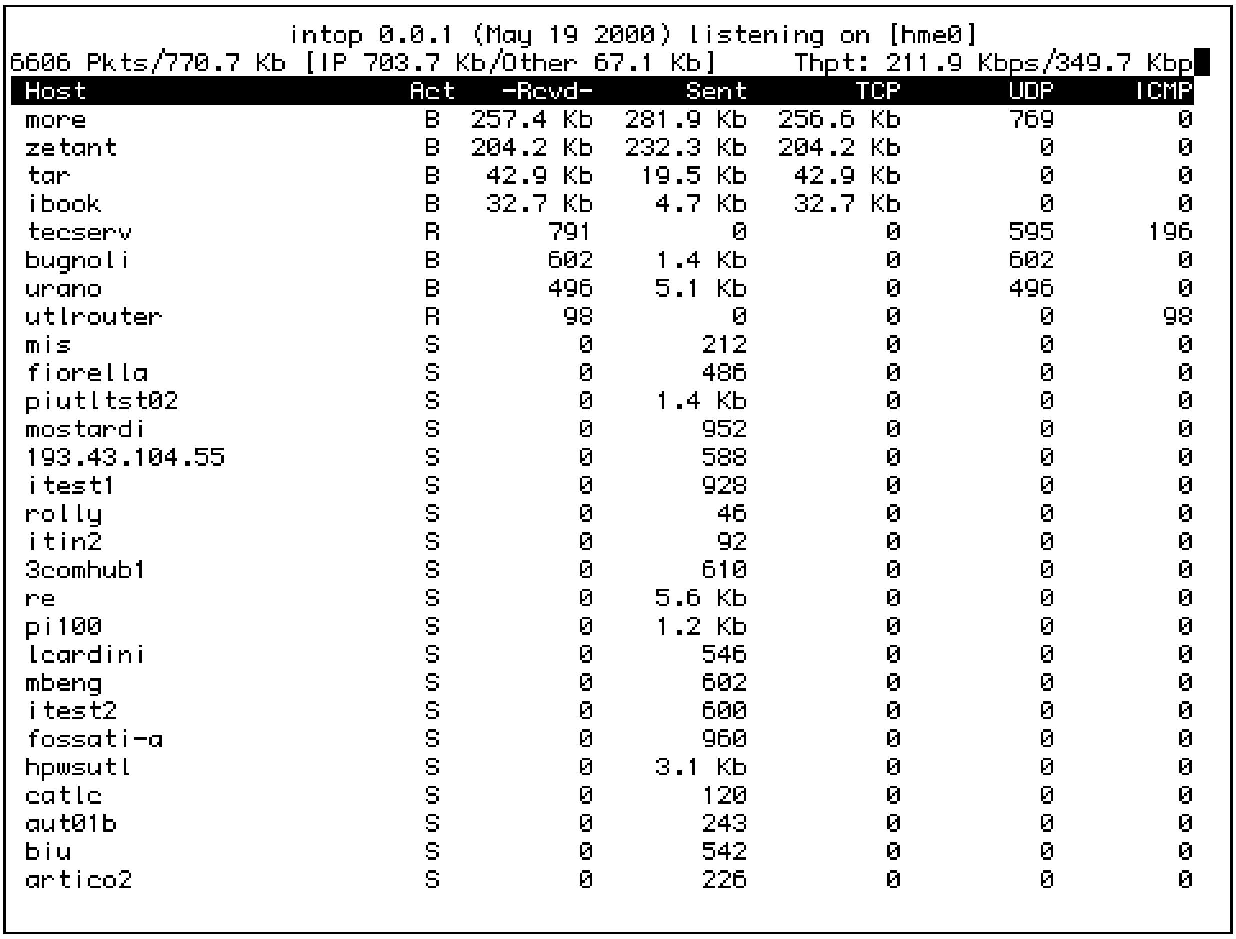
ntop is able to analyse individual captured IP packets and relate them to *active TCP sessions*. This is possible because ntop implements the TCP protocol machine [5]. In Figure 5 the *active TCP sessions table* is showed, with an entry for each active connection. In this way it is possible to recognize specific flows and the traffic associated to them. The listing includes for each entry: calling- and called- host addresses, data sent and received, connection time, and session duration.



Host	Total
amazone	217.7 Kb
demeter	146.3 Kb
utip193	64.5 Kb
zeus2	23.0 Kb

Figure 6: ntop's WAP plug-in - accessing 'top receivers' from a WAP device

As mentioned in Section Installation, ntop's functionality can be extended using plug-ins. Figure 6 depicts ntop being accessed via a WAP device (for the example, an emulator of a WAP device [10] was used). This is possible through the installation and activation of a WAP plug-in [4], which is responsible for the generation of final statistics in WAP format.



```
                    intop 0.0.1 (May 19 2000) listening on [hme0]
6606 Pkts/770.7 Kb [IP 703.7 Kb/Other 67.1 Kb]     Thpt: 211.9 Kbps/349.7 Kbp
 Host                     Act    -Rcvd-      Sent      TCP        UDP     ICMP
 more                      B   257.4 Kb  281.9 Kb 256.6 Kb        769        0
 zetant                    B   204.2 Kb  232.3 Kb 204.2 Kb          0        0
 tar                       B    42.9 Kb   19.5 Kb  42.9 Kb          0        0
 ibook                     B    32.7 Kb    4.7 Kb  32.7 Kb          0        0
 tecserv                   R        791         0        0        595      196
 bugnoli                   B        602    1.4 Kb        0        602        0
 urano                     B        496    5.1 Kb        0        496        0
 utlrouter                 R         98         0        0          0       98
 mis                       S          0       212        0          0        0
 fiorella                  S          0       486        0          0        0
 piutltst02                S          0    1.4 Kb        0          0        0
 mostardi                  S          0       952        0          0        0
 193.43.104.55             S          0       588        0          0        0
 itest1                    S          0       928        0          0        0
 rolly                     S          0        46        0          0        0
 itin2                     S          0        92        0          0        0
 3comhub1                  S          0       610        0          0        0
 re                        S          0    5.6 Kb        0          0        0
 pi100                     S          0    1.2 Kb        0          0        0
 lcardini                  S          0       546        0          0        0
 mbeng                     S          0       602        0          0        0
 itest2                    S          0       600        0          0        0
 fossati-a                 S          0       960        0          0        0
 hpwsutl                   S          0    3.1 Kb        0          0        0
 catlc                     S          0       120        0          0        0
 aut01b                    S          0       243        0          0        0
 biu                       S          0       542        0          0        0
 artico2                   S          0       226        0          0        0
```

Figure 7: `intop` *-* `ntop` *in interactive text mode - Source [5]*

Figure 7 shows `ntop` in its interactive mode, also known as `intop`. It is a shell to `ntop` and presents data in textual format, organized in tables. In this example it is possible to view the list of hosts that have sent/received data. The other columns highlight host activity, considering in particular sent and received data, TCP, UDP and ICMP data. A thorough (though currently outdated) description if `ntop`'s user interface can be found on `ntop`'s User Guide [5].

5 Alternative Approaches to Monitoring

Simple alternatives to network monitoring are packet tracers and decoders, often-called *network sniffers*. Examples are `tcpdump` [11] and `snoop` [15]. These tools are responsible for capturing packets from the network and often require off-line analysis tools to correlate captured data and identify network flows. Sniffers usually provide details on packet activity and lack information on the network as a whole [8]. Protocol analysers, such as Ethereal [2], typically focus on the content of single network packets and not on global network activities. These solutions lack high-level support to management activities.

More appropriate and advanced alternatives include RMON (Remote Network Monitoring) management platforms [16]. Those platforms promote a decoupling between *probes* and *managers*. *Probes* are devices that collect data from the network and *managers* are applications that provide useful higher level information for the human operator. RMON managers can be seen as data analysers, but will also configure probes and retrieve relevant collected data (via SNMP). In RMON, flexibility was achieved through modularity and standardization. RMON's MIB (Management Information Base) and architecture have been defined in RFCs [16].

The RMON standards define the way in which a manager can retrieve information from probes and which pieces of information are available. In an RMON configuration, a manager can collect data from several probes. Therefore, it is possible to monitor several subnets from one central manager. This possibility is also available for the latest version of ntop. Despite the fact that ntop runs as a single application, including the probe and the analyser, it provides an API for remote programs to read (and in a future version modify) traffic information. This API is called the *remote interface*. The remote interface has to be ported to client-side platforms that would like to interact with ntop "probes" (it is currently available in a limited number of programming languages and platforms). This restricts the usage of ntop's remote interface to the platforms supported. The RMON standards, on the other hand, specify a *communication protocol* between manager and probes. This standardization opens the path to multi-vendor management solutions. RMON-ready managers can interact with probes implemented in any platforms and RMON-ready probes can interact with managers implemented in any platforms.

With ntop, specialized functions are typically implemented via new versions and optional plug-ins. As RMON is defined in a standard, it is possible to create specialized managers for different purposes (for instance, a security failure detection manager) or to use different applications for data analysis *that will inter-operate with probes from different vendors*.

RMON-base solutions are quite powerful but unfortunately need sophisticated SNMP managers that are able to configure the probes properly, and analyse collected network statistics. Due to the complexity and costs of RMON solutions, those are basically used by advanced network managers in large institutions.

Tools for network monitoring such as NeTraMet [1] and NFR [12] offer advanced programming languages for analysing network flows and building statistical event records. These languages are useful for experienced network operators. They have been designed as instrumentable network daemons more suitable for monitoring networks in a mid/long time period [8].

ntop has shown to be a valuable tool for quick access to network monitoring, with a simple to use integrated web interface, minimal requirements and lightweight CPU utilization. It is available for network administrator with minimal (installing, learning) effort and cost, as opposed to expensive and complex (yet sophisticated and flexible) management platforms.

6 References

[1] Brownlee N. NeTraMet 4.2 Users' Guide, Information Technology Systems & Services, The University of Auckland, New Zealand, August 1998. Available at http://www.auckland.ac.nz/net/Accounting/usguide.pdf

[2] Combs, G. et al. The Ethereal Network Analyzer, available at: http://ethereal.zing.org/

[3] Deri, L., Suin, S. and Carbone, R. Ntop - Network Top, available at: http://www.ntop.org/

[4] Deri, L. Beyond the Web: Mobile WAP-based Management. Centro Serra, University of Pisa, Italy. Available at http://jake.unipi.it/~deri/WAP.pdf.gz

[5] Deri, L. NTOP User's Guide - Network Usage Monitor for Unix Systems. Centro Serra, University of Pisa, Italy. Available at ftp://ftp.unipi.it/pub/local/ntop/snapshots/NTOP.pdf.gz

[6] Deri, L. and Suin, S. Effective Traffic Measurement using ntop. IEEE Communications Magazine, 38(5), pp 138-145, May 2000

[7] Deri, L. and Suin, S. Improving Network Security Using Ntop. In Proceedings of the RAID 2000 - Workshop on the Recent Advances in Intrusion Detection, Toulouse, France, October 2000.

[8] Deri, L. and Suin, S. Ntop: beyond Ping and Traceroute. In Proceedings of the DSOM'99, Zürich, Switzerland, October 1999.
[9] Free Software Foundation. GNU General Public License. Available at http://www.gnu.org/copyleft/gpl.html
[10] gelon.net. WAP browser @ Gelon.net. Available at http://www.gelon.net/
[11] Jacobson, V., Leres, C., and McCanne, S. tcpdump, Lawrence Berkeley National Labs, Available at ftp://ftp.ee.lbl.gov/
[12] Network Flight Recorder, Inc. Network Flight Recorder. Available at http://www.nfr.net
[13] opensource.org. The Open Source Page. Available at http://www.opensource.org/
[14] Stallings, W. SNMP, SNMPv2, SNMPv2 and RMON 1 and 2, Third Edition, Addison Wesley, Sept. 1999.
[15] Sun Microsystems, Inc. Snoop UNIX man pages (SunOS 5.6).
[16] Waldbusser, S., Remote Network Monitoring Management Information Base, IETF STD 0059, May 2000.