The dependability of an IP network – what is it?

A baseline paper of the IPLU project

May 16th, 2006

The IPLU team:

Kimmo Ahola, Eija Myötyri, Ilkka Norros (coordinator), Leena Norros,
Urho Pulkkinen, Pertti Raatikainen, Tapio Suihko

Contents:

1 Introduction

It is generally recognized that telecommunication networks are one of the most vital infrastructures. A mega-trend in telecommunications is that all services, traditional ones as well as new ones, move to the Internet, that is, become to be realized as services offered by means of one generic service, the global delivery system of Internet Protocol (IP) packets.

The Internet can be considered as a new, generic medium, even in the strict sense defined by Marshall McLuhan [5]. It can and probably will swallow to a large extent the older electronic media like telephone and television, which become to just particular services offered by means of the underlying universal packet transfer infrastructure. This process is, however, not smooth but poses serious problems whose final solutions are not clear. For example, economy drives towards All-IP solutions offering more or less free telephony, but a dramatic fall of the profitability of legacy telephone networks may have undesired consequences.

Since a large part of existing services and functions are moving to the Internet, so that extensive loss of IP connectivity would in fact paralyse the society, it is more than natural to ask questions about the *dependability* of IP networks, i.e., questions about their availability, reliability, controllability, vulnerability, security, etc. Can one rely on this new infrastructure as much as on the traditional technologies and the old ways of acting? High level of dependability is an evident requirement for a global information infrastructure, but this has in fact not been the trademark of the Internet. Rather, although the dominant experience is the impressive success of the Internet, we are accustomed to many kinds of negative surprises like the effects of virus attacks.

The Internet was not designed for a global infrastructure, it has grown to one, like an organism. IP networks resemble living organisms also in being under constant attack and extra load by malicious and detrimental elements, and surviving and getting stronger through learning in this fight. The Internet has developed to a publicly controlled infrastructure, but, again like living organisms, it carries some problems and vulnerabilities of its early design.

It is good to keep in mind that we live in a rapid growth phase of telecommunications that makes it difficult to speak *sub specie eternitatis:* for example, what amount of communication will be found “sufficient” and make the growth slow down, or will any? In any case, right now we are about to make an irreversible shift to the All-IP paradigm, whose dependability aspects are not completely understood nor taken care of yet.

1.1 The IPLU project

The IPLU project is a Finnish publicly funded research project. The acronym comes from the project’s name “IP-verkkojen LUotettavuuden arviointimenetelmät” (“Methods for assessing the dependability of IP networks”). IPLU aims at a holistic view and understanding of what the crucial aspects of Internet's dependability are, and how the research and practical measures aiming at improved dependability should be focused. The project maintains the web page http://iplu.vtt.fi.

1.2 The aim and structure of this document

This paper is a "baseline paper" of the IPLU project that is partly an interim report but more a worked-out starting point for its remaining part and future research. It also serves as background material for participants of the Finnish seminar and international workshop that the project organizes on May 17 and May 18-19, respectively, 2006.

The body of this document consists of three sections. Section 2 reviews the current architecture of IP networks, with special attention to vulnerabilities and other dependability aspects. Section 3 proposes a conceptual framework for discussing the dependability of IP networks and the Internet. A set of standard aspects of dependability is augmented with certain system-theoretic notions. Section 4 discusses the problems of measuring dependability.

2 On the architecture of IP networks and IP-based services

2.1 A basic architecture

The Internet is a network that interconnects different physical networks that implement the Internet Protocol (IP) on the network layer. The connected networks usually operate independent of the others and are called autonomous systems (AS). Each AS connects to one or several other ASs as illustrated in Figure 1.

An AS is a set of routers having a single policy and running under a single administration. Each AS has an identifying number, assigned by an Internet Registry or Internet Service Provider (ISP). Routing information between ASs is exchanged via an Exterior Gateway Protocol (EGP), such as BGP4. An AS may be a stub AS (single attachment point to outside), multi-homed non-transit AS (multiple attachments but no transit traffic between other ASs), or multi-homed transit AS. In February 4th 2006, there were 21423 ASs in the Internet and 18625 of them were stubs.

The heart of the Internet is the Internet Protocol (IP) that defines the format and semantics of the Internet packets. Doyle *et al.* [3] compare the Internet protocol stack nicely with Lego toys, where Lego's "snap protocol" corresponds to the role of IP as the thin waist of the hourglass-like architecture.

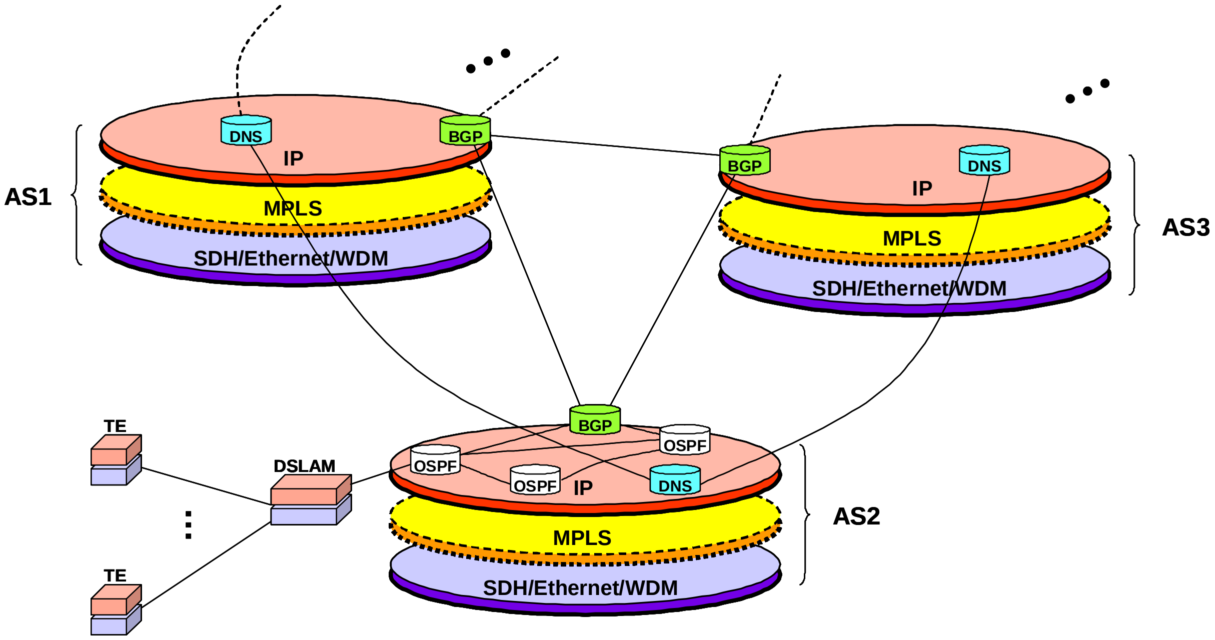
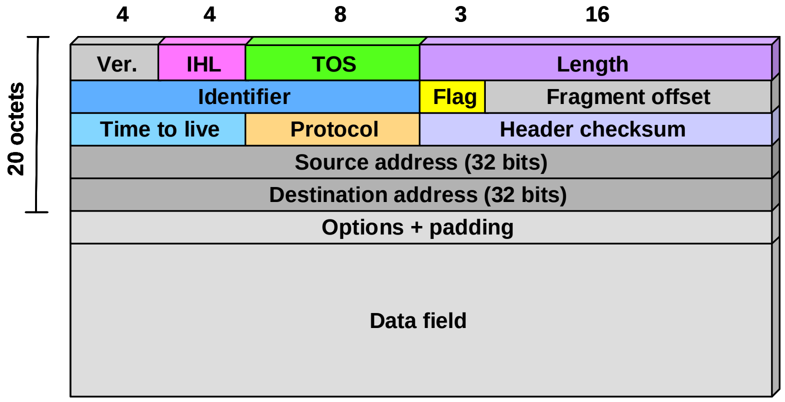


Figure 1. Main elements of the Internet architecture.

An IP packet is a datagram (see Figure 2), which is processed in the network as an independent entity without any reference to higher level notions like connections. The most important information in the packet header is the destination address. The main principle is that a packet coming to a router is forwarded to some of the router's output ports according to the packet's destination address. The routers maintain routing tables that include routing information in the form of prefixes. Each prefix represents a group of IP addresses and the routers job in making a routing decision is to find the longest prefix that matches the address. The found prefix finally is the key to determine the desired output port.

Other important IP packet fields are those dedicated for protocol, service class, time-to-live and source address. The protocol field indicates the type of the transport layer protocol, e.g., TCP, UDP or OSPF, carried by the IP packet. The service class field enables to differentiate the service quality through the handling of packets, for example to give priority to delay-constrained traffic. Techniques for traffic differentiation have been developed under the title Differentiated Services (DiffServ). Service differentiation is complicated in practice and has not yet been very widely applied, although modern routers can support it by sophisticated queueing systems

The time-to-live field is used for preventing unrestricted circulation of IP packets. It is decremented by each router the packet traverses, and a packet with zero time-to-live is deleted. The source address field identifies the sending end of the IP packet. The payload of an IP packet can be a full-fledged IP packet, possibly in encrypted form. This technique, called tunnelling, has many important uses, e.g., it allows build closed and encrypted virtual networks over a public IP network.



Version - IP version number
IHL - Internet header length in 32-bit words (min. size = 5)
TOS - type of service (guidance to end-system IP modules and routers along transport path)
Length - total length (header + payload) of IP packet in octets
Identifier - sequence number, which together with address and protocol fields identify each IP packet uniquely
Flag - used in connection with fragmentation: **more bit** indicates whether this fragment is the last one of a fragmented packet and **don't fragment bit** inhibits/prohibits packet fragmentation
Time-to-live - defines the maximum time in seconds a packet can be in transit across the Internet (decremented by each visited router by a defined amount)
Protocol - indicates the type of protocol (TCP, UDP, etc.) carried by the IP packet
Fragment offset - indicates where in the original user data message this fragment belongs to, measured in 64-bit units (all but last the fragment has a data field that contains a multiple of 64-bit payload)
Header checksum - carries a checksum calculated over the header bits

Figure 2. Structure of IPv4 packet (RFC 791).

2.2 IP protocol suite

The IP protocol as such is not enough to guarantee successful transmission of data in the Internet. A number of other protocols are needed to control and manage data communications. Figure 3 gives an example IP protocol stack. As seen, the IP protocol can operate over different link layers and it can carry different transport layer protocols. The straightforward stack structure is violated by some protocols that are separate protocols but seem to operate "inside" the IP or link layer protocol or between the IP and link layer. Next, some of the major Internet protocols are explained to give an overall understanding of the performance of the Internet.

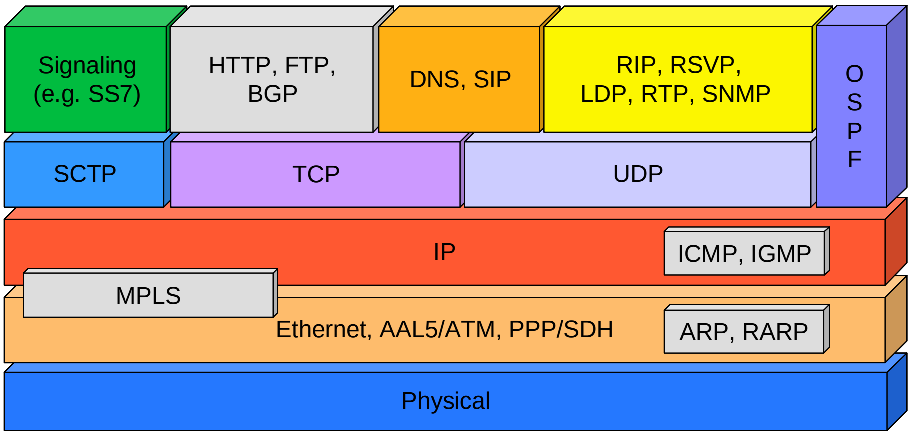


Figure 3. An example Internet protocol stack.

2.2.1 IP layer control protocols: ICMP and IGMP

ICMP - Internet Control Message Protocol (RFC 792) is a protocol, which is used by the routers to communicate with hosts and other routers. ICMP messages are generated in response to errors in IP datagrams or for diagnostic or routing purposes. Programs like `ping` and `traceroute` let users communicate directly with routers. For example, if a host is unreachable, an ICMP message with this information is returned to the sender. Examples of other possible messages are time exceeded, echo request/reply and router solicitation/advertisement.

IGMP - Internet Group Management Protocol (RFC 3376) is used by IP nodes to report their IP multicast group memberships to neighboring multicast routers. The multicast routers multicast Membership Query messages to discover, which groups have members on their attached local networks. Hosts respond to a Query by generating Membership Reports reporting each group to which they belong to on the network interface from which the Query was received.

2.2.2 Transport layer

Since each IP packet is processed independently, there are no mechanisms in a pure IP network to prevent routers and links from being overloaded. Heavy overload causes high packet loss rate and, if there were unlimited temporary storage, extremely long delays. In this sense, *a pure IP network is unstable*. Stability is brought to the system by the **Transfer Control Protocol** (TCP/RFC 793). TCP works end-to-end and it is one of the core protocols of the Internet. The protocol guarantees reliable and in-order delivery of data streams. TCP uses a number of flow-control mechanisms to achieve high performance and avoid congesting the network. TCP uses the notion of port numbers to identify sending and receiving applications. All sent packets are acknowledged by the receiver. It is quite remarkable that TCP performs so well although its only way (in its present form) to observe a congestion inside the IP network is to miss the acknowledgement of a lost packet. When this happens, the source reduces its transmission speed radically.

TCP's flow control is not adequate for real-time applications like voice and video. This will become a serious problem if file transfer ceases to be the dominant Internet application type, or if high availability is required for real-time services over the present-day Internet. **User Datagram Protocol** (UDP/RFC 768) is the protocol for time constraint traffic. **UDP** is stateless in nature, it lacks reliable packet transmission and it does not offer ordering guarantees. UDP uses the notion of port numbers, such as TCP, to identify sending and receiving applications.

Another transport layer protocol worth to mention is **Stream Control Transmission Protocol** (SCTP/RFC 2960), which is a unicast transport layer protocol that provides similar services as TCP, ensuring reliable, in-sequence transport of messages with congestion control. While TCP is byte-oriented, **SCTP** deals with framed messages. SCTP was originally intended for the transport of telephony control (SS7) protocols over IP.

2.2.3 Routing protocols: OSPF, IS-IS and BGP

In order for the routers to direct incoming IP packets to desired destinations, routers implement routing functionality. Routing protocols are used in collecting information of the network topology and routing algorithms are used in computing routing information based on the collected information and some rules. The computed routing information is then stored in routing tables.

The routing protocols are divided roughly into two groups based on the scope of operation, i.e., Interior Gateway Protocols (IGP) and Exterior Gateway Protocols (EGP). The IGPs are used inside an AS and the EGPs operate between ASs.

Open Shortest Path First (OSPF/version 2; RFC 2328) is an example of the IGP protocols. It implements a distributed routing algorithm enabling each router of an AS to maintain a full picture of its routing domain. This is done by frequent exchange of messages with other routers of the domain. OSPF uses Dijkstra's algorithm to compute the shortest path tree by using cost as its routing metric. All routers in the same area construct an identical link state database of the network topology. Routers form adjacencies when they have discovered each other. OSPF uses both unicast and multicast to send Hello Packets and Link State Advertisements (LSAs). OSPF uses IP directly (IP protocol 89). OSPFv3 (RFC 2740) specifies support for IPv6. OSPF is used within an Autonomous System and its sub-networks.

A widely used alternative of OSPF is the Intermediate System-to-Intermediate System (IS-IS) protocol (ISO 10589, RFC 1195). Like OSPF, IS-IS uses Dijkstra's algorithm for identifying the best path through the network. The use of IS-IS in TCP/IP networks ("Integrated IS-IS") is specified in RFC 1195. IS-IS is widely deployed by network operators, because is runs over OSI layer 2 protocols and disturbances on IP layer, such as Distributed Denial of Service (DDoS) attacks, do not affect IS-IS.

Border Gateway Protocol (BGP/RFC 4271) is an example of the EGP routing protocols. **BGP** is designed for exchange of routing information among ASs. Each AS has one or several border routers that serve as gateways to the outside world and advertise in both directions what networks and at what cost are reachable through each border router.

BGP maintains a table of IP address prefix to path mappings, which designate network reachability between ASs (a path consist of a list of AS numbers). Instead of using link or path metrics it makes routing decisions based on network policies or rules.

Peering between BGP routers, also called as 'BGP speakers', is manually configured. TCP is used for all transport. The peers send periodical (60 second interval by default) KEEPALIVE messages. They also exchange route UPDATE messages that carry the address prefix/path mappings that they know. This information is flooded to the whole network. Each AS advertises its original prefixes to its peers. The source for routing updates may be manual configuration or other routing protocols, which may have different priorities in the routing policy (e.g., Cisco's "distances").

When BGP runs between ASs, it is called Exterior BGP (EBGP), and within an AS, it is referred to as Interior BGP (IBGP). If the role of a BGP router is to route IBGP traffic, it is called a transit router. Routers that are located on the boundary of an AS and use EBGP are called border or edge routers.

All routers within a single AS and participating in IBGP routing must be configured in a full mesh, which leads to a scaling problem. Route reflectors (RFC 2796) and confederations (RFC 3065) have been defined to alleviate this problem.

Security Issues: BGP does not protect the integrity, freshness, and origin authentication of messages. It neither validates the authorization of announcing reachability information nor validates authenticity of that information. If the underlying TCP traffic is not encrypted, a passive attacker as an eavesdropper can learn about business relationships between AS operators. An active attacker as a man-in-the-middle can insert, delete, modify, and replay messages. A spoofed message can be used to crash the BGP state machine or advertise false IP address origin (a.k.a. prefix hijacking). Spoofed messages may also be used to flood false updates and fill routing tables. Attributes or path vectors of update messages can be tampered with to subvert routing. All these can also be used in DoS attacks. A router that goes offline will reset its route table, thus causing route flapping. All TCP and ICMP attacks can be applied to BGP traffic. A misconfigured router (e.g. false export, origin, or filter configuration, like false prefix de-aggregation) may cause a “black hole” in the global Internet.

2.2.4 Switching and connection management below the IP layer

The IP provides forwarding of datagrams from router to router toward the destination host. It is often considered advantageous to have as few router hops as possible. The number of router hops can be reduced by building new direct links between router pairs. This can be done without extending the physical network topology, if links can be concatenated on a protocol layer below IP layer. In fact, one can have different topologies on several layers below IP.

Multiprotocol Label Switching (MPLS) has been developed to speed up packet forwarding in IP networks. MPLS is a packet-switching mechanism that is considered as an OSI “Layer 2.5” protocol, which emulates properties of a circuit-switched network. It can be applied in carrying different sorts of traffic, including IP packets, native ATM, SONET, and Ethernet frames (e.g., IP VPNs and Layer 2 pseudo wires).

MPLS resembles ATM in the sense that it uses link-specific 20-bit labels in a similar way as ATM uses link-specific VPI/VCI identifiers. MPLS offers traffic engineering capability as available in ATM networks. MPLS may use an existing ATM network infrastructure, as its labeled flows can be mapped to ATM VPI/VCIs, and vice-versa.

When an unlabeled packet enters an MPLS ingress router, the router determines the forwarding equivalence class (FEC) of the packet and inserts label(s) in the packet’s newly created encapsulating MPLS header. The labeled packet is propagated along a Label Switched Path (LSP) that is functionally equivalent to a virtual circuit. At the exit, a Label Edge Router (LER) (or the penultimate LSR) pops the label and forwards the packet as a standard router. MPLS connections are set up and torn down by a distributed hop-by-hop protocol, but usually the paths are designed in a centralized fashion within an AS.

In a push operation a new label can be pushed on top of the existing label, effectively “encapsulating” the packet in another layer of MPLS. This allows hierarchical routing of MPLS packets, e.g., between ASs. Notably, this is used by MPLS VPNs.

The burden of IP routing can be alleviated also by building dense logical topologies on optical networking layers: routers can be by-passed even without converting the optical signal into an electric form. The ideas of MPLS connection management have been generalized in Generalized MPLS (G-MPLS), where identifiers of wavelengths in Wavelength Division Multiplexing (WDM) and even dark fibres in a fibre bundle are considered as labels. This approach offers a unified treatment of several layers of logical links built with different technologies.

Asynchronous Transfer Mode (ATM) can in pure IP context be seen as one way to realize MPLS, but one should remember that ATM is also a fully specified end-to-end connection-oriented networking concept generalizing the telephone network.

2.2.5 Domain Name System/Service (DNS)

Domain Name System (DNS/ RFC 1035) is not an as integral part of the Internet architecture as the datagram principle and the routing protocols, but its role is absolutely necessary to build up a usable network that supports some higher level logical notions like names, references and meanings, including the requirements set for interfaces to human beings. As the interworking between the conventional telecommunications networks and IP networks increases, there will also be a need to set up sessions connecting the different networks. ENUM is an IETF defined concept to convert E.164 numbers to DNS supported domain names.

DNS is a huge distributed database, or directory, of alphanumeric names given to most Internet addresses. DNS is hierarchically organized so that each host knows, as part of its network configuration, the address of at least one DNS server. If the first server does not know the meaning of an alphanumeric address name, it transfers the query to another server, either a higher one or one closer to the destination, and this continues until the answer is found or recognized as unknown. The root zone of the DNS is managed with 13 root name servers, which are named from A root server to M root server and compose the highest level of the DNS hierarchy. Thirteen is the maximum number of the root servers possible in the current DNS architecture, because 13 addresses at most can be fit inside a 512-byte UDP reply packet.

DNS can already be considered as being in a crisis - see, for example, the informational IETF RFC 3467 by J. Klensin. There are about 10 million sub-domains in the .com alone. On one hand, good new English names have become difficult to find, and on the other hand, the use of non-English alphabets is not possible. It seems reasonable to expect that DNS will be complemented by some much more advanced directory system in the foreseeable future.

One major operational threat is Distributed Denial of Service (DDoS), which means that a lot of requests, e.g. ping-messages, are sent to disrupt most of the root servers. In the worst case, this kind of attack could paralyze the DNS and hence all applications, such as email, web, FTP, and instant messaging. To prevent this kind of problems, operators have started to deploy root servers using anycast addresses that allow multiple machines in different network locations to look like a single server. There had already been replicas of the root servers, but the difference was that previously there was only one main server and the others served as mirrors or backups.

Anycast enables setting up identical copies of existing servers, which share the same IP address and exactly the same data. The use of anycast reduces the average network latency between client and server across the entire Internet providing better service to a larger number of users. According to

CAIDA's estimate, there are actually close to 100 physical DNS root servers in 2004, even though there are only 13 root server addresses.

DNS pollution covers a range of data traffic that should not occur in the Internet at all. The DNS pollution is usually somehow pointless machine-generated traffic, which easily loads the root name servers. Examples of DNS pollution types are *A-for-A queries*, *RFC 1918 leak*, *invalid TLD*, *identical queries* and *repeated queries*, as well as *referral-not-cached queries*.

An A-for-A query happens when a DNS client asks for an address even though it already knows it. The RFC 1918 leak refers to the private addresses that should not be visible in the global Internet. However, there are millions of DNS packets sent daily to name servers outside private networks requesting or containing information on RFC 1918 addresses. A query for a name not matching an existing Top Level Domain (TLD) is classified as an invalid TLD. Some of the unknown TLD queries result from normal spelling errors by users, but most of them result from local configuration errors.

Identical queries and repeated queries enquire the same name, type, and class as queried before. The only difference is that the identical queries have the same ID whereas the repeated queries have different IDs. Repeated queries may result from a broken name server or client, in the worst cases the timeout before repeating the query is too small generating vain reproduction of queries. A referral-not-cached query occurs when a client sends a query for a different name in the same zone as its previous query. For example, if a client queries first *icann.org* and gets a referral to *.org* TLD name server, and then queries *ietf.org* from the root, the latter query is needless.

2.3 Some other important protocols

In addition to the protocols described in section 2.2, there are a number of other protocols that are not essential part of the TCP/IP protocol suite, but which in some other way are important in managing the transport of information via the Internet. Next some of them are introduced.

ARP – Address Resolution Protocol (RFC 826)

ARP is used to map an IP address to a hardware interface address (MAC address), e.g. 48-bit IEEE 802 address. Although ARP is an essential part of a working IP network, it is worth noticing that those packets are not IP packets.

Address resolution is needed when a source node wants to send an IP packet to a target node or interface on the same subnet such that the IP packet must be encapsulated in a link-layer frame that needs to contain the hardware address of the target interface. If the source node does not know the hardware address of the target, it queues the outgoing IP packet and multicasts (e.g. using broadcast Ethernet address) an ARP Query containing the target destination IP address to the link. The target replies with a unicast ARP Response that contains the requested address. The Query also contains the source node's address mapping - so the destination can learn them during the transaction. Nodes maintain a soft state ARP cache that contains the address mappings.

If the corresponding hosts do not reside on the same broadcast link, but a router between the hosts is configured to mediate packets between the hosts, the router will respond to the ARP query with its own MAC address – this is called proxy-ARP.

Security Issues: ARP messages are not protected. A node can send a fictitious ARP response (cache poisoning), thus impersonating any other host or router on the link (a.k.a. address spoofing). This can be exploited e.g. in a man-in-the middle or DoS attack. Another potential vulnerability is exhaustion of the ARP table space, which can be exploited in DoS attack. There are no standardized security mechanisms. Attacks can be prevented by relying on link-layer security or by using static tables (the latter does not scale). Address spoofing can be detected by monitoring address collisions.

RSVP(-TE) – Resource Reservation Protocol - Traffic Engineering (RFC 2205, RFC 3209)

RSVP signals QoS requirements and reserves router resources to provide a requested service to all nodes along a data path. RSVP works in conjunction with unicast and multicast routing protocols. Unidirectional reservation requests for a flow follow the same path through the network as the data comprising the flow. The source node initiates the reservation procedure by transmitting a `PATH` message that describes the source traffic characteristics in terms of peak data rate, average data rate, burst size, and minimum/maximum packet sizes. The destination node reserves the resources (soft state) along the path with an upstream `RESV` message that is propagated hop-by-hop towards the source.

RSVP-TE extends RSVP with a set of extensions that enable RSVP to be used for traffic engineering in MPLS environments. The primary extensions add support for assigning MPLS labels and specifying explicit paths as a sequence of loose or strict routes (thus overriding default paths or paths built by using routing protocols). RSVP-TE operates in “downstream-on-demand label advertisement” mode with “ordered LSP control”.

RTP – Real-time Transport Protocol (RFC 3550)

RTP defines a standardized packet format for delivering unicast or multicast audio and video over the Internet. It is frequently used in streaming media systems (in conjunction with RTSP) as well as videoconferencing and push-to-talk systems (in conjunction with SIP). The specification includes RTP Control Protocol (RTCP). The protocols normally use UDP. The services include payload-type identification, sequence numbering, time stamping, and delivery monitoring. The protocols themselves do not provide mechanisms to ensure timely delivery, QoS guarantees, or congestion control.

Application-specific usage of RTP is defined as profiles: RFC 3551 (RTP/AVP) defines a profile for Audio and Video Conferences with Minimal Control; RFC 3711 (SRTP) defines the Secure Real-time Transport Protocol profile.

FTP – File Transfer Protocol (RFC 959)

FTP is used to access files on a remote server. FTP uses two TCP connections: one for control (server port 21) and another for data. There are two modes for establishing the data connection: in active mode, the server initiates the data connection (from port 20) to a port indicated by the client, whereas in passive mode the client connects to a port indicated by the server. Due to firewalls and simple NATs, active mode is typically not usable.

Security Issues: FTP as such is not secure. All data, including the user’s credentials, are transmitted as clear text. Therefore, SSH File Transfer Protocol (SFTP) over SSH-2, FTP over SSH, FTPS (RFC 4127), or SCP should be used instead.

HTTP – Hypertext Transfer Protocol (RFC 2616)

HTTP is an application-level protocol for distributed, collaborative, hypermedia information systems. It is a stateless protocol, which can be used for many tasks beyond its use for hypertext, such as distributed object management systems (like network management), through extension of its request methods, error codes and headers. A feature of HTTP is the typing and negotiation of data representation.

SMTP – Simple Mail Transfer Protocol (RFC 2821)

SMTP is the protocol for e-mail transmission across the Internet. DNS MX (Mail eXchange) records are used to determine the SMTP server for the email recipient’s domain name. SMTP does not support fetching emails from a remote server on demand. To do this, email clients use POP3 or IMAP protocols. SMTP was originally purely ASCII text-based. Standards such as MIME have been developed to encode binary files for transfer with SMTP.

SNMP – Simple Network Management Protocol (RFC 1157)

SNMP is a protocol for managing nodes in the network. Network administrators use SNMP to monitor and map network availability, performance, and error rates. SNMP uses UDP. The SNMP architecture consists of a collection of network management stations and network elements. Management stations execute management applications, which monitor and control network elements that contain agents responsible for performing the management functions on managed objects. The types of managed objects are defined in the Structure of Management Information (SMI), which applies a subset of the ASN.1 language. The managed objects are comprised in a virtual database – Management Information Base (MIB).

RFC 1441 introduces SNMPv2 and RFC 3411 defines SNMPv3. The corresponding, more complex, ISO-defined network management protocol is Common Management Information Protocol (CMIP).

2.4 On user access

Although data communications networks are said to converge, the user access seems to diverge. Numerous wire-line and wireless access techniques have been introduced and new ones are constantly being developed. As for the wire-line access, the major point is to leverage the old-established access infrastructure, i.e., the copper based twisted-pair wires, and provide users with adequate bandwidth for their future needs. The target of the various wireless solutions is to offer the required bandwidth but also to support the mobility of users. The foremost technical challenge in wireless solutions is interworking of the different networks, i.e., seamless handover/roaming.

Provided that the transport distance is long and the required bit rates are high (at least several tens of Mbit/s), the optical wire-line solutions seem to be the way to go in broadband access. Optical solutions still seek for economical solutions, but in the long run the optics is anticipated to become general.

What ever the physical access technique, home users normally have a broadband access subscriber connection, e.g. an Asymmetric Digital Subscriber Line (ADSL) connection, for their data communication needs. The use of a broadband connection entails that the user has a broadband modem (ADSL modem), which connects to the operator's Digital Subscriber Line Access Multiplexer (DSLAM), as shown in Fig. 1. A DSLAM concentrates a large number of users to the operator's central office. At the central office, the DSLAM is connected to a router.

At the user premises, the ADSL modem is connected to a small router or switch that implements the Network Address Translation (NAT) protocol, which enables multiple network devices to share a single IP address generally provided by the Internet Service provider (ISP). The router/switch may also have the ability to provide port-based control, firewall management and Dynamic Host Configuration Protocol (DHCP) service for all the user's data communications devices. DHCP allocates an IP address for the user's computer as the computer is turned (powered) on. The allocated address is normally valid until the computer is turned off. These functions, including routing and switching, can also be implemented in the ADSL device itself.

2.5 On the migration of legacy services to IP

The IP networks are foreseen to integrate the legacy services into the same service infrastructure as the new services, specially tailored for IP transport. When a legacy service, like voice, originates or terminates in a legacy network and is carried partly in an IP network, the integration becomes difficult. Timing, bit rate accuracy and delay constraints of the legacy networks and their services are not inherently supported by the IP network. Numerous additional protocols and other amendments need to be developed to maintain the legacy service quality.

An alternative solution is to redevelop the legacy services for the IP networks, in a similar way as the telephony service was converted to Voice over IP (VoIP) service. The quality of the new service is not always the same as it used to be, but it fits better for the IP networks and does not suffer from the deficiencies of the IP transport. The new implementations of the old circuit switched services entail development of appropriate signalling and service description methods, such as SIP and SDP.

SIP – Session Initiation Protocol (RFC 3261)

SIP is a protocol for initiating, modifying, and terminating an interactive user session that involves multimedia elements such as video, voice, instant messaging, online games, and virtual reality. SIP works in concert with other protocols like RTP and is only involved in the signaling portion as a carrier of SDP. SIP is used by 3GPP in the IMS architecture, and it is a key protocol in VoIP and future media services.

SDP – Session Description Protocol (RFC 2327)

SDP is a format for announcing streaming media sessions and related initialization parameters. SDP was initially a component of the Session Announcement Protocol (SAP), but found other uses in connection with RTSP, SIP and just as a standalone format for describing multicast sessions. SDP describes a session in terms of session name and purpose, time(s) the session is active, the used media, and information for accessing the media (addresses, ports, formats, etc).

2.5.1 Voice over IP (VoIP)

VoIP or IP telephony is a term for a variety of techniques to transport voice and video information in real time. Voice and video signals are converted into digital form and are conveyed in data packets over the Internet. Phone calls originating from or terminating to a legacy telephone network go via special gateways that carry out necessary conversions, e.g. voice coding conversion.

The basic building blocks of the VoIP systems are the signalling protocols and voice coding protocols. The signalling protocols are needed to establish and release voice connections, while coding protocols take care of the coding and transport of voice information. There are two non-compatible signalling standards: H.323 and SIP. H.323 as the older and more complicated one is more stable and versatile than SIP. Although SIP is simpler, it is a mature signalling protocol and is gaining more and more popularity. SIP is foreseen to be the signalling protocol of the 3G networks.

In order to make a VoIP call, the user needs a VoIP capable terminal and a broadband connection (at least 256 kbit/s). The terminal may be a computer equipped with a microphone and an earpiece or a special VoIP phone. A conventional telephone can also be used if the user has an IP converter. Some mobile phones implement a Java based VoIP solution, which can be used over the GPRS connection.

2.5.2 IP-television (IPTV)

IPTV is a system that delivers digital television services to customers over a broadband connection by utilising the IP protocol. IPTV is often provided in conjunction with Video on Demand (VoD) and may also include Internet services such as Web access and VoIP. This kind of package is often supplied by a broadband operator and is called Triple Play.

IPTV covers both live TV (multicasting) and stored video (VoD). Video content is typically sent in MPEG-2 transport streams which are delivered via IP multicast. IP multicast is a method that allows to send information to multiple computers at the same time. The newly released H.264 format is thought to replace the older MPEG-2. In standards-based IPTV systems, the primary protocols used for IPTV are IGMP version 2 (for live TV channel selection signaling) and Real Time Streaming Protocol (RTSP/RFC 2326) (for VoD).

Advantages of IPTV include two-way capability, not present in traditional TV distribution, as well as point-to-point distribution allowing each user to view individual broadcasts. Since IPTV systems send less information than standard analog or digital television, IPTV promises lower costs for operators and lower prices for consumers. Major television broadcasters worldwide have started transmitting their broadcast signals over the Internet. These free IPTV channels only require an Internet connection and an Internet enabled device such as a personal computer, iPod, HDTV connected to a computer or even a 3G cell phone to watch the IPTV broadcasts.

3 A conceptual framework for the dependability of IP networks

3.1 *The multi-aspect notion of dependability*

We are interested in IP networks as parts of the global Internet, a huge system that provides global connectivity of electronic packet transfer. This connectivity is a new, generic medium that is capable of merging into itself all earlier electronic media, including telephone and television. Our focus is on the generic IP medium, not on the potentially countless higher layer services. Although it is often thought that the user does not and doesn't need to understand the nature of the underlying packet layer service, we propose that a future citizen should possess a "network literacy" that allows him/her to understand adequately the heterogeneous connectivity affordances and, on the other hand, the essentially end-to-end character of IP-based services.

In assessing the dependability of this system, we first distinguish three main kinds of relevant actors:

1. The *users* who use the medium for myriads of purposes.
2. The *providers* who construct and operate the system and bear most responsibility for its proper functioning.
3. The *designers* who create the algorithms according to which the system works.

To complete the picture, we add two more kinds of actor:

4. The generic *adversary* who "arranges" failures by all possible means.
5. The *regulator authority* who sets the requirements on the dependability of the network.

The notion of *dependability* is usually defined as a collective term with a number of different *aspects*. What these aspects of dependability are varies a lot according to the context. For example, the International Electrotechnical Commission (IEC) defines dependability as "the collective term used to describe the availability performance and its influencing factors: reliability performance, maintainability performance and maintainability support performance", and adds the note that "dependability is used only for general descriptions in non-quantitative terms" [1]. The Dutch government report [4] on Internet vulnerability lists three aspects of dependability: confidentiality, integrity and availability. As a third example, Alain Villemeur [8] defines dependability as "the science of failures" and specifies that dependability includes the more specific attributes of reliability, availability, maintainability and safety. Since three respectable, rather randomly chosen sources differ this much in their definitions of what dependability is, one could feel free to select any set of aspects that suits to a given purpose. We do this to some extent in fact, but mostly we try to adapt to the definitions of [2], published in 2004 as the first paper of the new journal *IEEE Transactions on Dependable and Secure Computing*.

The aspects we shall consider are the following: *availability*, *maintainability*, *reliability*, *robustness*, *invulnerability and controllability*. Most of the above notions are collected in Figure 4.

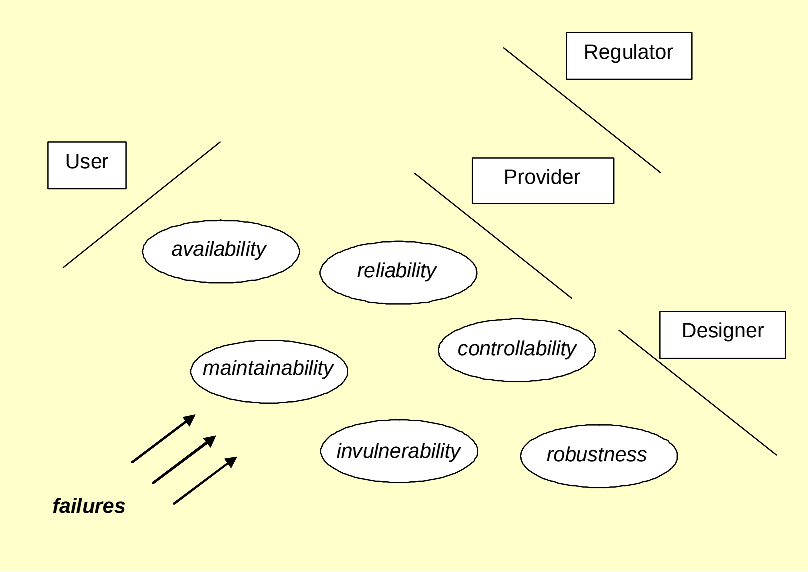


Figure 4. A conceptual framework for network dependability.

We consider these notions closer in the next subsection. Before that, we introduce them briefly and discuss their relation to some other notions related to dependability.

Availability, maintainability and reliability belong to the standard attributes of dependability. Avizienis *et al.* [2] define them as follows: availability means readiness for correct service, maintainability means the system's ability to undergo modifications and repairs, and reliability means the continuity of correct service. In our approach, we give these notions the following additional associations:

- availability is the aspect of dependability that directly concerns the user; it is also the most observable and quantifiable aspect
- maintainability refers to the provider's procedures and processes that are needed to run, renew and update the network hardware and software
- reliability is associated to the system's ability to function despite of failures of individual components.

In the context of IP networks, however, we would like to add certain notions that direct the attention to the specific nature of this particular system. Its nature is far too dynamic to be understood by using only concepts developed for such "traditional" complex systems as aircraft, power plants, or the electrical power network. Doyle *et al.* [3] point rightly out that the dynamics of the Internet cannot be understood without an adequate theoretical understanding of the multi-layer communication protocol structure. In particular, many protocols include feedback functions that make them adaptable to their environments and perform distributed optimization. An IP network is largely a self-controlled system. On the other hand, we don't trust a completely self-controlled system. Therefore we propose the following three additional aspects of dependability that have a control-theoretic flavour.

First, the *robustness of design* of the basic algorithms and protocols, meaning stability with respect to arbitrary inputs, has great importance. The self-regulation provided by TCP is an example.

Second, the special, highly dynamic character of an IP network prompts, in line of the ideas in Doyle *et al.* [3], to pay attention to a *vulnerabilities of design* that are, in fact, neglected aspects of the generally robust and optimized design. It is not sufficient to restrict to what could be characterized as "classical failures", when the system has the character and complexity of the Internet. Even when its designs were completely robust with respect to legitimate kinds of inputs, it could be extremely sensitive to certain non-anticipated "inputs" like viruses, or simply erroneous configuration by human operators.

Finally, although, or because, the self-controlling character of many protocols is inevitably a true strength of IP technology, we want to include certain amount of *controllability* to the aspects of network dependability. Controllability is crucial in special circumstances.

Two more notions that are deeply connected with dependability are *quality of service* and s*ecurity***.**

The notion of *Quality of Service* (QoS) is intimately related to the availability of a service. A human user is seldom interested in the service of packet connectivity as such, but higher layer applications that set requirements on transfer speed and delay. In this study we don't give to QoS the status of an independent aspect of dependability but consider it as an additional attribute of the packet delivery service that is or is not available.

We leave s*ecurity* here for relatively little attention, although it is an indispensable part of the satisfactory realization of most of the aspects of dependability listed above. Security means *integrity* of information and *confidentiality*, absence of unauthorized disclosure of information, and availability for authorized actions [2]. Information security is a very active and well established area, and in this paper we rather feel as our task to understand *what else* is needed for dependability than security. Like QoS, user level security can in the present context be considered as part of the service. On the other hand, extensive security mechanisms also increase system complexity and can sometimes be counterproductive.

3.2 Remarks on the selected aspects of dependability

3.2.1 Availability

In the IP networking context, the availability of service must be further specified in at least two dimensions:

- for whom: end user, Autonomous System
- what quality (bandwidth, loss, delay)

We could define for example that the availability of the Internet for a given user (end-device, autonomous system etc) is the long-term average part of time that this user has connectivity of "satisfactory quality" to most of the normally accessible end-devices connected to the Internet. The phrase "satisfactory quality" can, if needed, be specified separately for each service or service class. The terms "most" and "normally" can, in principle, be specified numerically, although the global connectivity is hard to measure and estimate.

Plain packet forwarding is not enough for the applications because the contemporary Internet architecture does not completely conform to the ideal end-to-end principle [SRC84], where the intelligence resides in the end systems. Practical use of the network requires some control functions and services delivered over specific protocols, like DHCP, DNS, AAA, SIP, and middle-boxes that embed application intelligence (e.g., NAT/ALGs, firewalls, and proxies). Each of these must work to provide a full-fledged service. On the other hand, even strongly deteriorated service is not always zero service for a user with sufficient "network literacy" who understands to switch to an application with more modest bandwidth requirements.

The cause of experienced unavailability can reside anywhere in the network, and traffic congestion may restrict the availability even when all network elements are up and running. The user does not care much about how serious this cause is from the network point of view. It is clear that a good information infrastructure must have a very high availability for its end users. Unavailability periods may be very disturbing for the user.

Unavailability of *emergency services* is not only disturbing but directly dangerous for the user. In the telephone context, special handling of emergency calls, like free availability and priority, is strictly enforced. There has so far been no counterpart to emergency calls in an All-IP network. This problem is urgent since, as it now seems, a large part of telephony will move out from the telephone networks, cellular telephone networks included.

Many questions discussed in this subsection are reflected in FICORA's regulations 29 D/2005 (call blocking probability), 28 F/2005 (performance requirements to the signalling network), 50 B/2003 (obligation to monitor the quality of service), and 33 B/2005 (securing emergency traffic, identification of the geographical location), but most of these concern only telephony.

3.2.2 Reliability

The standard approach of a reliability analysis is to break a system hierarchically into subsystems and finally into atomary components that either work or not. A high-reliability system is designed by preparing for component failures by redundancy and protection mechanisms.

In this definition, links, switches and routers, user databases and name-servers, as well as protocol software implementations are understood as "components". Crucial factors of network reliability are topological redundancy, capacity redundancy, and the protection mechanisms that can provide bypassing of failed components without observable service deterioration. Generally, the loss of a network node is much more serious than a loss of a link, since all links to that node are lost simultaneously. On the software side, the hardware protection can be paralleled with recovery from deadlocks and livelocks. Since we are considering a large network, the effects of "normal" component failures need by no means be local or restricted.

We don't count taking care of traffic congestion as an aspect of reliability, although it affects the availability of network services, as pointed out in the previous subsection. We would rather subsume this problem under the aspects of robustness and controllability.

Redundancy underlies many approaches to fault tolerance (resilience). This could be spatial (replication of components) or temporal (e.g. automatic repeat requests, ARQ). Fault removal is carried out through recovery mechanisms.

Failure recovery consists of mechanisms for healing from network anomalies (like link failure or excessive bit error rate). Recovery can be implemented at any layer of the protocol stack. An issue is how recovery mechanisms in different layers could interwork - how to avoid harmful interference between layers.

Self-healing can be classified according to the use of

- link vs. path recovery (i.e. local vs. global repair)
- pre-computed backup paths vs. dynamically computed recovery paths (i.e. protection vs. restoration)
- centralized vs. distributed computation.

A special case of recovery is "graceful restart", where a router is able to keep its packet forwarding state even if the control-plane software is restarting from a crash. Graceful restart has been specified for various routing protocols (BGP, OSPF, IS-IS) and for MPLS.

SDH, for example, includes extensive protection mechanisms, which detect, e.g., a loss of signal in a link and may signal the event to neighbour nodes using various types of alarm messages. As a rule of thumb, SDH restoration takes 50 ms with pre-defined and reserved protection paths. The actual time may vary between 5-200 ms. Ethernet and MPLS recovery can be based on the same diversity protection principles as SDH.

FICORA's regulation 27 E/2005 sets requirements for the protection of network elements according to how many users depend on them.

3.2.3 Maintainability

The notion of *maintainability* refers to the necessity of renewal by all real systems, both natural and technical. The maintainability of a communication network is characterized by the possibilities to extend, renew and update the network. The maintenance aspect must be taken into account in the design of high availability systems. so that all components can be set in downstate for renewal and maintenance without noticeable disruptions of service. As regards the basic protocols, only smooth transitions come into question, since it is not possible nor safe to install and configure new software of, say, a router at many sites simultaneously.

Every Internet user has observed that software updates as well as hardware replacements often cause non-negligible interruptions of service even when the proceed as planned. It is also not rare that the breaks are extended by unexpected difficulties in installation and configuration. This concerns much more the periphery of the network and private networks than the core networks, where critical components are duplicated so that uninterrupted service is possible. In an All-IP world, however, similar maintenance principles should be extended much farther in the network periphery.

3.2.4 Robustness of design

Unlike the notion of reliability, which in this paper has an external and largely statistical character, the robustness of design is an internal and non-statistical characteristic of the system. A system design is robust, if it performs on a satisfactory level with all possible inputs and noise perturbations, and converge to stability also in situations when parts of the system are down. A system design is provably robust, if its robustness can be mathematically proven in a model that includes all relevant aspects of the system. Provable robustness of basic protocols should be the goal in the design of the information infrastructure [3].

Robustness is especially an attribute of the feedback systems realized in the Internet protocols. In addition to normal operation, robust solutions are needed for exceptional situations to avoid instability, hysteresis and avalanche phenomena. A robust network design should cope with naturally emerging traffic overloads as well as with malicious intentional overload attacks.

The TCP protocol is no doubt an instance of successful robust design of a completely distributed (that is, end-to-end) protocol whose end effect is a rather fair and effective sharing of bandwidth in the whole network. Mathematical analysis of TCP-like protocols has advanced a lot during this decade, and their effects are rather well understood both qualitatively and quantitatively.

Much less is rigorously known about routing protocols. The experience is that intra-domain routing with OSPF or IS-IS normally converges within a few seconds after a link down event. The restoration times with the inter-domain routing protocol BGP may be of the order of minutes. Qualitatively, the intra-domain behaviour can be characterized as robust: if two end systems are connected by forwarding paths in both directions, a route between them is established automatically through a fully distributed protocol. However, temporary deterioration of service quality (delay, packet losses, and reordering) is typically unavoidable, and IP routing does not provide bandwidth recovery. BGP is more problematic even qualitatively.

This can be contrasted with connection-oriented approaches applied on lower layers discussed in subsection 3.2.2. In the present conceptual framework, we subsume lower layer protection to the reliability aspect rather than the robustness of design aspect.

3.2.5 Vulnerabilities of design

We propose to define the notion of vulnerability of design of a communication network in a way that distinguishes it from “ordinary” system failures coped with by the reliability aspect of dependability. In spirit of [3], we say that the vulnerabilities of a system are possibilities of behaviour that lead to serious degradation of performance by causes that come from outside of the system aspects taken into account in the design.

It is important to distinguish such vulnerabilities of design from “simple” component failures in a classical application of reliability theory. Software errors and failures are already difficult to deal with in reliability theory, because their disastrous effects can be completely disproportional to the “size” of the error itself. However, the most intriguing vulnerabilities come from unexpected ways of using the system’s own, designed possibilities *inside* the system. The realization of a vulnerability leads, hopefully, to an improved or new design. This process can be considered as gradual adaptation of the system to its environment.

As an important example, the BGP routing protocol is highly vulnerable against erroneous configuration. The problem is alleviated by procedures and rules directed to avoidance of such errors. Cracker activity has dangerous potential in the Internet, because a person knowing right passwords can generate network control messages from his/her normal terminal (this is impossible in the telephone network, where the user and control planes are strictly separated). The basic architecture of IP networks makes it possible to operate anonymously and cover one's tracks, which is fought by administrative traffic filtering and traceback systems.

FICORA’s regulation 13/2005 presents several explicit rules on route advertisements in order to prevent the emergence of ”black holes” and other vulnerabilities in the BGP routing system.

3.2.6 Controllability

The controllability of a communication network is characterized by the possibilities of single agents (mainly: the network operators) to accept, reject, or route traffic offered to the network, and to open and close individual services or the whole network.

The basic architecture of the Internet includes no centralized control, and the Internet as a whole seems to be uncontrollable. Several autonomous systems have physically global dimensions. As regards global connectivity, the Internet has no single top level of routing hierarchy. However, one should note in this context that a fairly large part of all Internet traffic traverses through some of the very few biggest Exchange Points that switch traffic between large ASs.

The routing protocols work through messages, which the routers send to each other independently in a distributed fashion. Approximately fair sharing of resources is realized in a totally distributed way by the end-to-end control provided by TCP. Centralized control is used mainly on the lower layers: MPLS, ATM, SDH, and optical layers (WDM and fibre). The telephone network also works with centralized control.

The main tool of exerting additional control in an IP network is *traffic filtering*, which means removal of packets with specified characteristics, typically origin or destination addresses or TCP port numbers. Filtering is a logically heavy operation, depending on the length of the list templates to be filtered. It is not clear how filtering scales when traffic increases by orders of magnitude.

FICORA's regulation 50 B/2003 sets several kinds of control requirements. Regulation 27 E/2005 obliges the provider to possess the ability to trace back malicious calls.

3.3 Classification of failures and their causes

A *failure* of a component or system means its transition to a state where it does not any more deliver its correct service. Thus, a failure is on the "surface" of the object under consideration and has a cause somewhere "deeper" inside the object or in its relations to the rest of the world. Network software has the particular feature that it runs in numerous copies at a large number of routers and other processors and thus has a strongly non-local character. In this paper we make just a few remarks that show how many are the types of failures and their causes that should be taken into account when assessing the dependability of a large IP network.

3.3.1 Failures, errors and faults in computing systems

Avizienis *et al.* [2] presents a detailed conceptual structure of various notions related to failures. They define *error* as the deviation of a state of the system from its correct service state. Note that an error need not be manifested as a failure. The cause of an error is called a *fault*. Cross-tabulating dichotomies like hardware/software, natural/human-made, non-malicious/malicious etc., [2] ends up with a taxonomy of as many as 31 classes of faults, although most of the formal intersections are empty (for example, the nature is never malicious).

The size of software needed in a big router is considerable, tens of megabytes. The update intervals vary from a few months to a year. Careful procedures must be applied by the updates. Since the core software is usually well tested by its producer, configuration errors are more frequent causes of failures than software bugs.

To give an idea of the frequencies of component failures in IP networks, we mention some statistics of the reliability of ISP networks, collected in [6] (MTTF reads "mean time to failure"):

- In one ISP (Sprint), 20% of links have MTTF < 1d
- In one ISP (Sprint), 70% of links have MTTF < 10d
- In the Internet backbone paths, mean time to fail-over is ~ 2d
- In the Internet backbone, ~ 20% of the paths stayed unchanged in 5d
- In the global Internet routing table, 0.2-1% of the entries suffered from configuration errors.

3.3.2 Physical faults

Physical faults appear rather frequently in components that exist in large numbers like linecards, whereas the hardware of big routers and switches is highly reliable. They are also typically renewed much before their age would that require, because the traffic has so far been continuously increasing and upgrading to higher capacity is necessary.

Accidental cable cuts occur regularly here and there. This is a classical class of faults that seems hard to get rid off. The implications of a cable cut can be surprising, if fibres that should secure each other go in the same duct. The problem is well recognized, but not completely solved in practice. In the network periphery the network has usually a tree topology and the links are not secured. Thus, a cable cut interrupts the service for all users downstream from the cut.

Physical losses of network nodes can be caused by accidents like fires or, less dramatically, faults in air-conditioning of the router room. Such scenarios are classical in reliability assessments, but All-IP solutions with their all-eggs-in-same-basket character prompt to search for new, highly resilient solutions that are not realised in present IP networks outside the core.

3.3.3 Attacks

Fighting with intrusion and denial of service attacks are nowadays part of the daily life of network operators. Attacks are mostly connected to vulnerabilities of some kind:

- neglected information security
- errors in network software
- errors in terminal equipment software
- vulnerabilities of design.

3.3.4 Economy

We make two brief remarks on the relation between dependability of the network and economy. First, improving the dependability is costly, and therefore this process must be enforced by an authority. In Finland, this is taken care by the Finnish Communications Regulatory Authority (FICORA).

Second, financial difficulties of a network operator easily lead both to reduction of expert personnel and to general deterioration of the processes that should guarantee the dependability of the network. The extreme phase, operator bankruptcy, is a threat handled by the Finnish law so that the service should not be interrupted abruptly but moved to another operator. Nevertheless, some big companies prefer to guarantee “bankruptcy-secured” connectivity by connecting their private network to more than one public operator.

3.3.5 Politics

Before its commercial opening in 1990s, the Internet connected mainly the research world. Its management was based on trust and collaboration that exist in the scientific community. Its U.S. origin was very strongly present. Commercialization brought with it spam and attacks as normal phenomena, but IP technology and the Internet have been able to strengthen the architecture correspondingly. The Internet has connected the world in an unforeseeable way with very strong impact on democracy. The role of the U.S. in IP technology and Internet usage (think about the role of Google) is still very dominant. One should analyse, how the Internet would work under seriously hostile relations between countries or alliances.

4 How to measure dependability?

The measurement of dependability is important due to many reasons. It is needed for evaluating and understanding the costs and other consequences of failures and disturbances. Dependability indices are also useful in designing the structure of the network and selecting the redundancy of components and subsystems. Furthermore, the planning of maintenance may require quantitative information of the systems dependability.

The dependability can be measured in different ways. One possibility is to measure the observed dependability on the basis of the knowledge on occurred disturbances. These kind of measures are already used as a part of Service Level Agreements (SLA), which are made between clients and network operators. Although the SLAs are commercial agreements freely negotiated between the partners, it would be beneficial that "best practice" recommendations would develop that contain also measurable and verifiable criteria on the fulfillment of the agreement.

Other dependability indices are based on a predictive approach: the future dependability performance and failure behavior is predicted by appropriate models. In addition to measures based on failure behavior, it is possible to develop indices describing the structure of the network, e.g. connectivity, and structural importances of the network components.

The basic conventional dependability concepts and their relationships with the systems performance is described in Figure 5.

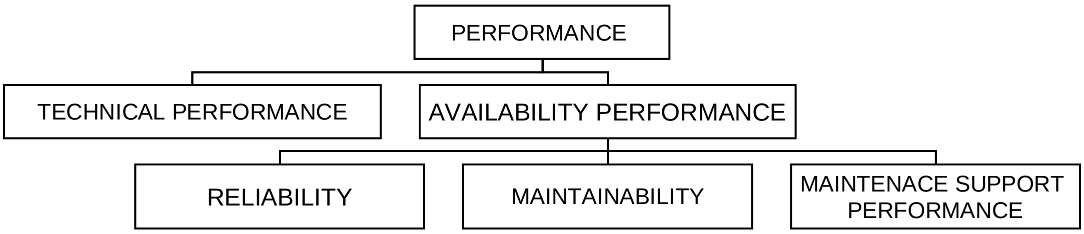


Figure 5. Reliability concepts.

The overall performance of a system consists of technical performance and availability performance. Technical performance refers to technical characteristic such as capacity, efficiency etc. The availability performance refers to the security by which the technical or engineering characteristics are kept up during the systems life-cycle. Availability is approximately the part of the time which the system is operating according to its specifications. It consists of reliability, the system's capability of operating without failures (measured e.g. by mean time between failures, or failure rate), maintainability and maintenance support performance which characterize the capability to bring a failed system in to operation (measured e.g. by mean down-time). The above dependability concepts are suitable for any type of systems, independently on the technology. However, they are too general for defining dependability indices.

The form of a dependability index depends on the level on which the index is defined. It is not useful or possible to measure the availability performance of the whole network with the same measure as that of one single component. Furthermore, the form of the index depends also on its use: an index which is good for maintenance planning may not work for design purposes.

The dependability measures can be defined on various levels of the network. The simplest measures can be defined for single hardware components. Their dependability performance is rather exhaustively characterized by failure rate or mean time between failures (MTBF). The maintainability of single components can be measured by mean time to repair (MTTR), and the availability is the part of the time which the component operates according to its specifications. These measures can be determined either on the basis of operational experience of simple component reliability models.

In many cases the network hardware components are doubled by stand-by components. The reliability performance of such standby systems can easily be determined by using rather simple reliability models; in that case the reliability indices are often predictive or computational. In principle, they correspond rather directly to the indices of single components.

The functions of network components are based on proper functions of software. If the component reliability indices are based on observed performance, the contribution of software is basically similar to that of hardware, and it can be measured in the same way. The predictive reliability measures of software are more problematic. There are no generally accepted measures or models for software reliability/availability.

The reliability indices of single nodes and single connections can also be defined in principle in the same way as for components. However, the nodes and connections require the proper operation of several hardware and software components, i.e. they are systems. The determination of reliability indices for this kind of objects requires reliability modelling, which takes into account possible dependencies between the components and the reliability structure of the systems. Suitable measures for this kind of subsystems are failure rate, availability or duration curves. Duration curves give the probability (or percentage of time) that the system provides certain level of service during the time period under consideration.

The reliability indices for a whole network (or part of network) are more problematic. The number of clients in a network is large, and the service level of each client should me monitored or measured. The structure of the (core) network is complicated, the logical and physical network do not have the same structure. The functionalities of network depend on several protocols. It is not easy to characterize the reliability structure of the network, and to build a reliability model of network. The model-based indices require further development work. It seems, however, that measures describing overall service level can be defined. They could measure, for example, the expected or mean number of clients (connections) being served at least on certain service level certain percentage of time. Another possibility could be the evaluation of the long-term average part of time that the network provides full connectivity of satisfactory quality between all nodes of the network.

Table 1 summarises the possible measures for networks.

Table 1. Reliability indices

Level of measurement	Reliability indices	Comments
Components	• Observed failure rates, component availabilities and down times • Modelled/predictive failure rates, component availabilities and down times	• Require component reliability models • Measures for software components and human actions may be problematic
Functions of components	• observed failure probabilities, failure rates and availabilities • modelled failure probabilities, failure rates and availabilities	
Simple subsystems, single connections	• observed and modelled failure rates, component availabilities and down times • duration curves, the probability (or percentage of time) that the system provides certain level of service during the time period under consideration	• systems have often redundancy, which must be modelled
Networks	• e.g. expected or mean number of clients (connections) being served at least on certain service level certain percentage of time • the long-term average part of time that the network provides full connectivity of satisfactory quality between all (or certain set of) nodes of the network.	• Models of network structure required • Dependencies on protocols must be taken into account • Impact of human errors must be taken into account

The measurement principles discussed above concern only un-planned unavailability periods of the system. In addition to these, the network system may be unavailable because of planned maintenance and configuration, which should be taken into account in defining dependability indices.

5 References

[1] Reliability Division of the American Society for Quality, http://standardsgroup.asq.org/dependability/tc56/faq.html.

[2] A. Avizienis, J.-C. Laprie, B. Randell and C. Landwehr. Basic concepts and taxonomy of dependable and secure computing. IEEE Trans. Dependable and Secure Computing **1**(1), 11-33, 2004.

[3] J.C. Doyle, J. Carlson, S.H. Low, F. Paganini, G. Vinnicombe, W. Willinger, J. Hickey, P. Parrilo and L. Vandenberghe. Robustness and the Internet: Theoretical Foundations. Draft. March 2002. http://netlab.caltech.edu/internet.

[4] Internet vulnerability. Joint report of Dutch Ministry of Transport, Public Works and Water Management, and Dutch Ministry of Economic Affairs. July 2001.

[5] M. McLuhan. Understanding Media - The Extensions of Man. McGraw-Hill, New York, 1964.

[6] D. Pei, D. Massey and Lixia Zhang. A framework for resilient Internet routing protocols. IEEE Network **18**(2), 5-12.

[7] J. Salzer, D. Reed, and D. D. Clark, "End-to-end arguments in system design," ACM Transactions on Computer Systems, 2, no. 4, Nov. 1984, pp. 277-288.

[8] A. Villemeur. Reliability, Availability, Maintainability and Safety. John Wiley & Sons, 1991. Two volumes.