Network Layer

We have briefly seen the Network Layer header in the IP to see its addresses

Network Layer

We have briefly seen the Network Layer header in the IP to see its addresses

We now need to look at the Internet Protocol (network layer) in more detail

Network Layer

We have briefly seen the Network Layer header in the IP to see its addresses

We now need to look at the Internet Protocol (network layer) in more detail

It is the basis the Internet is built upon

Network Layer

We have briefly seen the Network Layer header in the IP to see its addresses

We now need to look at the Internet Protocol (network layer) in more detail

It is the basis the Internet is built upon

It is actually relatively simple, but allows more complex stuff to be layered on top

Network Layer

We have briefly seen the Network Layer header in the IP to see its addresses

We now need to look at the Internet Protocol (network layer) in more detail

It is the basis the Internet is built upon

It is actually relatively simple, but allows more complex stuff to be layered on top

We shall start by describing IP version 4, IPv4

Network Layer

We have briefly seen the Network Layer header in the IP to see its addresses

We now need to look at the Internet Protocol (network layer) in more detail

It is the basis the Internet is built upon

It is actually relatively simple, but allows more complex stuff to be layered on top

We shall start by describing IP version 4, IPv4

And talk about IPv6 later

IP

IP is a best-effort, connectionless, unreliable, packet based protocol

IP

IP is a best-effort, connectionless, unreliable, packet based protocol

Recall: “unreliable” means “not guaranteed reliable”

IP

IP is a best-effort, connectionless, unreliable, packet based protocol

Recall: “unreliable” means “not guaranteed reliable”

“best-effort”: no guarantee of delivery, or of any special features, like quality of service

IP

IP is a best-effort, connectionless, unreliable, packet based protocol

Recall: “unreliable” means “not guaranteed reliable”

“best-effort”: no guarantee of delivery, or of any special features, like quality of service

“connectionless”: each packet is independent of all others, there is no relationship between individual packets (in this layer)

IP

IP's primary purpose is getting packets from source to destination

IP

IP's primary purpose is getting packets from source to destination

It doesn't rely on any particular property of a link layer, so it can run on top of almost any link layer, even unreliable ones

IP

IP's primary purpose is getting packets from source to destination

It doesn't rely on any particular property of a link layer, so it can run on top of almost any link layer, even unreliable ones

But, first, some vocabulary

IP

Recall that IP is a cooperative system: for a packet to get from source to destination it is handed from one network to the next, hop by hop

IP

Recall that IP is a cooperative system: for a packet to get from source to destination it is handed from one network to the next, hop by hop

The nodes in the network have various roles:

IP

Recall that IP is a cooperative system: for a packet to get from source to destination it is handed from one network to the next, hop by hop

The nodes in the network have various roles:

- Host. A machine you actually use to do some work

IP

Recall that IP is a cooperative system: for a packet to get from source to destination it is handed from one network to the next, hop by hop

The nodes in the network have various roles:

- Host. A machine you actually use to do some work
- Bridge. Connects two physical networks together

IP

Recall that IP is a cooperative system: for a packet to get from source to destination it is handed from one network to the next, hop by hop

The nodes in the network have various roles:

- Host. A machine you actually use to do some work
- Bridge. Connects two physical networks together
- Gateway. Provides a connection off the local network

IP

Recall that IP is a cooperative system: for a packet to get from source to destination it is handed from one network to the next, hop by hop

The nodes in the network have various roles:

- Host. A machine you actually use to do some work
- Bridge. Connects two physical networks together
- Gateway. Provides a connection off the local network
- Router. A machine joining two or more networks and whose primary function is to determine where a packet goes next (i.e., routing)

IP

Recall that IP is a cooperative system: for a packet to get from source to destination it is handed from one network to the next, hop by hop

The nodes in the network have various roles:

- Host. A machine you actually use to do some work
- Bridge. Connects two physical networks together
- Gateway. Provides a connection off the local network
- Router. A machine joining two or more networks and whose primary function is to determine where a packet goes next (i.e., routing)

These are not mutually exclusive: gateways and routers can be hosts; gateways do trivial routing

IP

Marketing alert: things you see described as “routers” in the shops are unlikely to be actual routers, which are specialist bits of equipment

IP

Marketing alert: things you see described as “routers” in the shops are unlikely to be actual routers, which are specialist bits of equipment

To them, the word “router” seems to means “a box you plug into the network”

IP

Marketing alert: things you see described as “routers” in the shops are unlikely to be actual routers, which are specialist bits of equipment

To them, the word “router” seems to means “a box you plug into the network”

It should really mean “a box that engages in routing protocols”

IP

The basic idea is that a packet does not know how to get from source to destination: this is the routers' job (and it can be quite complex: see later)

IP

The basic idea is that a packet does not know how to get from source to destination: this is the routers’ job (and it can be quite complex: see later)

The IP layer takes bytes from the transport layer and prepends a header to support routing (and other things), producing packets often called *datagrams* in this layer

IP

The basic idea is that a packet does not know how to get from source to destination: this is the routers' job (and it can be quite complex: see later)

The IP layer takes bytes from the transport layer and prepends a header to support routing (and other things), producing packets often called *datagrams* in this layer

The IP specification says datagrams can be up to 64KB in size, but they are usually in the region of 1500 bytes (Ethernet, again), but can be much larger (e.g., 9000 bytes) in specialist networks

IP

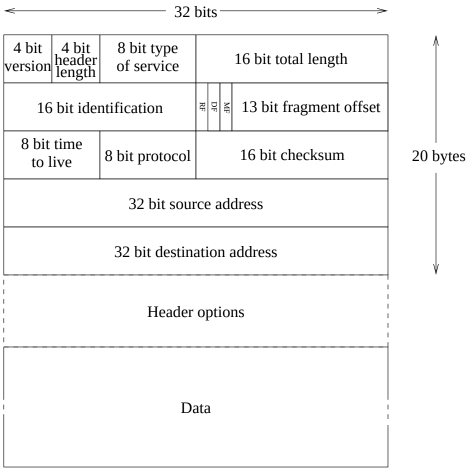
The basic idea is that a packet does not know how to get from source to destination: this is the routers' job (and it can be quite complex: see later)

The IP layer takes bytes from the transport layer and prepends a header to support routing (and other things), producing packets often called *datagrams* in this layer

The IP specification says datagrams can be up to 64KB in size, but they are usually in the region of 1500 bytes (Ethernet, again), but can be much larger (e.g., 9000 bytes) in specialist networks

We return to the IP header

IP



IPv4 datagram header

IP

- Version. Four bit field containing the value 4. A later version of IP (IPv6) contains 6

IP

- Version. Four bit field containing the value 4. A later version of IP (IPv6) contains 6
- Header length. There are some optional fields, so the header can vary in size, so this is needed to pinpoint the end of the header. Given as a number of 4 byte words. Four bits, maximum value 15, so maximum header length of 60 bytes

IP

- Version. Four bit field containing the value 4. A later version of IP (IPv6) contains 6
- Header length. There are some optional fields, so the header can vary in size, so this is needed to pinpoint the end of the header. Given as a number of 4 byte words. Four bits, maximum value 15, so maximum header length of 60 bytes
- Type of service. Eight bits. To indicate to a router how this datagram should be treated in terms of cost, speed and reliability (if possible)

IP

- Version. Four bit field containing the value 4. A later version of IP (IPv6) contains 6
- Header length. There are some optional fields, so the header can vary in size, so this is needed to pinpoint the end of the header. Given as a number of 4 byte words. Four bits, maximum value 15, so maximum header length of 60 bytes
- Type of service. Eight bits. To indicate to a router how this datagram should be treated in terms of cost, speed and reliability (if possible)

E.g., for audio it is better to get data through quickly rather than 100% reliably as the human ear is more sensitive to gaps than occasional errors

IP

TOS/DS

The TOS field, these days called the *Differentiated Services Field* (DS field), is to inform routers on the best way to treat this datagram

IP

TOS/DS

The TOS field, these days called the *Differentiated Services Field* (DS field), is to inform routers on the best way to treat this datagram

This allows the implementation of *Quality of Service* (QoS)

IP

TOS/DS

The TOS field, these days called the *Differentiated Services Field* (DS field), is to inform routers on the best way to treat this datagram

This allows the implementation of *Quality of Service* (QoS)

The full range of options available is complex (see RFC2474 for details), but can indicate things like

IP

TOS/DS

The TOS field, these days called the *Differentiated Services Field* (DS field), is to inform routers on the best way to treat this datagram

This allows the implementation of *Quality of Service* (QoS)

The full range of options available is complex (see RFC2474 for details), but can indicate things like

- Minimise delay. Do not hold onto this datagram longer than necessary, and perhaps prioritise it over others

IP

TOS/DS

- Maximise throughput. Not quite the same as minimising delay, since collecting together several small datagrams and sending them off together may be more bandwidth efficient

IP

TOS/DS

- Maximise throughput. Not quite the same as minimising delay, since collecting together several small datagrams and sending them off together may be more bandwidth efficient
- Maximise reliability. Try not to drop this datagram if the router is becoming overloaded; drop another datagram first

IP

TOS/DS

- Maximise throughput. Not quite the same as minimising delay, since collecting together several small datagrams and sending them off together may be more bandwidth efficient
- Maximise reliability. Try not to drop this datagram if the router is becoming overloaded; drop another datagram first
- Minimise cost. For this datagram cost is more important than reliability or speed. This datagram can be delayed if it makes transmission cheaper

IP

TOS/DS

Early routers ignored the TOS field, but these days QoS is very important

IP

TOS/DS

Early routers ignored the TOS field, but these days QoS is very important

Modern routers do (or should) pay attention to the DS field

IP

TOS/DS

Early routers ignored the TOS field, but these days QoS is very important

Modern routers do (or should) pay attention to the DS field

Here, as in some other parts of the IP specification, a router may ignore some information if it wishes. It might be the software is so old it does not recognise a modern field; or it might simply be unable to make use of the information. You are strongly recommended to act on the information, though

IP

TOS/DS

Early routers ignored the TOS field, but these days QoS is very important

Modern routers do (or should) pay attention to the DS field

Here, as in some other parts of the IP specification, a router may ignore some information if it wishes. It might be the software is so old it does not recognise a modern field; or it might simply be unable to make use of the information. You are strongly recommended to act on the information, though

Exercise Look up the problems *Explicit Congestion Notification* (ECN) had when it was introduced

IP

- Total Length. Of the entire datagram, including header, in bytes. 16 bits, so giving a maximum size of 65535 bytes. Much larger than domestic networks need, but too small for high-speed networks

IP

- Total Length. Of the entire datagram, including header, in bytes. 16 bits, so giving a maximum size of 65535 bytes. Much larger than domestic networks need, but too small for high-speed networks

As usual, larger packet sizes mean lower overheads:

- Time overhead in hosts of splitting data into datagrams, adding headers, then removing headers and reassembling
- Bandwidth overhead as each header is 20 or more bytes that is not data; plus more gaps between datagrams
- Time overhead in routers of processing packets

IP

The Total Length field is essential over Ethernet as it pads short frames

IP

The Total Length field is essential over Ethernet as it pads short frames

Thus we get better independence from the MAC/PHY layers

IP

- Identification. 16 bits. A value that is unique to each datagram sent by the source, often incrementing by 1 for each successive datagram sent

IP

- Identification. 16 bits. A value that is unique to each datagram sent by the source, often incrementing by 1 for each successive datagram sent

Used in *fragmentation* to reassemble the fragments of a single datagram. All the fragments get their own IP header, but share the same identification

IP

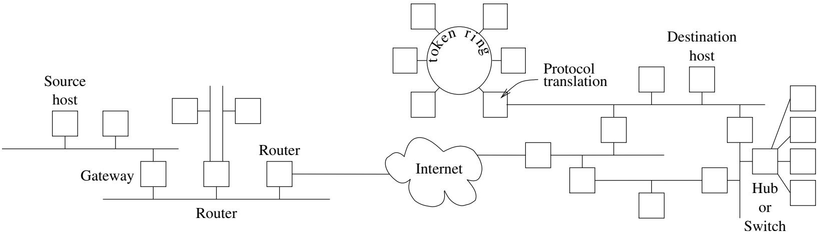
- Identification. 16 bits. A value that is unique to each datagram sent by the source, often incrementing by 1 for each successive datagram sent

Used in *fragmentation* to reassemble the fragments of a single datagram. All the fragments get their own IP header, but share the same identification

So we need to discuss fragmentation

IP

Fragmentation



Many kinds of hardware in the Internet

The path a packet takes from source to destination will typically go through a wide variety of differing kinds of hardware

IP

Fragmentation

For example, your home Ethernet/Wi-Fi might support 1500 byte packets, while your VDSL link might have a maximum of 1480 bytes

IP

Fragmentation

For example, your home Ethernet/Wi-Fi might support 1500 byte packets, while your VDSL link might have a maximum of 1480 bytes

Thus IP must face the problem of differing link layer properties, in particular maximum frame size

IP

Fragmentation

If a big datagram hits a part of the Internet that can only cope with small datagrams, there is a problem

IP

Fragmentation

If a big datagram hits a part of the Internet that can only cope with small datagrams, there is a problem

IPv4 deals with this by *fragmentation*: a datagram can be subdivided by a router into several smaller datagrams; it is the destination's problem to glue them back together

IP

Fragmentation

If a big datagram hits a part of the Internet that can only cope with small datagrams, there is a problem

IPv4 deals with this by *fragmentation*: a datagram can be subdivided by a router into several smaller datagrams; it is the destination's problem to glue them back together

In the right order

IP

Fragmentation

If a big datagram hits a part of the Internet that can only cope with small datagrams, there is a problem

IPv4 deals with this by *fragmentation*: a datagram can be subdivided by a router into several smaller datagrams; it is the destination's problem to glue them back together

In the right order

The fragmentation fields in the IP header deal with this

IP

Fragmentation

- Flags. Three bits: two used and one reserved

IP

Fragmentation

- Flags. Three bits: two used and one reserved

1. RF. Reserved for later use, must be 0 (see RFC3514 for a suggested use)

IP

Fragmentation

- Flags. Three bits: two used and one reserved

1. RF. Reserved for later use, must be 0 (see RFC3514 for a suggested use)
2. DF. Don't fragment. If a host can't (or doesn't want to) deal with fragments this bit is set to inform the routers on the path to the destination. A router might choose an alternative non-fragmenting route, or simply drop the datagram and send an error message back to the source which can then send smaller datagrams.
   All hosts are required to be able to accept datagrams of 576 bytes

IP

Fragmentation

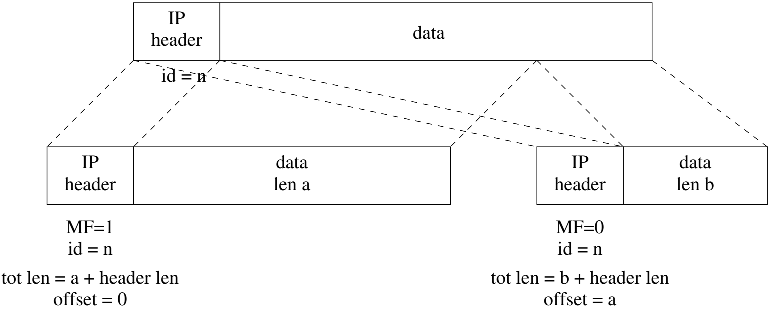
- Flags. Three bits: two used and one reserved

1. RF. Reserved for later use, must be 0 (see RFC3514 for a suggested use)
2. DF. Don't fragment. If a host can't (or doesn't want to) deal with fragments this bit is set to inform the routers on the path to the destination. A router might choose an alternative non-fragmenting route, or simply drop the datagram and send an error message back to the source which can then send smaller datagrams.
   All hosts are required to be able to accept datagrams of 576 bytes
3. MF. More fragments. All fragments except the end fragment have this set

IP

Fragmentation

Fragment Offset. Where this fragment came from in the original datagram



Fragmenting an IP datagram

IP

- Fragment Offset. 13 bits, giving the byte offset divided by 8. E.g., value of 20 means an offset of 160 bytes

IP

- Fragment Offset. 13 bits, giving the byte offset divided by 8. E.g., value of 20 means an offset of 160 bytes

So 13 bits is enough to cover the 16 bit range of sizes

IP

- Fragment Offset. 13 bits, giving the byte offset divided by 8. E.g., value of 20 means an offset of 160 bytes

So 13 bits is enough to cover the 16 bit range of sizes

And every fragment (apart from the end fragment) must be a multiple of 8 bytes long: the router doing the fragmentation must ensure this

IP

- Fragment Offset. 13 bits, giving the byte offset divided by 8. E.g., value of 20 means an offset of 160 bytes

So 13 bits is enough to cover the 16 bit range of sizes

And every fragment (apart from the end fragment) must be a multiple of 8 bytes long: the router doing the fragmentation must ensure this

Note that a datagram may be split into any number of smaller fragments, not just two

IP

Fragmentation

Every fragment has a copy of the original IP header, but with the various fragmentation and length fields set appropriately

IP

Fragmentation

Every fragment has a copy of the original IP header, but with the various fragmentation and length fields set appropriately

In more detail: each fragment header will be a copy of the original header apart from

- total length: set to the fragment size
- MF: set to 1 if this is not the end fragment
- fragment offset: set appropriately
- (TTL and checksum: set appropriately)

In particular, all the fragments of the original datagram have the same the identification field value

IP

Fragmentation

In particular, all the fragments of the original datagram have the same the identification field value

When the fragment with MF $= 0$ is received, its fragment offset and length will give the length of the original datagram

IP

Fragmentation

In particular, all the fragments of the original datagram have the same the identification field value

When the fragment with MF $= 0$ is received, its fragment offset and length will give the length of the original datagram

The destination can then reassemble the original datagram when all the fragments have arrived

IP

Fragmentation

In particular, all the fragments of the original datagram have the same the identification field value

When the fragment with MF $= 0$ is received, its fragment offset and length will give the length of the original datagram

The destination can then reassemble the original datagram when all the fragments have arrived

Fragments are IP datagrams, so as always, they can arrive in any order; or not at all

IP

Fragmentation

IPv4 spends a lot of effort coping with fragmentation

IP

Fragmentation

IPv4 spends a lot of effort coping with fragmentation

It is costly and should be avoided

IP

Fragmentation

IPv4 spends a lot of effort coping with fragmentation

It is costly and should be avoided

- Performing fragmentation in a router takes time

IP

Fragmentation

IPv4 spends a lot of effort coping with fragmentation

It is costly and should be avoided

- Performing fragmentation in a router takes time
- More overhead as more datagrams for a given amount of data

IP

Fragmentation

IPv4 spends a lot of effort coping with fragmentation

It is costly and should be avoided

- Performing fragmentation in a router takes time
- More overhead as more datagrams for a given amount of data
- More overhead as more datagrams are traversing the network

IP

Fragmentation

IPv4 spends a lot of effort coping with fragmentation

It is costly and should be avoided

- Performing fragmentation in a router takes time
- More overhead as more datagrams for a given amount of data
- More overhead as more datagrams are traversing the network
- More datagrams means a greater probability one will be lost or corrupted

IP

Fragmentation

- If a fragment is lost, the *entire* original datagram must be retransmitted: there is no mechanism in IP to indicate which fragment was lost

IP

Fragmentation

- If a fragment is lost, the *entire* original datagram must be retransmitted: there is no mechanism in IP to indicate which fragment was lost
- Fragments are datagrams in their own right and can themselves be fragmented

IP

Fragmentation

- If a fragment is lost, the *entire* original datagram must be retransmitted: there is no mechanism in IP to indicate which fragment was lost
- Fragments are datagrams in their own right and can themselves be fragmented

Fragment processing software (particularly reassembly) has a history of buggy implementations leading to hacked machines

IP

Fragmentation

- If a fragment is lost, the *entire* original datagram must be retransmitted: there is no mechanism in IP to indicate which fragment was lost
- Fragments are datagrams in their own right and can themselves be fragmented

Fragment processing software (particularly reassembly) has a history of buggy implementations leading to hacked machines

Exercise Consider what happens when a fragment is further fragmented. Differentiate the cases of $\mathrm{MF} = 0$ and $\mathrm{MF} = 1$

IP

Fragmentation

Fragmentation is such a costly process that modern implementations try very hard to avoid it

They employ *MTU Discovery*

IP

Fragmentation

Setting DF in the header prohibits fragmentation; if a router cannot avoid fragmenting it drops the datagram and returns a “fragmentation needed but DF set” error message back. The sender can then send smaller datagrams

IP

Fragmentation

Setting DF in the header prohibits fragmentation; if a router cannot avoid fragmenting it drops the datagram and returns a “fragmentation needed but DF set” error message back. The sender can then send smaller datagrams

This allows *MTU Discovery*. The *Maximum Transmission Unit* (MTU) is the largest datagram a host or network can transmit

IP

Fragmentation

Setting DF in the header prohibits fragmentation; if a router cannot avoid fragmenting it drops the datagram and returns a “fragmentation needed but DF set” error message back. The sender can then send smaller datagrams

This allows *MTU Discovery*. The *Maximum Transmission Unit* (MTU) is the largest datagram a host or network can transmit

The *path* MTU is the smallest MTU for the entire path from source to destination

IP

Fragmentation

Setting DF in the header prohibits fragmentation; if a router cannot avoid fragmenting it drops the datagram and returns a "fragmentation needed but DF set" error message back. The sender can then send smaller datagrams

This allows *MTU Discovery*. The *Maximum Transmission Unit* (MTU) is the largest datagram a host or network can transmit

The *path* MTU is the smallest MTU for the entire path from source to destination

A datagram not larger than the path MTU will not get fragmented

IP

Fragmentation

Setting DF in the header prohibits fragmentation; if a router cannot avoid fragmenting it drops the datagram and returns a "fragmentation needed but DF set" error message back. The sender can then send smaller datagrams

This allows *MTU Discovery*. The *Maximum Transmission Unit* (MTU) is the largest datagram a host or network can transmit

The *path* MTU is the smallest MTU for the entire path from source to destination

A datagram not larger than the path MTU will not get fragmented

MTU Discovery works by sending variously sized datagrams with DF set, and monitors the errors returned

IP

Fragmentation

When a datagram reaches the destination with no fragmentation error we have found a lower bound for the path MTU

IP

Fragmentation

When a datagram reaches the destination with no fragmentation error we have found a lower bound for the path MTU

This bound is approximate as the network is dynamic and paths may change!

IP

Fragmentation

When a datagram reaches the destination with no fragmentation error we have found a lower bound for the path MTU

This bound is approximate as the network is dynamic and paths may change!

This is the approach IPv6 adopts: don't have fragmentation in routers, but require MTU discovery (see later)

IP

Fragmentation

When a datagram reaches the destination with no fragmentation error we have found a lower bound for the path MTU

This bound is approximate as the network is dynamic and paths may change!

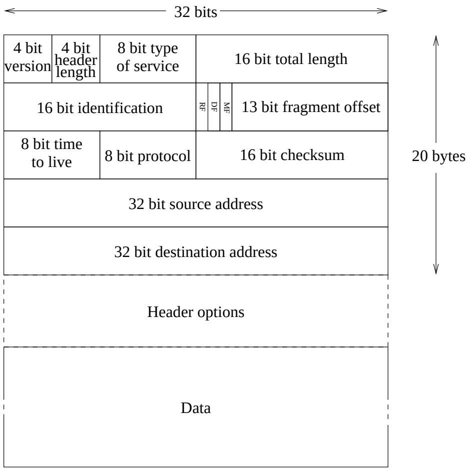
This is the approach IPv6 adopts: don't have fragmentation in routers, but require MTU discovery (see later)

Note: a host is not required to implement MTU Discovery in IPv4: it's just good if it does as fragmentation is such a large overhead

IP

Back to the IPv4 header fields

IP



IPv4 datagram header

IP

- Time To Live. An eight bit counter used to limit the lifetime of a datagram

IP

- Time To Live. An eight bit counter used to limit the lifetime of a datagram

Poorly configured routers might bounce datagrams back and forth or in circles indefinitely, thus clogging the network with lost datagrams

IP

- Time To Live. An eight bit counter used to limit the lifetime of a datagram

Poorly configured routers might bounce datagrams back and forth or in circles indefinitely, thus clogging the network with lost datagrams

The TTL starts at 64, or 32, say, and is reduced by one as it passes through each router

IP

If a TTL ever reaches 0 at a router, that datagram is discarded, and an error message ("time exceeded in-transit") is sent back to the source of the datagram

IP

If a TTL ever reaches 0 at a router, that datagram is discarded, and an error message (“time exceeded in-transit”) is sent back to the source of the datagram

This limits errant datagrams: eventually the TTL must reach 0 and the datagram is dropped

IP

If a TTL ever reaches 0 at a router, that datagram is discarded, and an error message (“time exceeded in-transit”) is sent back to the source of the datagram

This limits errant datagrams: eventually the TTL must reach 0 and the datagram is dropped

Eight bits means a maximum path of length 255, but this seems enough for the current Internet: no valid paths as long as this are known

IP

If a TTL ever reaches 0 at a router, that datagram is discarded, and an error message (“time exceeded in-transit”) is sent back to the source of the datagram

This limits errant datagrams: eventually the TTL must reach 0 and the datagram is dropped

Eight bits means a maximum path of length 255, but this seems enough for the current Internet: no valid paths as long as this are known

The *width* of the Internet is the length of the longest path: this is uncertain and constantly changing but definitely over 32

IP

Originally the TTL was to be a measure of *time*, reducing by one for each second in a router. In practice no implementations did this, but just decremented by one regardless. This is now the expected behaviour

IP

Originally the TTL was to be a measure of *time*, reducing by one for each second in a router. In practice no implementations did this, but just decremented by one regardless. This is now the expected behaviour

Again: this is IP being pragmatic, following what people actually do in implementations, rather than the letter of the specification

IP

Originally the TTL was to be a measure of *time*, reducing by one for each second in a router. In practice no implementations did this, but just decremented by one regardless. This is now the expected behaviour

Again: this is IP being pragmatic, following what people actually do in implementations, rather than the letter of the specification

Exercise Why doesn't everyone simply put 255 into the TTL field?

IP

- Protocol. This eight bit field connects the IP layer to the transport layer. This is a value indicating which transport layer software to pass the contents of the datagram to. For example, UDP is 17 and TCP is 6

IP

- Protocol. This eight bit field connects the IP layer to the transport layer. This is a value indicating which transport layer software to pass the contents of the datagram to. For example, UDP is 17 and TCP is 6
- Header checksum. As for the Ethernet header, this is a simple function of the bytes in the IP header. If the checksum is bad, the datagram is silently dropped. A higher layer must detect this and perform whatever action it needs. Recall that the IP layer is not guaranteed reliable

IP

The checksum is checked in each router, as well as the final destination

IP

The checksum is checked in each router, as well as the final destination

This ensures corrupted datagrams are dropped as soon as possible

IP

The checksum is checked in each router, as well as the final destination

This ensures corrupted datagrams are dropped as soon as possible

Note the checksum includes the TTL field so it must be recomputed and rewritten in the datagram by each router the datagram passes through (this increases the time a router must spend on each datagram)

IP

- Source and Destination Address. 32 bit numbers that uniquely determine the source and destination machines on the Internet

IP

- Source and Destination Address. 32 bit numbers that uniquely determine the source and destination machines on the Internet

"Uniquely" is now not true

IP

- Source and Destination Address. 32 bit numbers that uniquely determine the source and destination machines on the Internet

“Uniquely” is now not true

32 bits limits us to at most 4,294,967,296 hosts on the Internet

IP

- Source and Destination Address. 32 bit numbers that uniquely determine the source and destination machines on the Internet

“Uniquely” is now not true

32 bits limits us to at most 4,294,967,296 hosts on the Internet

Not enough, and a significant chunk of those addresses are reserved for special purposes and can’t be used for host addresses

IP

- Source and Destination Address. 32 bit numbers that uniquely determine the source and destination machines on the Internet

“Uniquely” is now not true

32 bits limits us to at most 4,294,967,296 hosts on the Internet

Not enough, and a significant chunk of those addresses are reserved for special purposes and can’t be used for host addresses

We will come back to this very important problem later