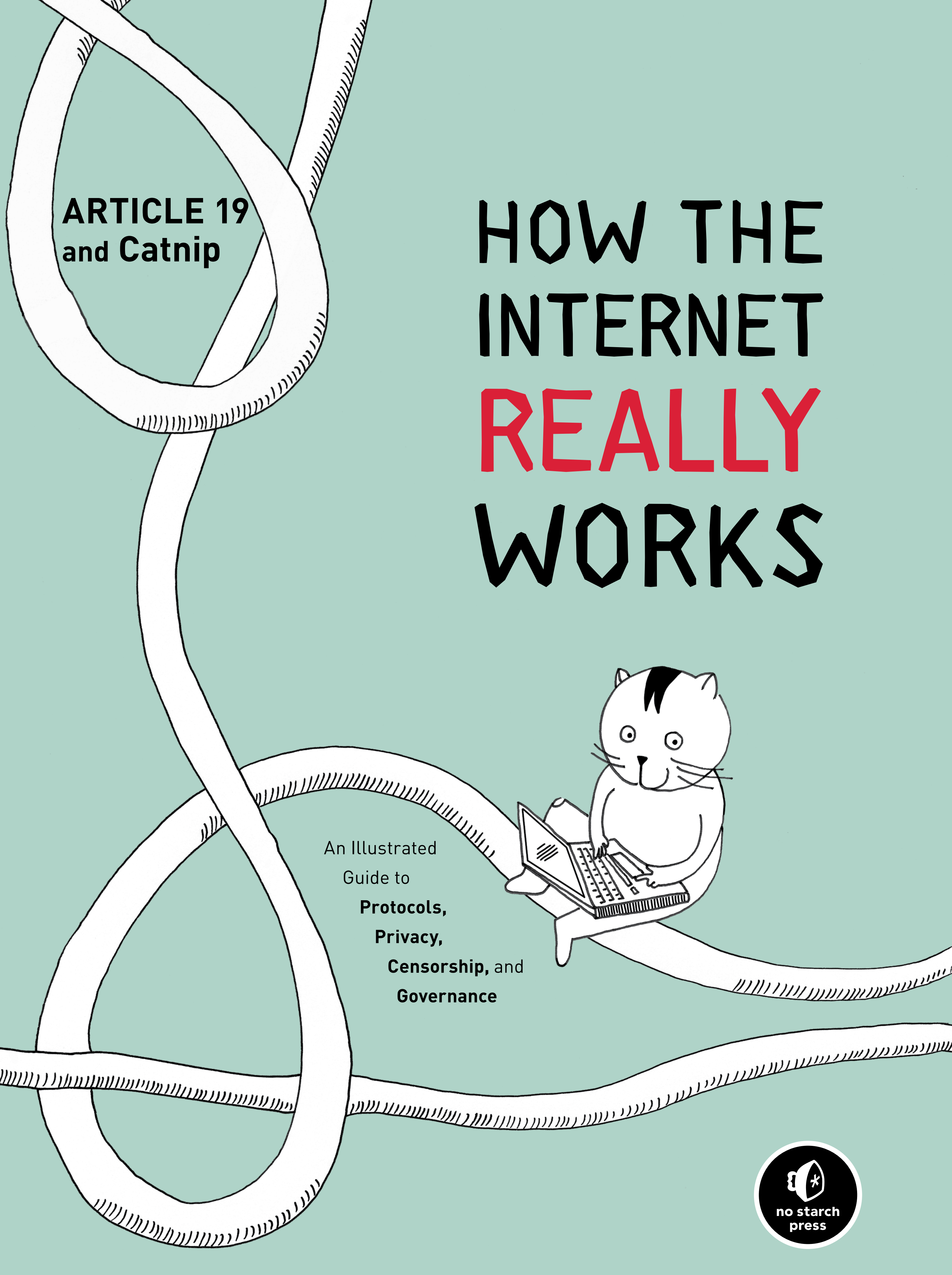


ARTICLE 19
and Catnip

HOW THE INTERNET REALLY WORKS

An Illustrated Guide to **Protocols, Privacy, Censorship,** and **Governance**

no starch press

HOW THE INTERNET REALLY WORKS

AN ILLUSTRATED GUIDE TO PROTOCOLS, PRIVACY, CENSORSHIP, AND GOVERNANCE

ARTICLE 19

Contributors

Ulrike Uhlig
Mallory Knodel
Niels ten Oever
Corinne Cath
Catnip

No Starch Press
San Francisco

ISBN (print): 978-1-7185-0029-7
ISBN (ebook): 978-1-7185-0030-3

Publisher: William Pollock
Executive Editor: Barbara Yien
Production Editor: Michele Mangelli
Cover and Interior Design: Ulrike Uhlig
Technical Reviewer: Eric Lawrence
Copyeditor: Sally Peyrefitte
Compositor: Hespenheide Design
Proofreader: Betsy Dietrich

For information on book distributors or translations, please contact No Starch Press, Inc. directly:
No Starch Press, Inc.
245 8th Street, San Francisco, CA 94103
phone: 1-415-863-9900; info@nostarch.com
www.nostarch.com

Library of Congress Cataloging-in-Publication Data
Names: Article 19, author. | Uhlig, Ulrike, contributor. | Knodel, Mallory, contributor. | Oever, Niels ten, contributor. | Cath-Speth, Corinne, contributor.
Title: How the internet really works : an illustrated guide to protocols, privacy, censorship, and governance / Article 19 ; contributors: Ulrike Uhlig, Mallory Knodel, Niels ten Oever, Corinne Cath-Speth.
Description: San Francisco : No Starch Press, 2020. | Includes bibliographical references and index. | Summary: "A comic-based introduction to the technical side of the internet, including transport protocols and basic internet infrastructure. Also explains broader concepts such as security and privacy in the context of the internet"-- Provided by publisher.
Identifiers: LCCN 2020021943 (print) | LCCN 2020021944 (ebook) | ISBN 9781718500297 | ISBN 9781718500303 (ebook)
Subjects: LCSH: Internet.
Classification: LCC TK5105.875.I57 A7835 2020 (print) | LCC TK5105.875.I57 (ebook) | DDC 004.67/8--dc23
LC record available at https://lccn.loc.gov/2020021943
LC ebook record available at https://lccn.loc.gov/2020021944

The information in this book is distributed on an "As Is" basis, without warranty. While every precaution has been taken in the preparation of this work, neither the authors nor No Starch Press, Inc. shall have any liability to any person or entity with respect to any loss or damage caused or alleged to be caused directly or indirectly by the information contained in it.

To Niels, Amelia, Mehwish, Vidushi, and Corinne.
And to Catnip, our mascot without a team.

ARTICLE 19 is an international nonprofit organization that seeks to promote, develop, and protect freedom of expression, including access to information. Headquartered in London, with offices in Bangladesh, Brazil, Kenya, Mexico, Senegal, Tunisia, Myanmar, and the USA, ARTICLE 19 works to bridge the knowledge gap about internet infrastructure and why it matters for people.

About the Contributors

Originally a professional front-end web developer, Ulrike Uhlig is also a Debian Developer and works on software related to privacy and anonymity online. Ulrike works with projects of the internet freedom community and nonprofit organizations at the intersection of technology, arts, and human rights. Born in East Berlin, she spent 15 years in France, where she obtained an MA in visual and contemporary arts. In addition to coauthoring *How the Internet Really Works*, Ulrike created the illustrations featured in the book.

Mallory Knodel is the chief technology officer for the Center for Democracy and Technology, the cochair of the Human Rights and Protocol Considerations research group of the Internet Research Task Force, and a chairing advisor to the Freedom Online Coalition. Originally from the United States but living in Nairobi, she has worked as a technical expert with grassroots and nonprofit organizations around the world since 2008. She holds a BS in physics and mathematics and an MA in science education.

Niels Ten Oever is a PhD candidate with the DATACTIVE Research Group at the Media Studies and Political Science Department at the University of Amsterdam, and postdoctoral scholar (abd) with the Communications Department at Texas A&M University. His research focuses on how norms, such as human rights, get inscribed, resisted, and subverted in the internet infrastructure through transnational governance. Previously Niels worked as Head of Digital for ARTICLE19, where he designed, raised funds for, and set up the digital program. He holds a cum laude MA in philosophy from the University of Amsterdam.

Corinne Cath is a doctoral student at the Oxford Internet Institute. As a cultural anthropologist, she applies the tools of anthropology to the study of internet governance, in particular, the culture of the often opaque organizations that enable the technical functioning of the internet. Within that context, she focuses on the participation of human rights and civil liberties NGOs that are aiming to change computer code instead of legal code to effect social change. She is funded by the Ford Foundation and the Alan Turing Institute.

About the Technical Reviewer

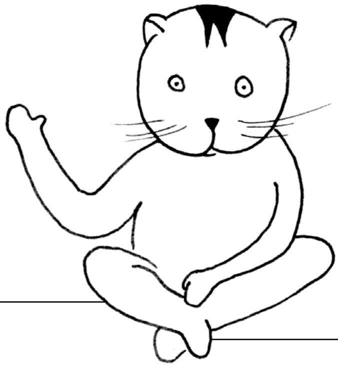
Eric Lawrence is best known as the developer of the Fiddler web debugging platform, used by web professionals worldwide. Currently a program manager for networking and privacy on the Edge browser team at Microsoft, Eric has developed web applications and browsers since 1999.

Brief Contents

Contents in Detail

Hi! I'm Catnip.

I've written a guide to how the internet works for my fellow cats—I mean people—who use technology, which is most of us. It's for those of us who are curious about basic internet infrastructure and how it operates. I've taken these basic concepts and presented them with concise, clear text alongside playful illustrations.



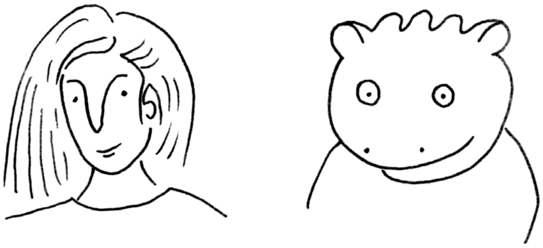
The guide answers questions like these:
- How does the internet work?
- What enables information to travel across the internet?
- What interferes with information traveling across the internet?
- What information is shared about users when they access the internet?
- What control do machines have?
- Who controls the internet?
- How is power over the decentralized internet distributed?

The chapters are broken down into questions about larger technological concepts such as transport protocols, security and privacy, algorithms, and internet infrastructure governance. The guide progresses from the most fundamental to the more complex concepts of how the internet works. So although each chapter is self-contained, reading from the beginning will ensure that your conceptual construction is built on foundational knowledge.

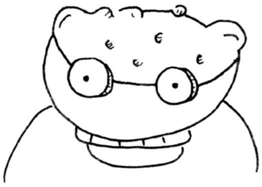
The guide ends with a chapter on how the user, as an important stakeholder, can shape the internet. It is critical to expand participation in internet governance and standards setting to currently underrepresented populations in technology. Furthermore, those who approach the development and use of technology in the public interest have a key role to play in ensuring that the internet, at a structural level, becomes a medium that enables social justice and human rights.

The information contained in this guide is best seen as supplementary to a practice, such as in a workshop or course, or for internet policy practitioners whose work can be strengthened with technical backing, for just two examples. You can reach out to us with your interest or questions on all of the above on *https://catnip.article19.org*!

Before you read further, I'd like to introduce you to my friends. They will accompany us throughout the book.



Meet Alice and Dragon. We work together and need to discuss all sorts of things every day. In our free time, we like to stay in touch. Sometimes we meet for a hike or for an evening out.

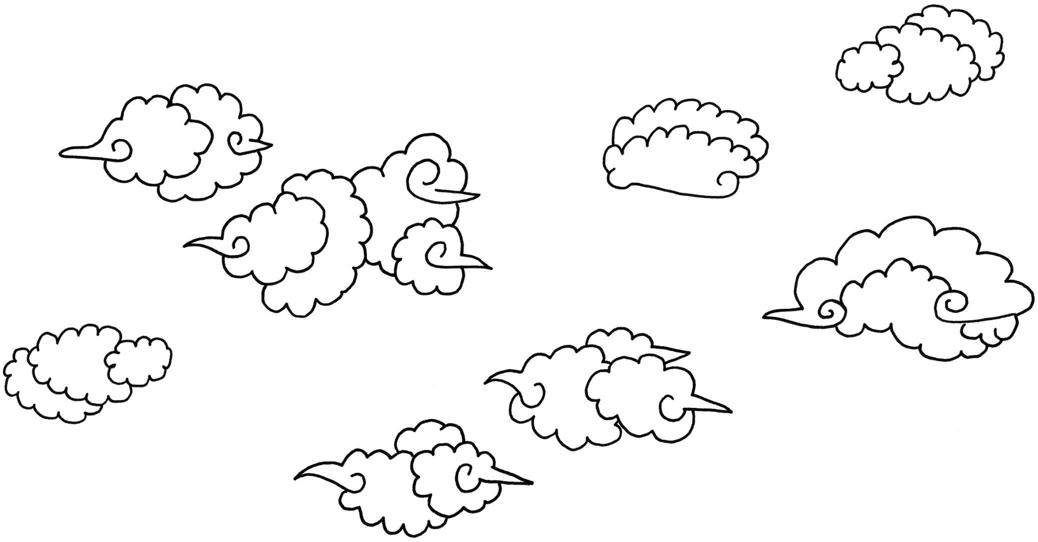


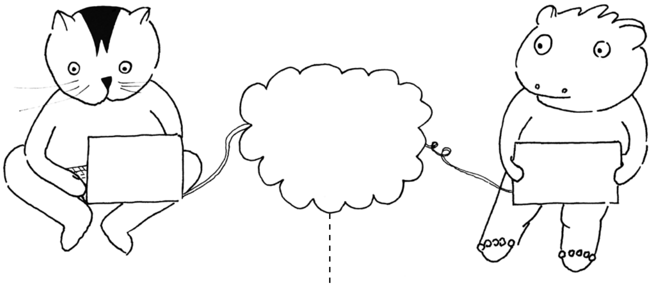
This is Eve, the notorious eavesdropper, who likes to listen into what everyone else is saying all the time.

This is the malicious Mallory, who likes to play practical jokes and makes it hard for people to communicate with each other privately.

1

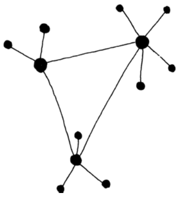
HOW IS THE INTERNET NETWORKED?

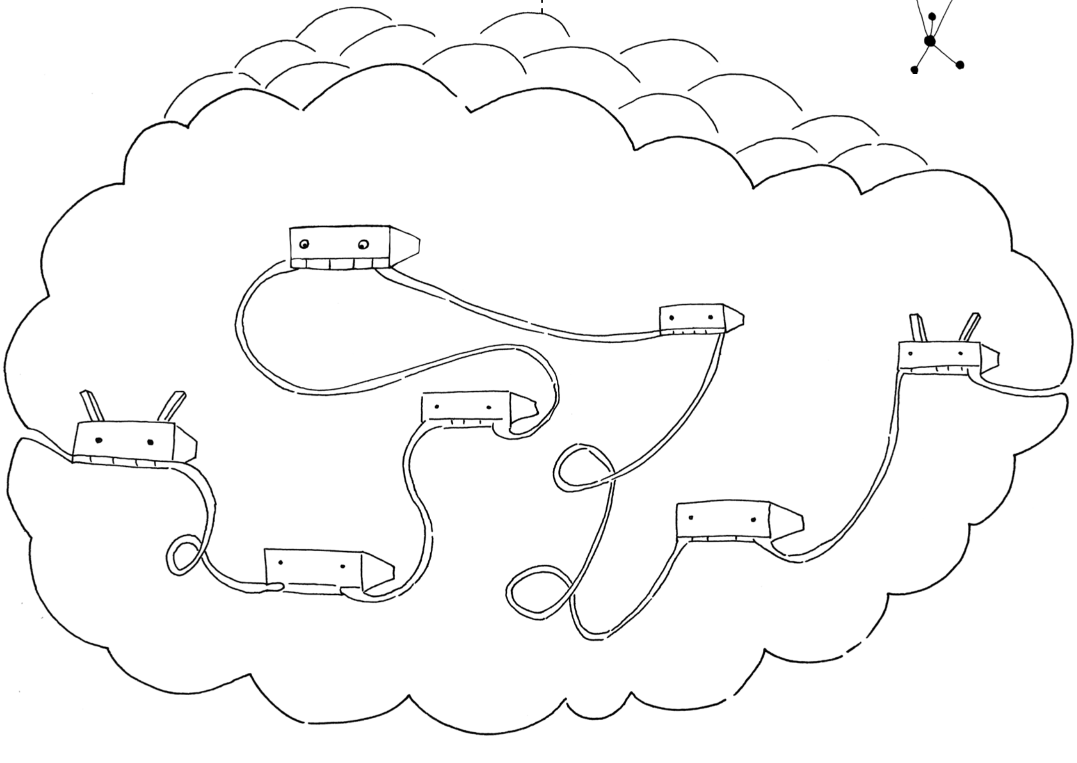




The internet is sometimes represented as a cloud made up of connections between devices. But this can be misleading, because there are few direct connections over the internet.

The internet is not a fully distributed network. In reality, the internet is decentralized with many centers or nodes, and direct or indirect connections between them.

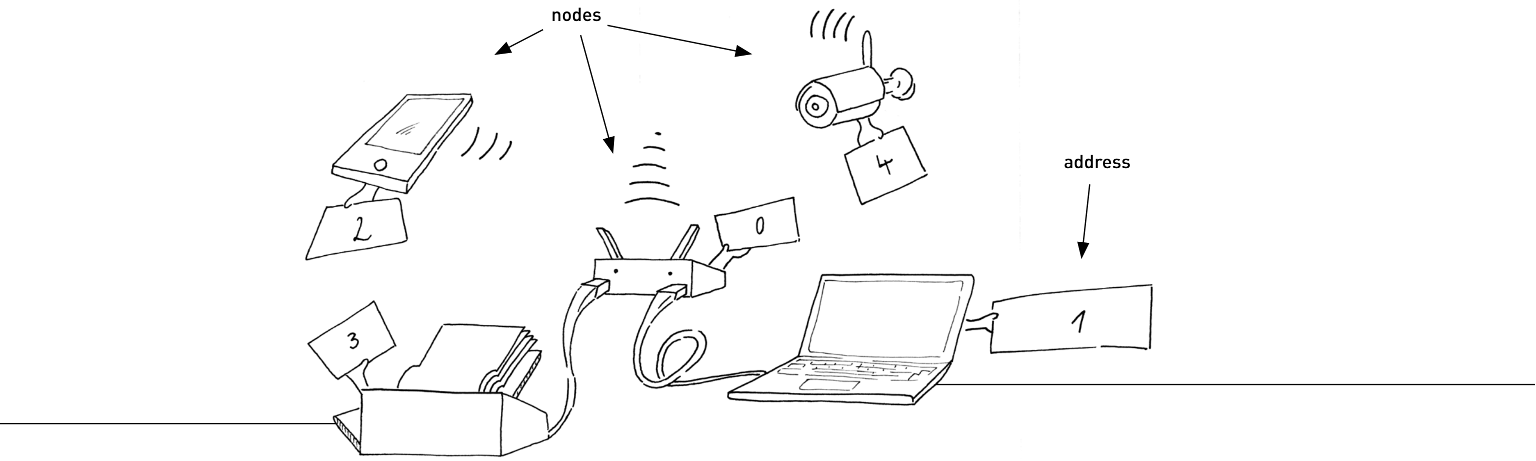




In this chapter, you'll learn how two devices can communicate with each other over networks via nodes, like routers, and how these connections between different networks make up the internet as we know it.

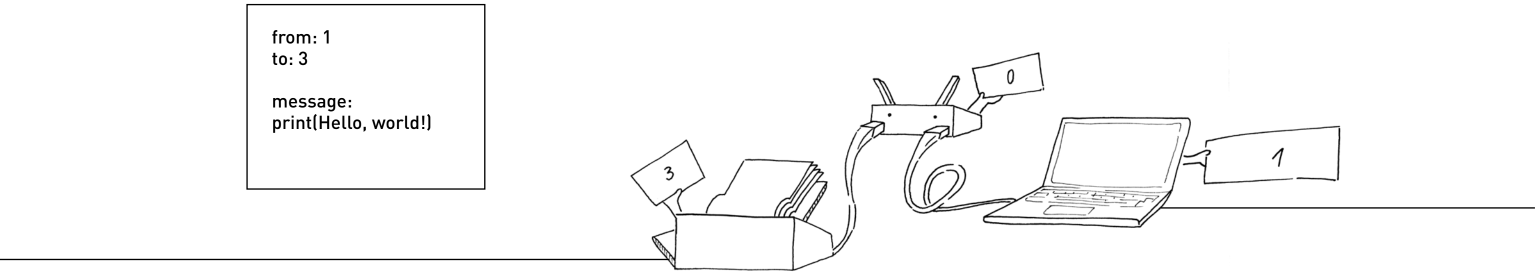
Nodes and Networks

Nodes are devices on networks that send or receive information.
For example, a node can be your laptop or a server that hosts websites.
On a network every node has an address, which is how nodes find one another.
Essentially any networked hardware with a network address is a node. On the internet a network address is an IP address.



Nodes can transfer messages to other nodes connected to the same network by providing the address of the destination node.

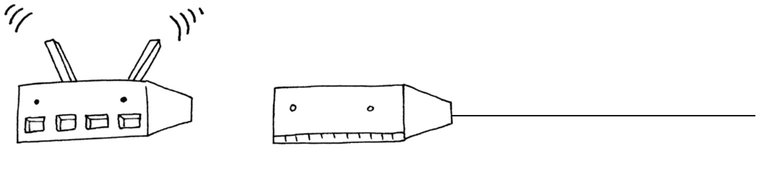
The address lets the network deliver the message to the right destination node, almost always passing it on through intermediate nodes.



from: 1
to: 3

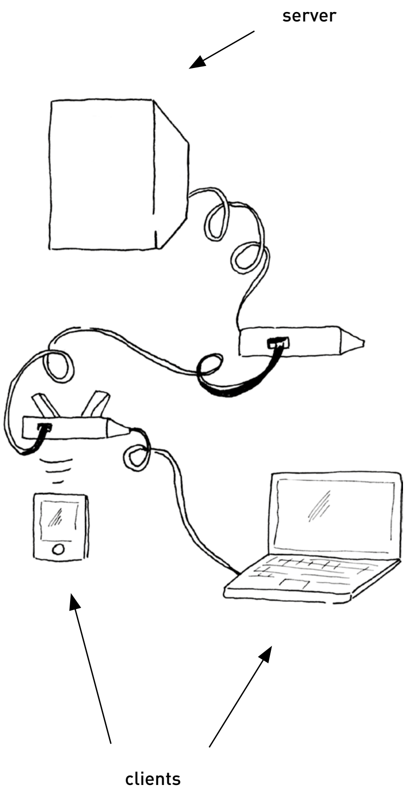
message:
print(Hello, world!)

We use devices called **routers** to connect different networks. Routers are devices that direct IP packets, the pieces of data that make up internet traffic, from one network to the other.



Servers and Clients

Nodes that provide services over a network are called **servers**. A server is a node that accepts connections from other nodes on a network and usually transmits information, receives information, or processes information as a service or application. Online gaming servers, website hosts, and email delivery services are examples of server nodes.

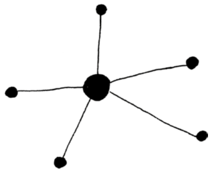


Network nodes that use a service are called **clients**. Clients can be actual user devices or client applications. For example, when we read our email we're connecting to our email server using a client application.

Network Types

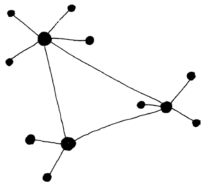
There are different types of networks that have distinct shapes when we map them out.

Centralized Network



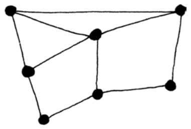
A network is **centralized** when many clients connect through a single router. Centralized services or local networks, such as game servers, are shaped like a star with a central point.

Decentralized Network



We call a network **decentralized** when many clients connect to many routers that connect to each other. The overall structure of today's internet is decentralized. A schematic map of services on a decentralized network, such as email delivery, is shaped like a constellation made up of many stars.

Distributed Network

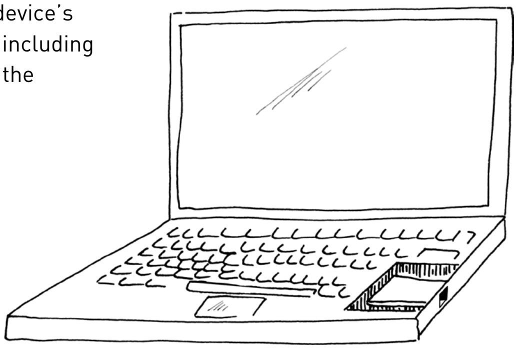


Sometimes clients can also be servers. On a **distributed** network all nodes are non-hierarchically connected to each other. A fully distributed network, where all nodes are equal and can speak directly to one another without central nodes, was once the desired utopia of the internet. The reality is that we're seeing increasing centralization of major internet services by a few dominant companies.

Hardware Addresses

The common hardware components of an electronic telecommunications device include the power source, sound and graphics cards, memory storage, processor chips, and various peripheral connections for cameras, headphones, and external drives. For the purposes of this guide to the internet, we'll focus only on the network card.

A **network card** handles a device's connection to the network, including providing the network with the device's identification.



Media Access Control Addresses (MAC)

Modern phones and computers have a tiny network card that allows them to connect to the internet. A node needs a network card to connect to the internet, just as a device needs an AM/FM receiver to tune in to the radio. An internet-enabled device's network card has a **MAC (Media Access Control) address**,[1] which is a unique, identifiable address that a device needs to communicate with another device, such as a home router. The router uses this address to identify the devices that are connected to it.

The MAC address is also called the **device ID**. Usually chosen by the manufacturer of the network card, it allows the software operating system running on the device to identify the card's exact model.

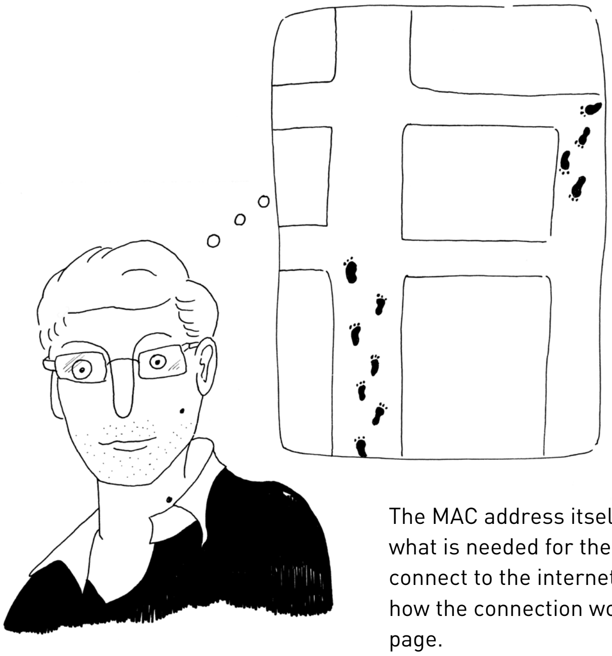
MAC addresses are useful only to local networks and shouldn't be required beyond this point; however, because they identify a device, and often its user, nonlocal network nodes can sometimes request and store MAC addresses.

Random MAC Addresses

According to Edward Snowden, the US National Security Agency (NSA) monitors MAC addresses of electronic devices to track the movements of everyone in a city.[2]

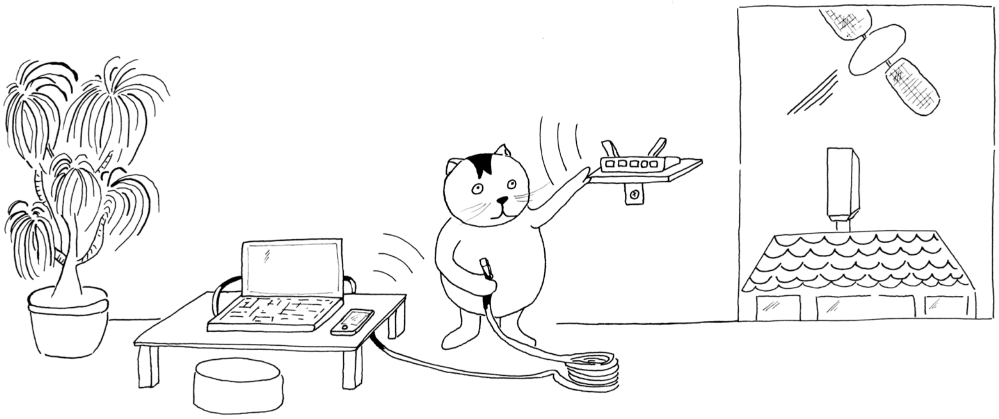
Free Wi-Fi hotspots can track users the same way.[3]

Some operating systems have started to randomize the MAC address to prevent **hardware addresses** from being too easily linked to real-world identities.



The MAC address itself is just part of what is needed for the computer to connect to the internet. We'll explain how the connection works on the next page.

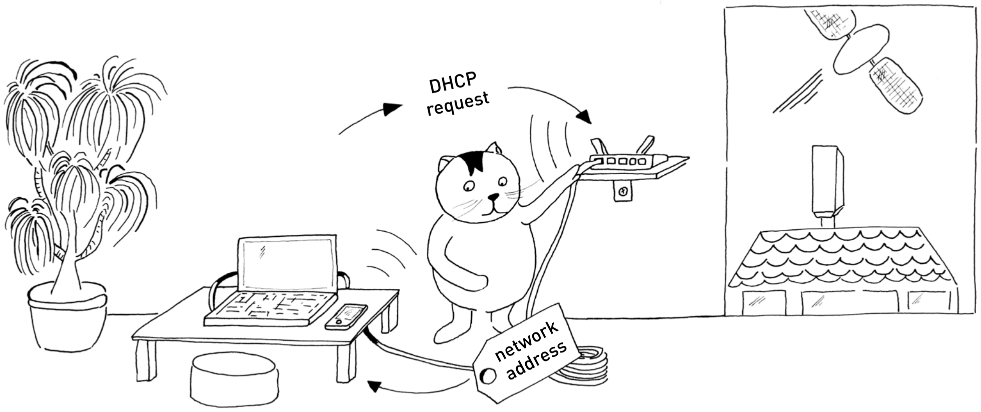
How a Device Becomes Part of a Network



When you connect your computer to the internet, you probably use an Ethernet cable or Wi-Fi to connect to your home router. Your mobile device might also use the Wi-Fi.

Alternatively you connect to the internet via your phone's mobile network or a satellite network.

Talking to the Router



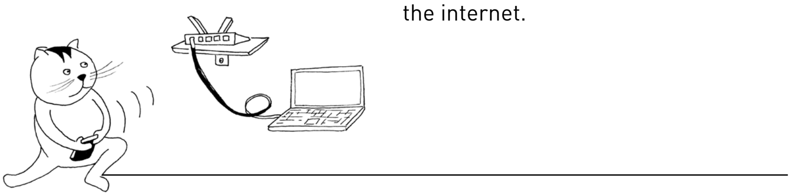
To communicate with other nodes on the network, in addition to the MAC address, you need a network address. To obtain a network address, your device needs to talk to the router.

Once physically connected through a cable, or after successfully choosing a Wi-Fi network and entering the password correctly, your device's network card gets assigned a network address through the **Dynamic Host Configuration Protocol (DHCP)**.[4]

If this request is successful, a router is simply set up to do two things. First, it hands out a network address to the device. Second, it tells your device to direct all of the data it wants to send to the network via the local network's standard gateway, which is often the router itself. The standard gateway is our device's entrance to the rest of the internet.

Getting Connected

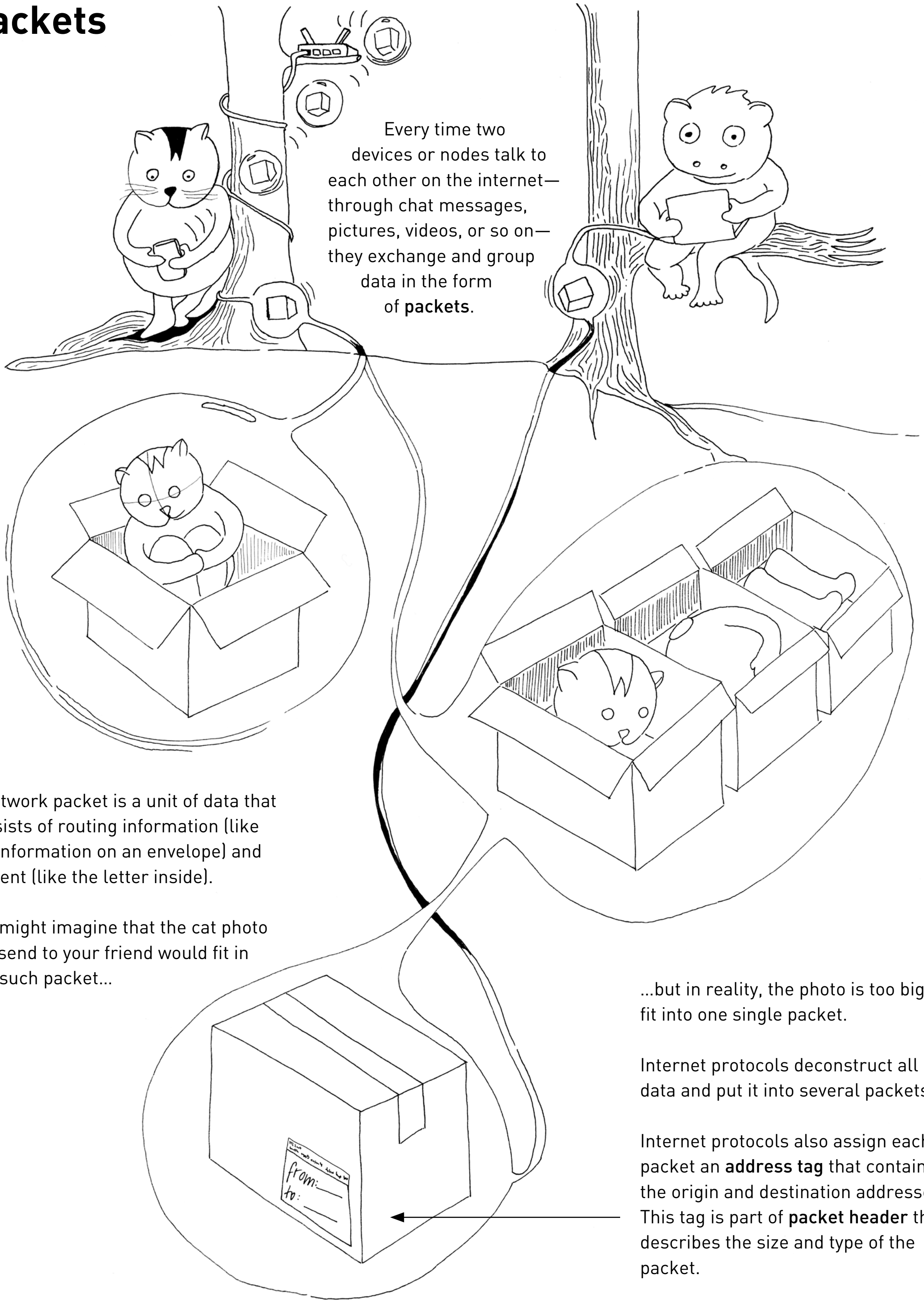
Once your network card's MAC has a network address, your device becomes part of the network and can transmit to and receive information from other networked and addressed nodes.



2

WHAT FORM DOES INFORMATION TAKE ON THE INTERNET?

Packets



Every time two devices or nodes talk to each other on the internet—through chat messages, pictures, videos, or so on—they exchange and group data in the form of **packets**.

A network packet is a unit of data that consists of routing information (like the information on an envelope) and content (like the letter inside).

You might imagine that the cat photo you send to your friend would fit in one such packet...

...but in reality, the photo is too big to fit into one single packet.

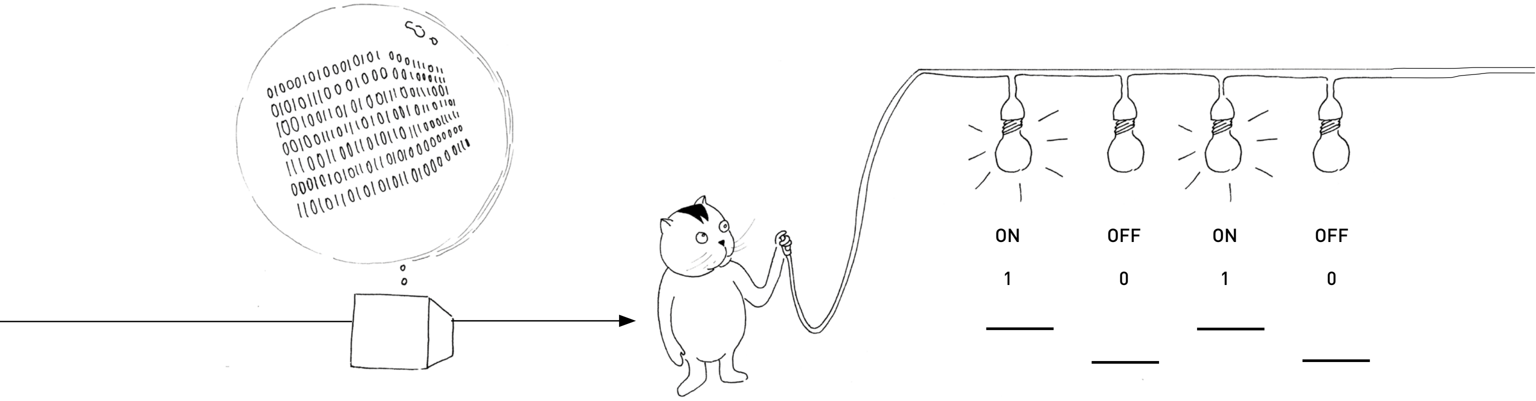
Internet protocols deconstruct all data and put it into several packets.

Internet protocols also assign each packet an **address tag** that contains the origin and destination addresses. This tag is part of **packet header** that describes the size and type of the packet.

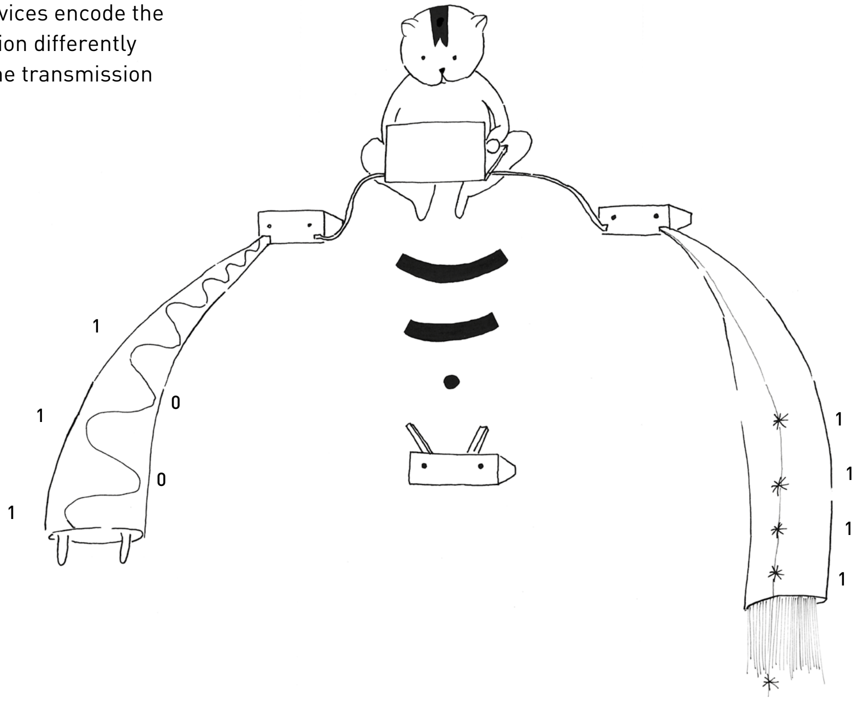
What Are Packets Made Of?

Packets are actually made up of 0's and 1's, otherwise known as binary data, because the information is stored as a sequence of two possible states.

Computers can only add and compare; therefore, "Is it a 1 or a 0?" is the most basic representation of data that a computer can process.



The network devices encode the binary information differently depending on the transmission medium:



Through copper wire, it transmits as electrical signals.

Through the air, it takes the form of radiowaves.

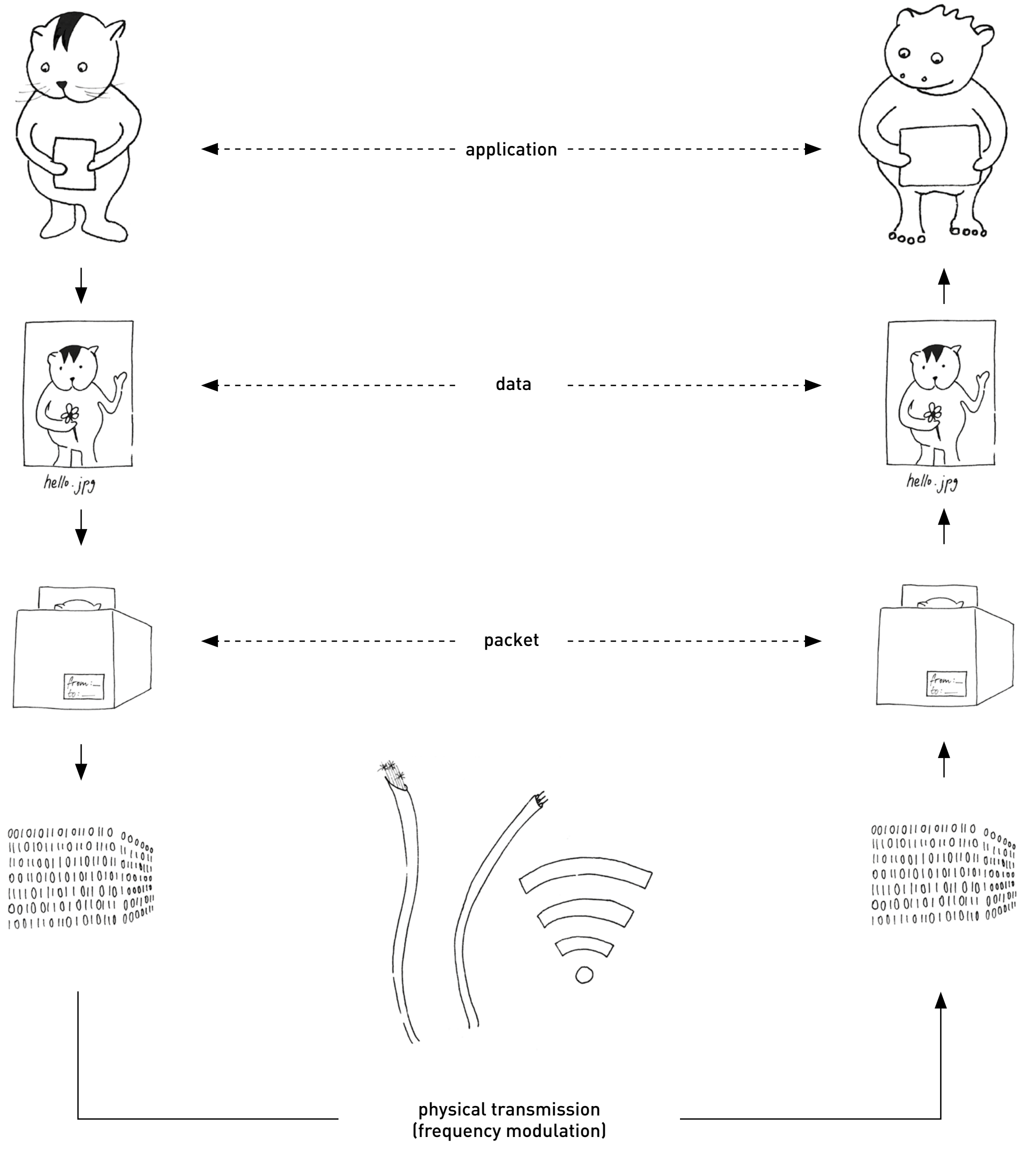
Through glass fiber, it is sent as light signals.

Transmitting Packets

Let's look at how packets are sent over the network.

Binary signals are transmitted through a process called **frequency modulation**. In this process, the transmitter translates the binary values into a series of signals that represent 0's and 1's...

...and the receiver translates the electrical, radio, or light signal frequencies back into 0's and 1's.

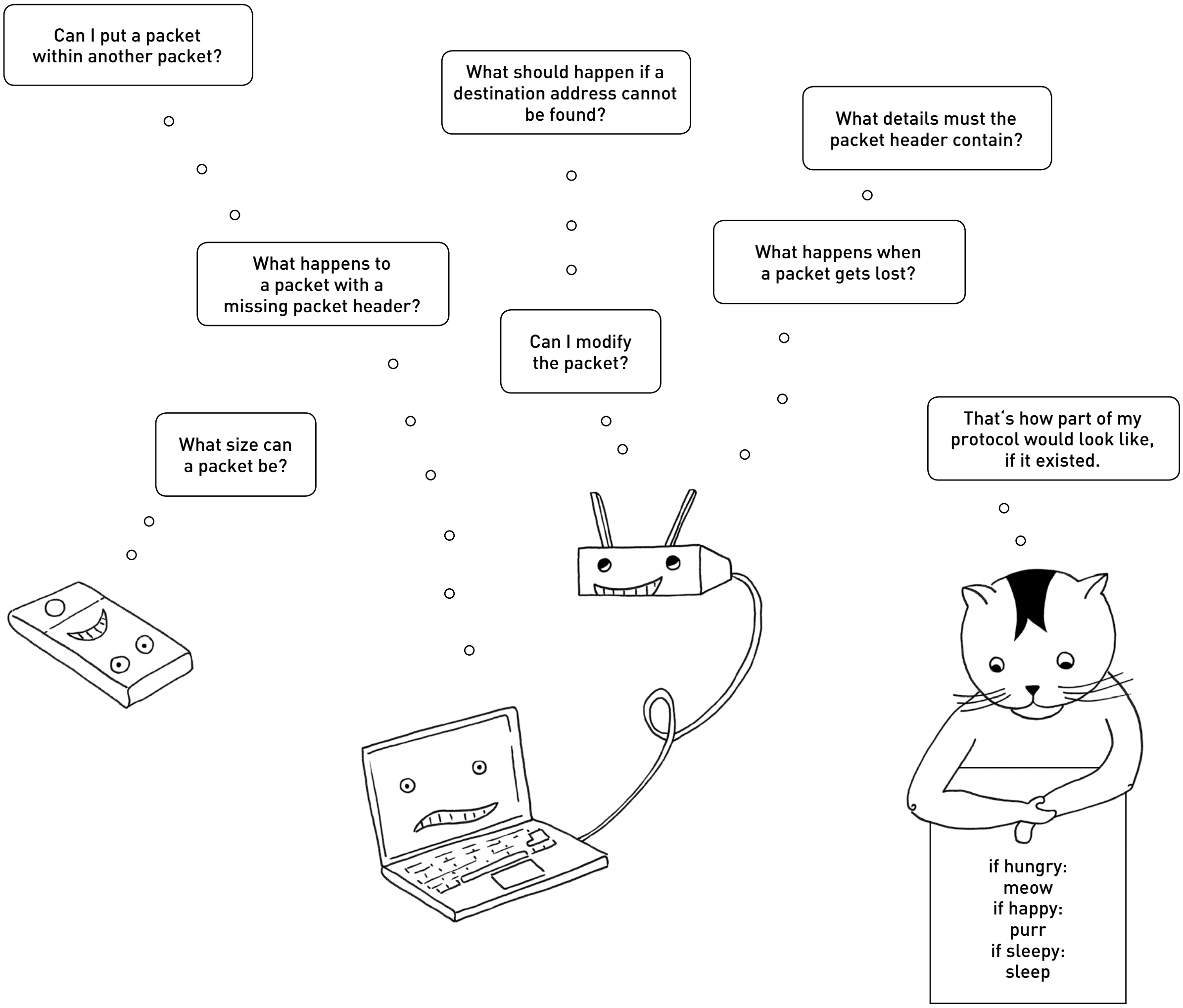


3

HOW DO DEVICES COMMUNICATE ON THE INTERNET?

Protocols

When devices talk to each other, they need a language that they both understand. We call this language a **protocol**, a system of rules with a particular syntax that defines how devices throughout the internet talk to each other and what to do when errors occur. Questions a protocol might answer:


Can I put a packet within another packet?
What should happen if a destination address cannot be found?
What details must the packet header contain?
What happens to a packet with a missing packet header?
What happens when a packet gets lost?
Can I modify the packet?
What size can a packet be?
That's how part of my protocol would look like, if it existed.
if hungry:
meow
if happy:
purr
if sleepy:
sleep

Sometimes there are different protocols for the same kind of communication. For example, several protocols define how to transport packets from node to node:

Transmission Control Protocol (TCP) is used to send packets accurately and to completion, but not necessarily in a timely way.

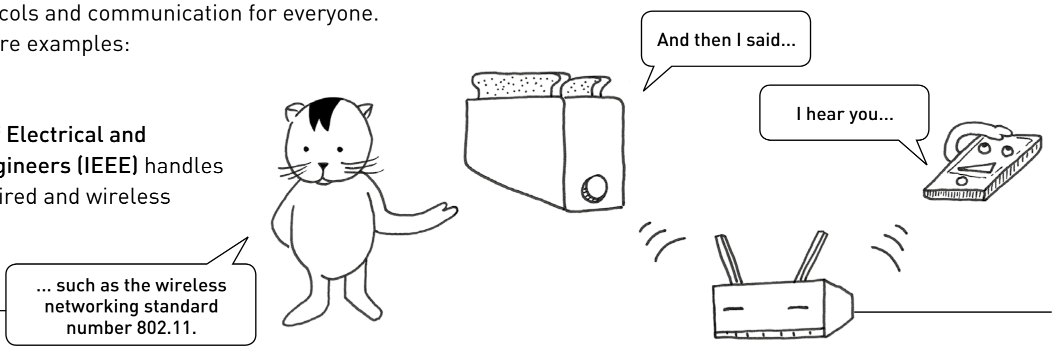
User Datagram Protocol (UDP) prioritizes speed by not worrying about the ordering or delivery of packets.

Quick UDP Internet Connections (QUIC) uses multiple, speedy UDP connections but in a manner that is reliable and accurate, like TCP.

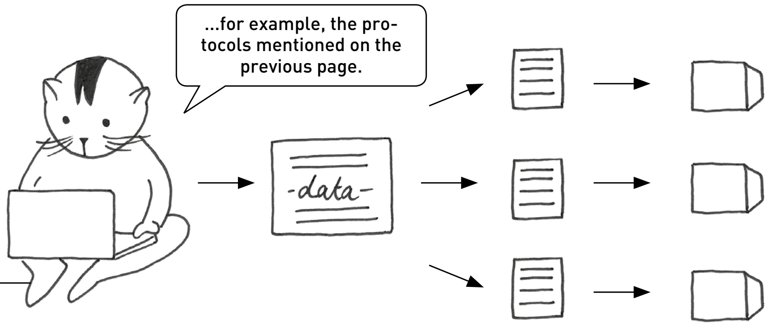
International Organizations for Protocols and Standards

International technical organizations and institutions define and improve standards for protocols and communication for everyone.
The following are examples:

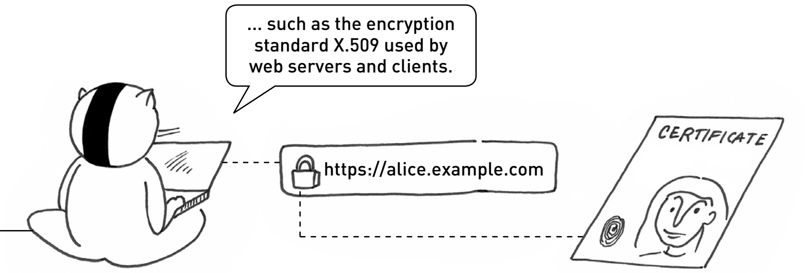
The Institute of Electrical and Electronics Engineers (IEEE) handles protocols for wired and wireless networking.



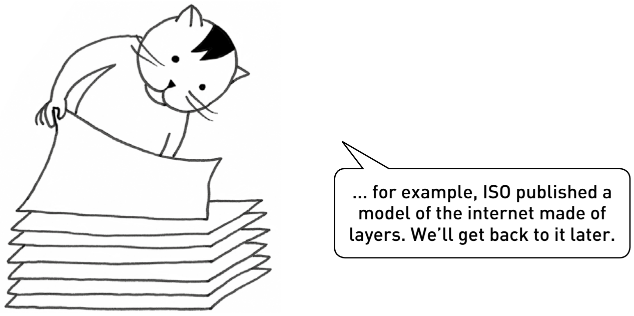
The Internet Engineering Task Force (IETF) creates and publishes internet communication protocols.


...for example, the protocols mentioned on the previous page.
data

The International Telecommunication Union's Standardization Sector (ITU-T) handles telecommunication protocols.


... such as the encryption standard X.509 used by web servers and clients.
https://alice.example.com
CERTIFICATE

The International Organization for Standardization (ISO) defines standards for application in a wide number of sectors: technology, business, government, and society.

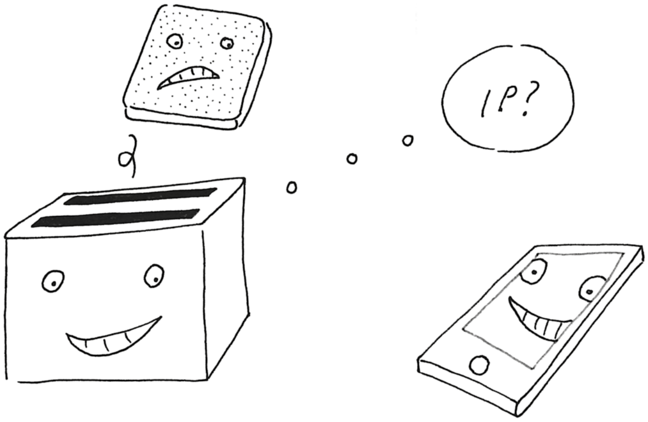

... for example, ISO published a model of the internet made of layers. We'll get back to it later.

The Internet Protocol (IP)

Let's take a closer look at one key protocol: the Internet Protocol, which defines the format of IP addresses and IP packets.

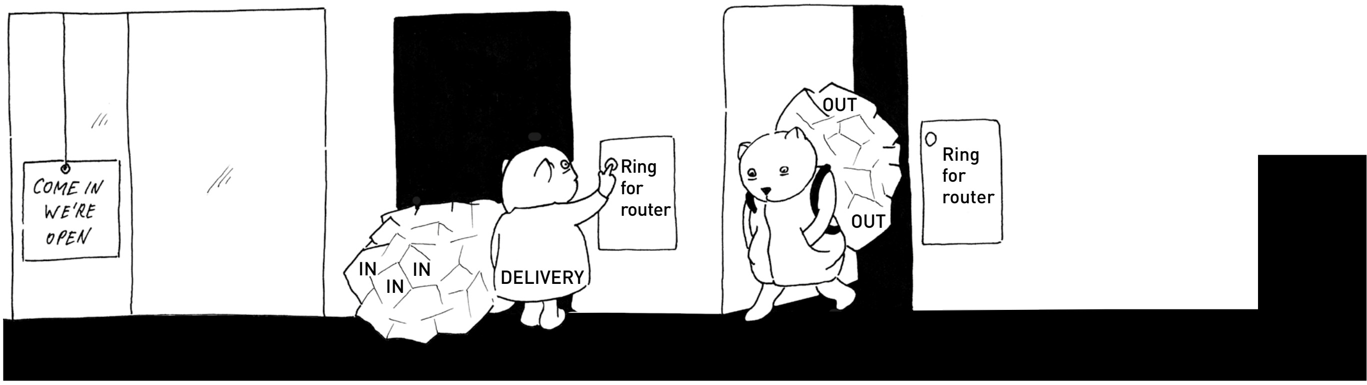
The **Internet Protocol (IP)** standardizes how packets are structured and how addresses on the internet must be formatted to deliver packets to their destination.

Once a device connects to a network, it receives a network address. To communicate with other devices on the internet, this address must follow the Internet Protocol standard. Let's look into this specific type of network address called an IP address.



Each device that speaks in "Internet Protocol" can communicate with other devices that can speak the same protocol, be they computers, phones, or even toasters.

Public and Private IP Addresses

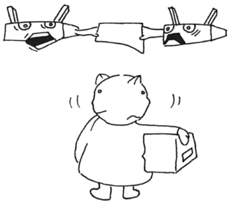


The Internet Protocol calls addresses:

public when they have direct access to the internet, like the one given to your home router by your ISP.

private when they have no direct access to the internet but connect through an intermediary, like those given to the devices connected to your home router. Private addresses are accessible only within private networks such as local networks (LAN) or virtual private networks (VPN).

Your home router has a public address, but since it handles packet delivery inside the house, it also acts as an intermediary for devices with private addresses within your home's private network.

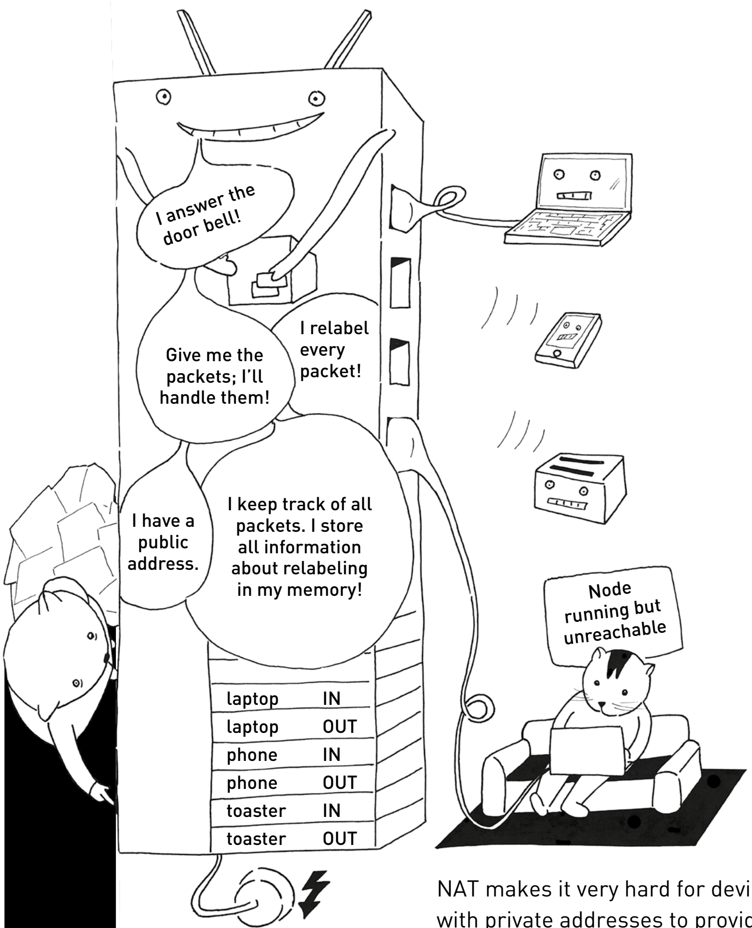


Only one device at a time can use a given address in a same network, so a public IP address must be unique for the whole planet.

Network Address Translation (NAT)

When a private network connects to the internet and sends out packets via a router, the router uses a technique called **Network Address Translation (NAT)** to rewrite the packets' address tags.

When using NAT, the router keeps all the information about who sent which packets in its memory. When the network receives a reply from the internet, the router rewrites the incoming packets' tags and sends the packets to the private address that its memory says is supposed to receive them. If the router turns off, all this information is lost.



I answer the door bell!
Give me the packets; I'll handle them!
I relabel every packet!
I have a public address.
I keep track of all packets. I store all information about relabeling in my memory!
Node running but unreachable

laptop	IN
laptop	OUT
phone	IN
phone	OUT
toaster	IN
toaster	OUT

NAT makes it very hard for devices with private addresses to provide services on the internet because they're not easily reachable from the outside.

IPv4 Addresses

NAT happens only in the fourth version of the Internet Protocol, known as **IP version 4 (IPv4)**. This is the version we still most commonly use.

An IPv4 address is made up of four blocks of numbers ranging from 0 to 255 separated by dots: 198.51.100.7

These four numbers are actually four groups of eight **binary digits (bits)**. A set of eight binary digits is often called a **byte** or an **octet**. When displayed to a user, IPv4 expresses these binary digits as decimal numbers to improve readability:

binary representation:	11000110	.00110011	.01100100	.00000111
decimal representation:	198	.51	.100	.7

Private addresses start with:

192.168.xxx.xxx
10.xxx.xxx.xxx
172.16.xxx.xxx - 172.31.xxx.xxx

Other addresses are reserved for specific use cases and cannot be used to route traffic over a network:

0.0.0.0 - 0.0.0.31
127.xxx.xxx.xxx

All other addresses are public.

IPv6 Addresses

The problem with the IPv4 address format is that it provides only approximately 4.3 billion addresses. That might seem like a lot, but with more and more devices connecting to the internet, we've run out of possible addresses! So we invented a newer version of IP—**IP version 6 (IPv6)**.

An IPv6 address is made up of eight blocks of 16-bit numbers, separated by colons. The 16-bit numbers are often represented as hexadecimal numbers, to improve readability.

binary representation	0010000000000001:0000110110111000:0000000000000000:0000000000000001: 0000000000000000:0000000000000000:0000000000010000:0000000111111111
hexadecimal representation	2001:0DB8:0000:0001:0000:0000:0010:01FF
short hexadecimal representation Consecutive blocks of zeros can be represented by a colon.	2001:0DB8:0000:0001::0010:01FF

IPv6 uses 128 bits for each address, which gives us 2^128 or 340,282,366,920, 938,463,463,374, 607,431,768,211,456 (more than 340 undecillion) possible addresses.

That's enough to provide 665,570,793,348,866,944 addresses for each square millimeter of Earth's surface.

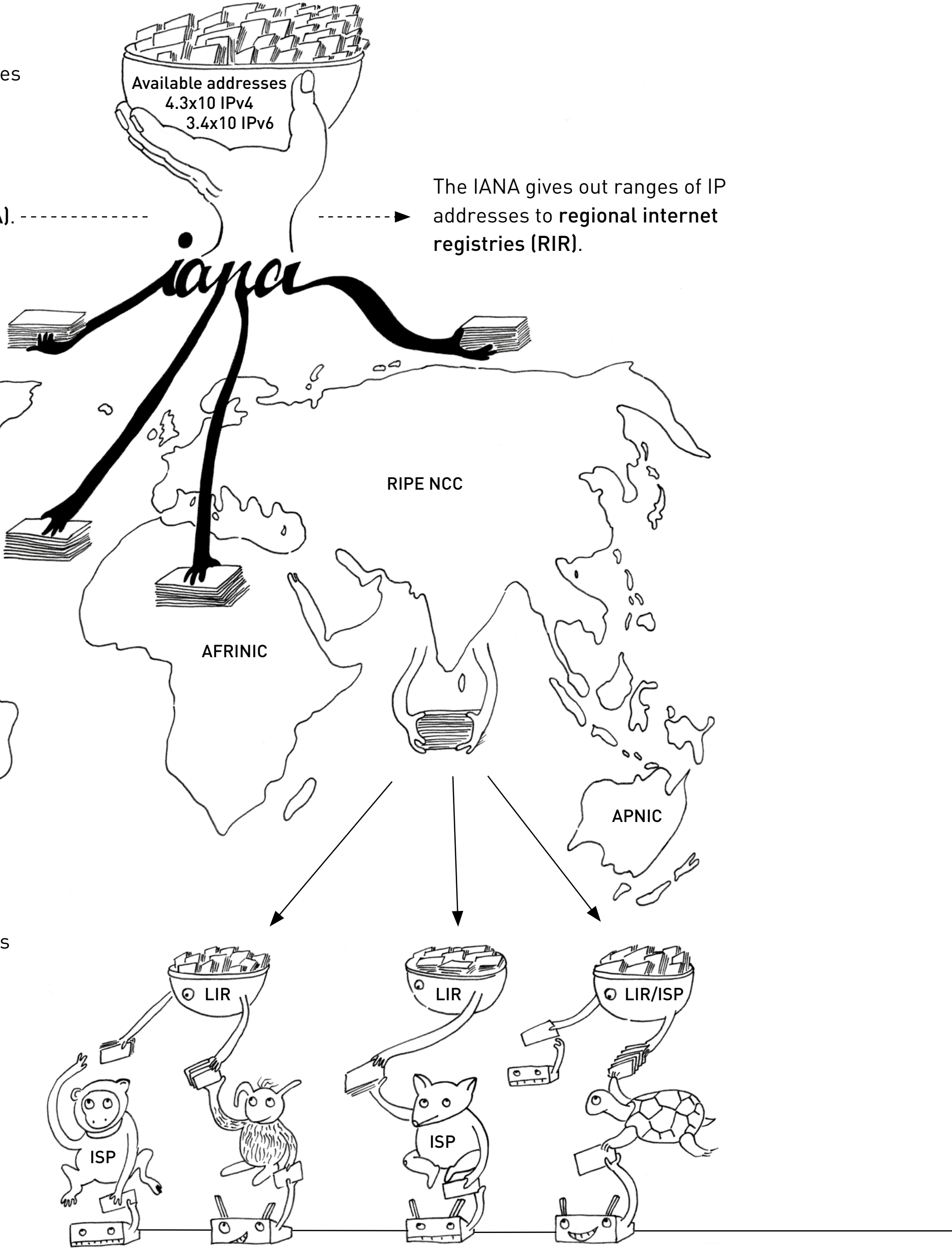
IPv6 also lets you use **unique local addresses (ULA)**, private IP addresses within networks that don't have access to the outside world. However, packets to and from these addresses are routable only within private networks and can't communicate with the global IPv6 internet.

IPv6 has another advantage over IPv4. Whenever a device connects to a home router that speaks IPv6, instead of receiving just one private address rewritten by NAT that isn't reachable from the outside world, the device receives a whole bunch of addresses that are all publicly reachable.
This makes it possible for devices to provide services and to actively participate in the internet.

Global IP Address Allocation

When a router allocates IP addresses to connected devices, it gets the addresses from a unique pool of possible public IPv4 and IPv6 addresses handled by the **Internet Assigned Numbers Authority (IANA)**.

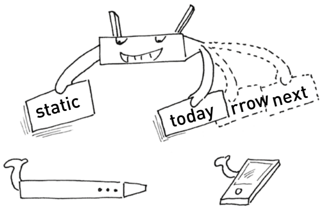
The IANA gives out ranges of IP addresses to **regional internet registries (RIR)**.



The RIRs then give out the IP ranges to **local internet registries (LIR)**.

LIRs either give out the ranges to further subcontractors or are themselves **internet service providers (ISP)**.

Finally, ISPs assign each of the routers it operates a public IP address.



It's worth noting that an IP address may be assigned **statically**, so that it's always the same for a certain device, or **dynamically**, so that it changes regularly.

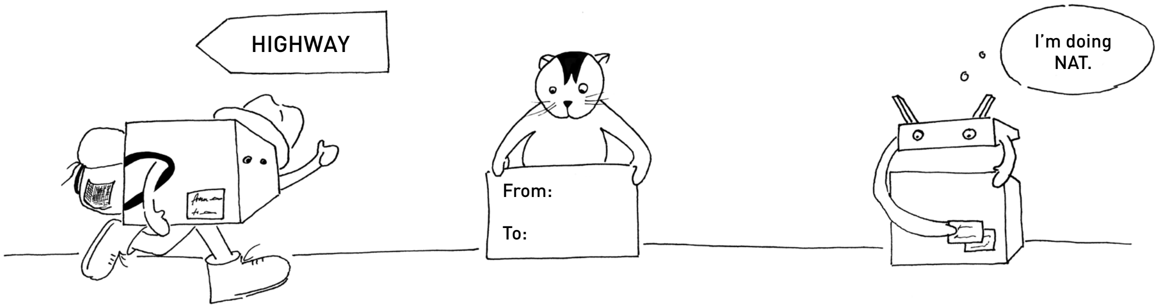
We use static IP addresses mostly for servers: they should always be reachable under the same address.

IP Routing

Now that we know how addresses on the internet look and where we get them from, let's look at how packets travel.

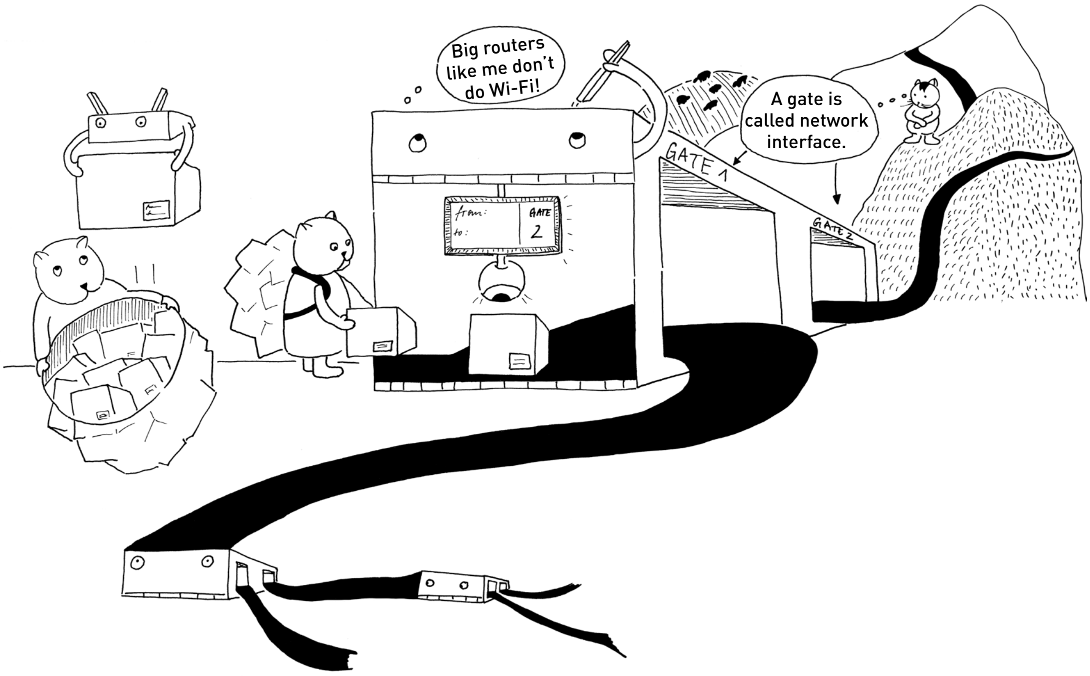
As we already mentioned, packets have tags called packet headers that contain sender and receiver addresses indicating where a packet came from and where it's going.

We've seen that on a network using NAT, a home router using IPv4 retags packets so that nodes with private addresses sending packets to the internet can receive a reply. The retagging isn't necessary on networks that don't use NAT, such as networks that use IPv6.



What happens at this point? When a router sends packets to another IP address on the internet, it first sends the packets to the next router it knows about. This is usually the big router of our ISP, which acts like a post office.

In the big router, packet headers are read and sent into the destination on the same network or to another router in the direction of the destination. The first part of an IP address indicates whether the packet targets a destination device on the router's same network, so the router knows if it has to send the packets to its own network or further on to the next router it knows about. Every router on the path does this until the packets arrive at their destination.

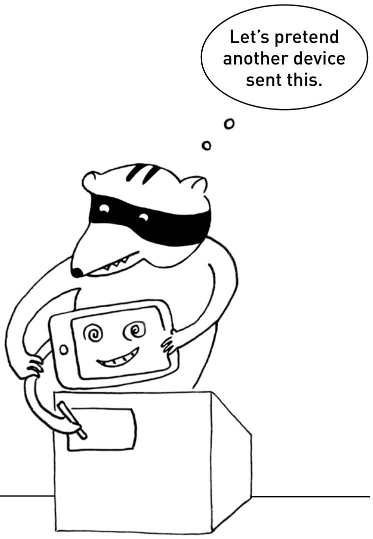

Big routers like me don't do Wi-Fi!
from:
to:
GATE
2
GATE 1
GATE 2
A gate is called network interface.

Internet Protocol Security (IPSec)

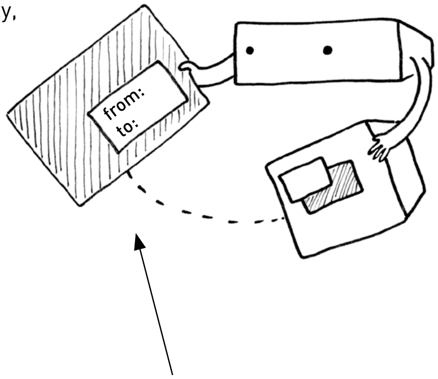
On the internet, packets are sent through several routers before arriving at their destination. Routers can read and modify packet tags but also copy, lose, drop, inspect, or modify the contents of packets.

This gives attackers the ability to send data packets while pretending to be another computer, by writing a fake sender IP address in the packet tag. This is called **IP spoofing**.

Changing the sender IP address of a packet tag lets an attacker hide their true origin and lets them remain anonymous or trick you into thinking that these packets are coming from somewhere they're not.



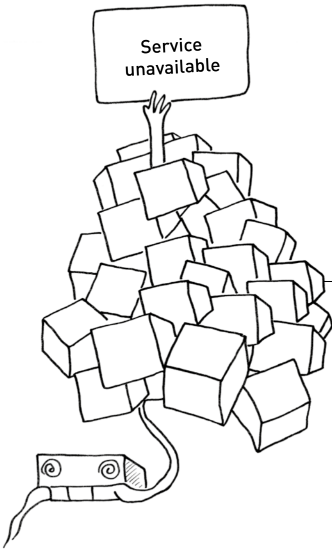
There's a solution to IP spoofing: **Internet Protocol Security (IPSec)**, a protocol that comes with a diversity of mechanisms to ensure the integrity, authenticity, and confidentiality of data packets.



Cryptographic functions provide information about the packet header.

Authenticity and integrity are provided through cryptographic functions verifying that the packet, packet header, and the originating address have not been modified or tampered with. Confidentiality is provided through encrypting the packet content.

While packets travel freely in plain IP, devices and routers using IPsec drop any packet that appears to be untrusted or otherwise invalid.

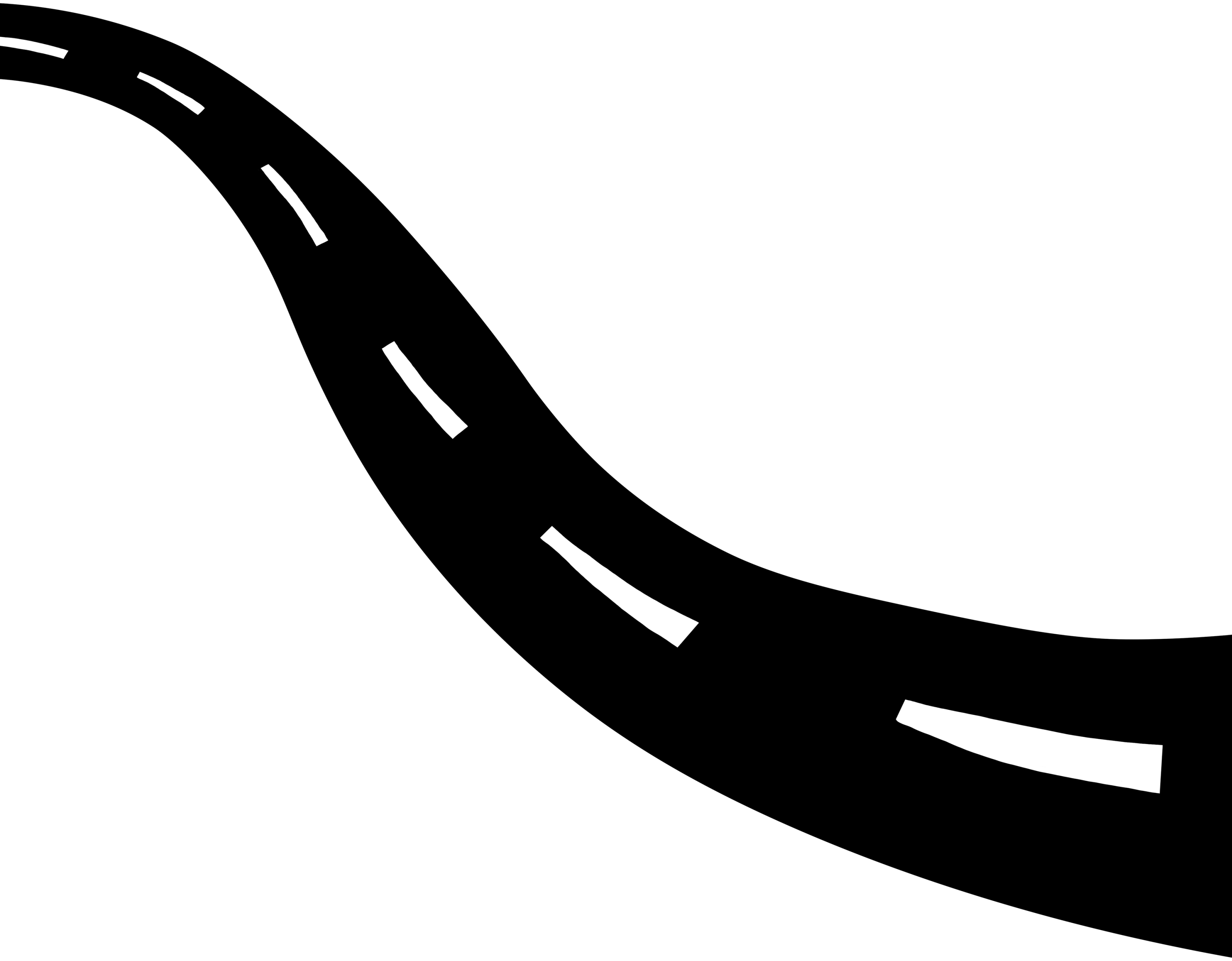


This is also how IPSec can protect against **denial-of-service (DoS)** attacks, wherein an unusually large amount of packets is sent to a target IP address, generally from many different originating addresses at the same time, until the target device is overloaded and cannot respond to requests anymore.

IPSec provides means to solve concerns about packets' integrity, authenticity, and confidentiality, but because of its complexity it is not widely used.

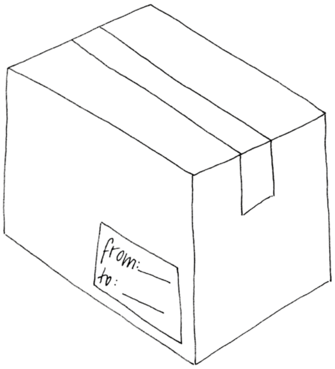
4

HOW DOES INFORMATION TRAVEL ON THE INTERNET?

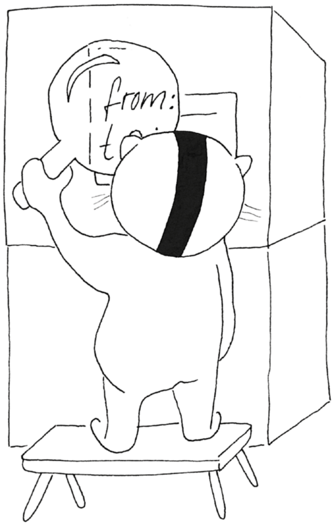


What we've learned so far:

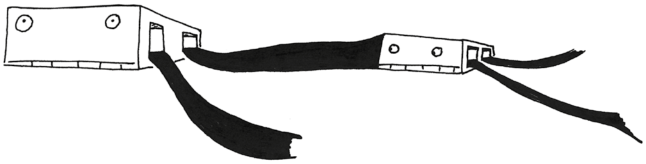
Data traveling over the internet takes the form of packets.



Packets have packet headers, which contain origin and destination addresses.



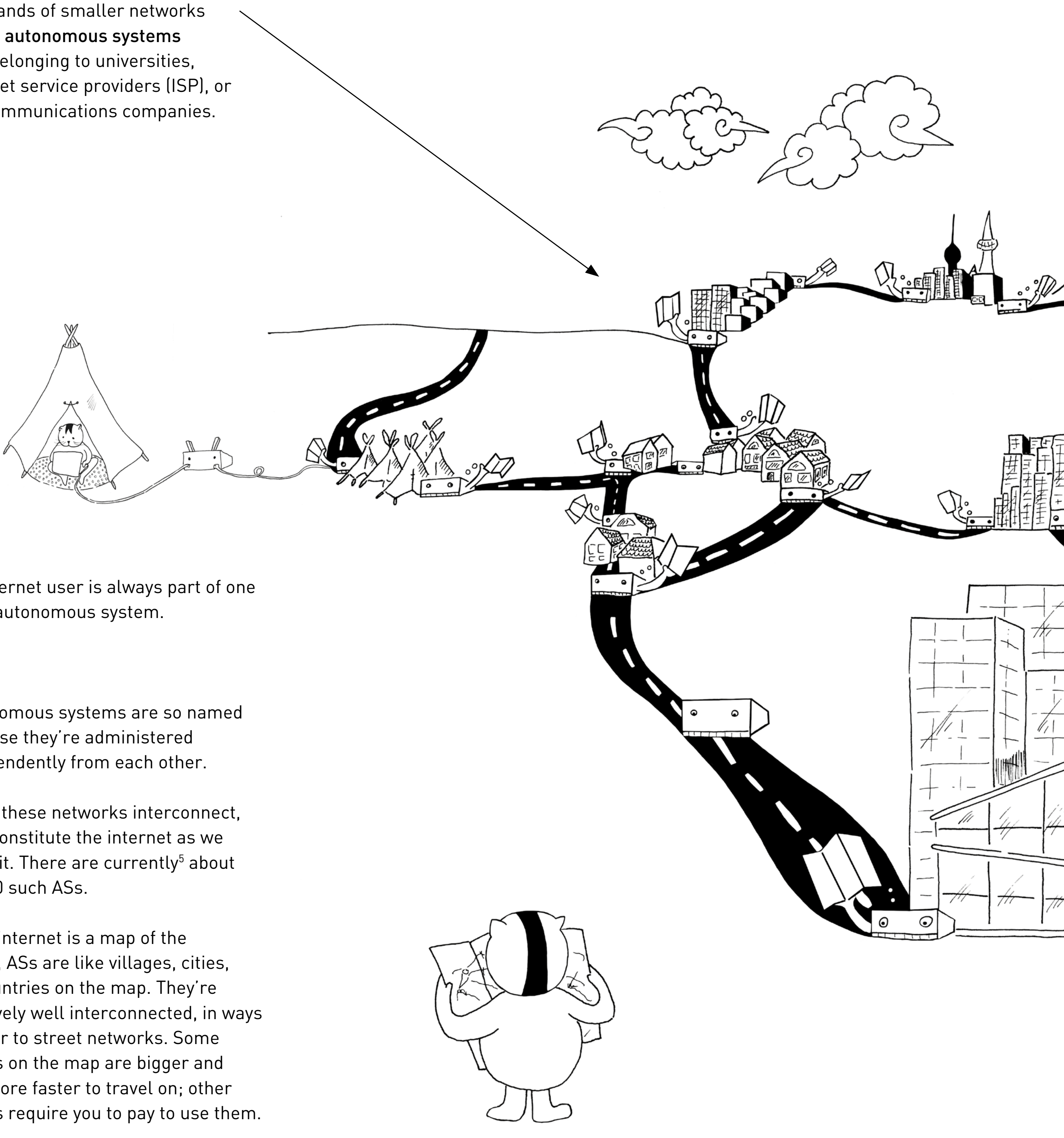
We also learned how routers direct those packets.



In this chapter we'll learn how routers know which path a packet should take by introducing the Border Gateway Protocol and internet exchange points. We'll also learn how transport protocols help establish connections between nodes on the internet and split data into packets.

The Map of the Internet

The internet isn't actually one big, unified network. Instead, it's a network made out of tens of thousands of smaller networks called **autonomous systems (AS)** belonging to universities, internet service providers (ISP), or telecommunications companies.



An internet user is always part of one such autonomous system.

Autonomous systems are so named because they're administered independently from each other.

When these networks interconnect, they constitute the internet as we know it. There are currently[5] about 97,000 such ASs.

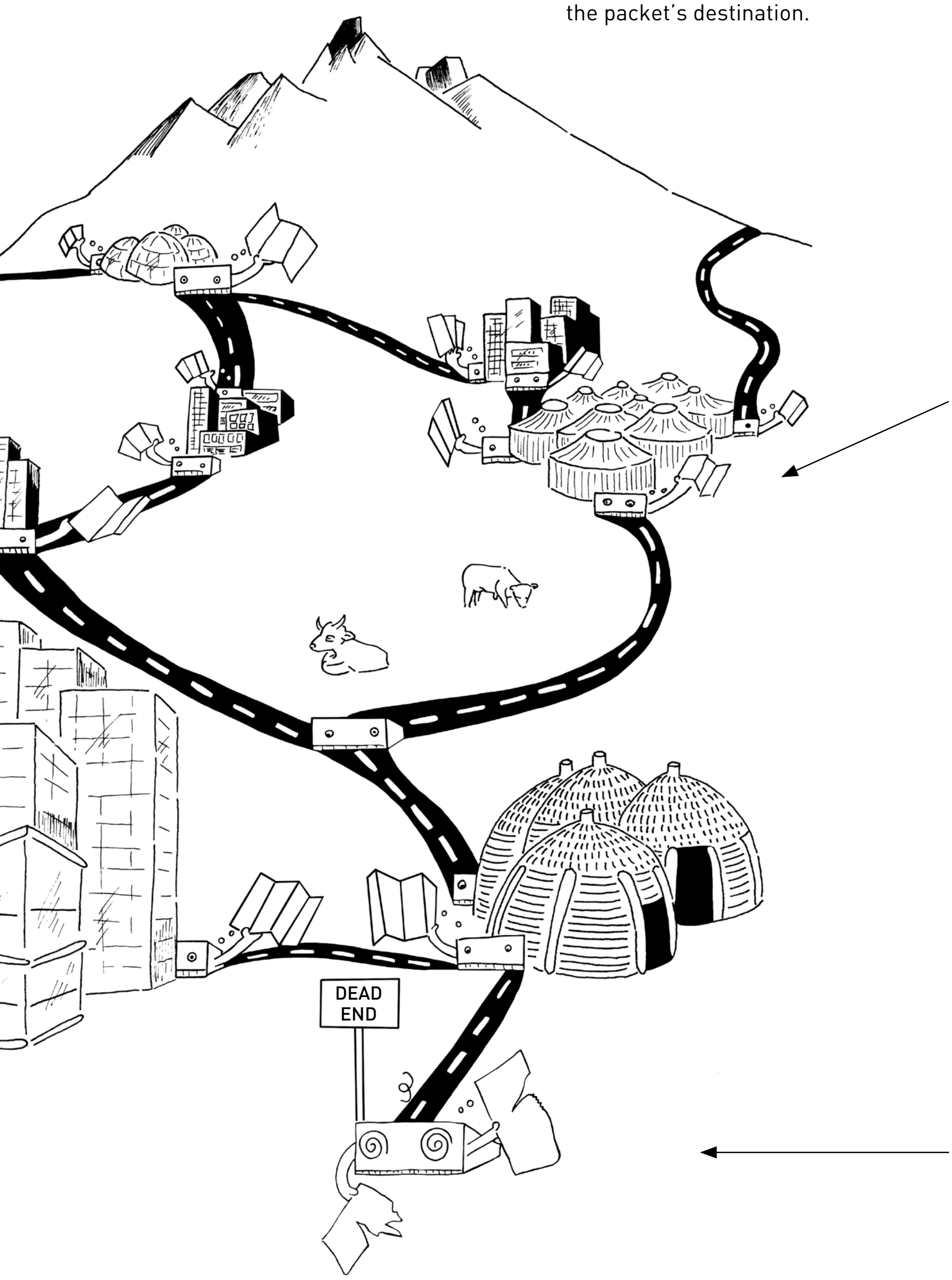
If the internet is a map of the world, ASs are like villages, cities, or countries on the map. They're relatively well interconnected, in ways similar to street networks. Some routes on the map are bigger and therefore faster to travel on; other routes require you to pay to use them.

Border Gateway Protocol (BGP)

The protocol that makes this interconnection possible is the **Border Gateway Protocol (BGP)**, the de facto routing standard on the internet.

BGP defines how information about IP packet routes are exchanged between ASs throughout the internet, making it possible to calculate the shortest and cheapest possible path from one place to the next, ultimately reaching the packet's destination.

With BGP, each AS controls its own map of the internet and references routes and distances to other networks from its own point of view. Very few BGP servers have a complete global map of all possible routes through the internet.



An AS is made up of many computers connected to each other through routers. Routers that act as entry and exit points of the whole AS are called **BGP routers**.

The BGP routers of different ASs talk to one another regularly, and when they initiate a talk, known as a session, they become **neighbors**. Whenever neighbors meet to talk, they exchange maps of all routes they know about and want to share. An AS uses BGP to keep track of routes in a table and calculates their priorities based on various attributes. BGP tends to favor an AS's own map relative to its own point of view, because an extra hop to a neighbor makes the path longer.

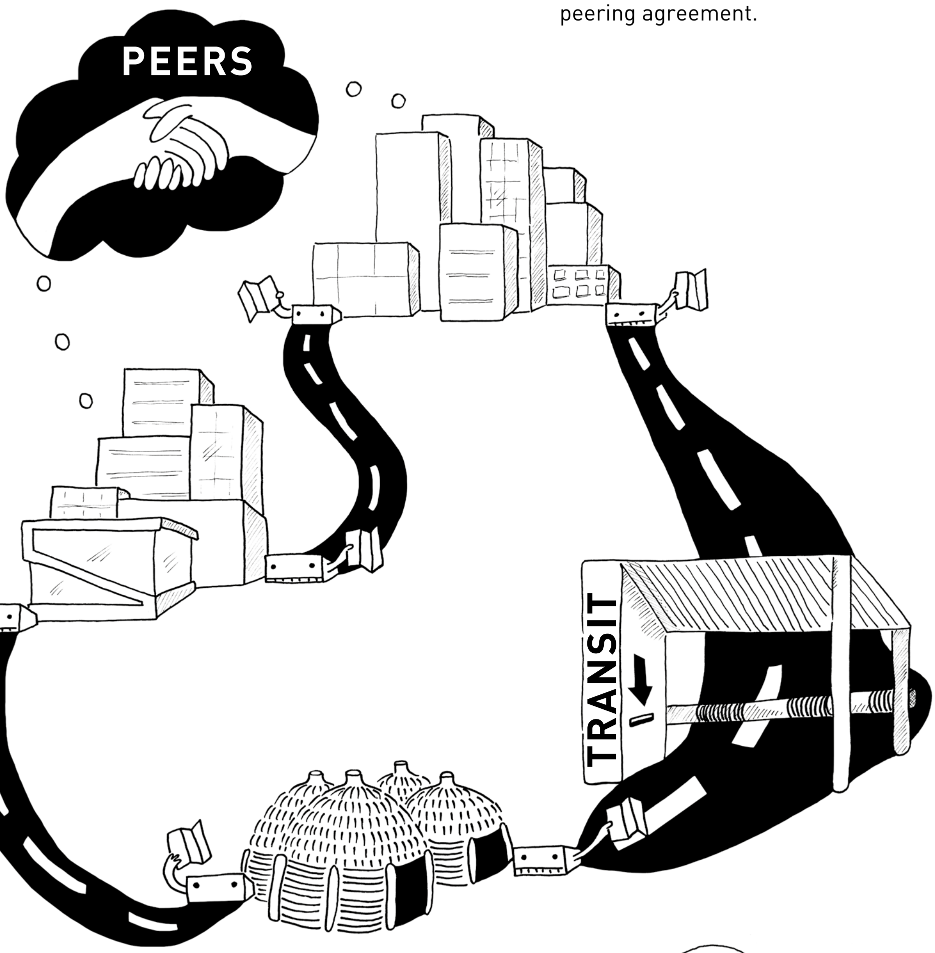
This system is very clever, but as you can guess, it's also quite prone to mistake. For example, if a neighbor shares the wrong map, or pretends to know how to get from one place to another when it actually doesn't, this can create an impasse or traffic congestion.

Peering

Big cities have big and modern highways between each other. If they have a similar number of inhabitants and traffic, they might agree to let the intercity highways be toll free. On the internet this agreement is known as **peering**.

When two ASs **peer** with each other, they agree to let data traffic pass from one AS to the other, usually for free.

But paying for peering has become common nowadays because the costs can be lower than the costs to transit across ASs with which they have no peering agreement.

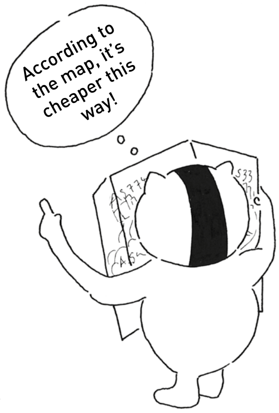


Transit

When packets from smaller places want to pass through these big cities, they likely have to pay a toll to use the big modern highways. The same goes for smaller ASs that want to exchange data with bigger ASs.

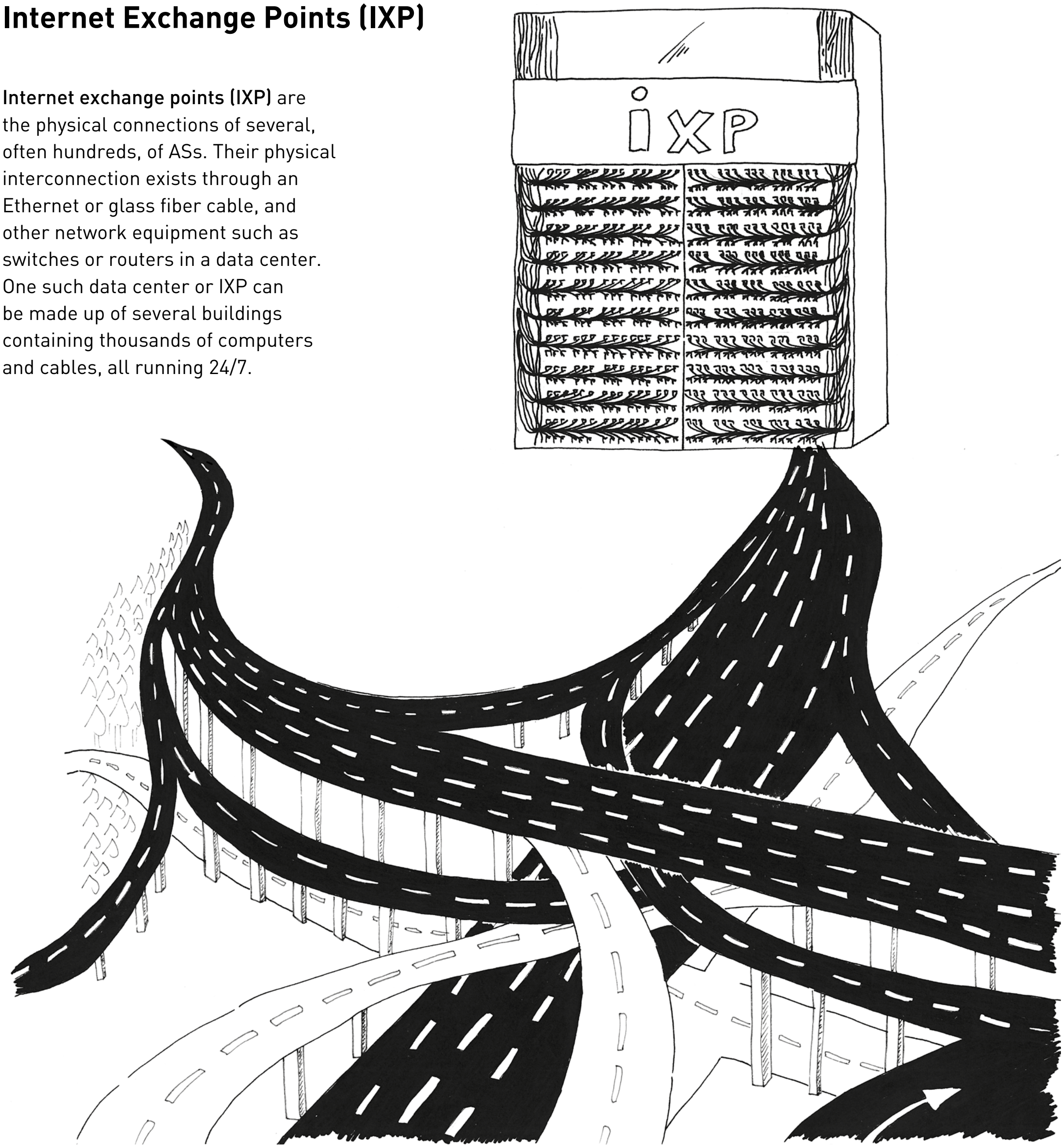
The more packets they want to bring, the more toll they have to pay. This is called **transit**. When BGP calculates the best possible path through the internet, it considers transit connections, not just the distance from one point to another, because it's cheaper for an ISP's network to send packets through a toll-free route than to pay for transit.

BGP is dynamic and optimizes packets' routing paths, but it isn't the only way to route traffic through the internet. Routes can also be statically configured.



Internet Exchange Points (IXP)

Internet exchange points (IXP) are the physical connections of several, often hundreds, of ASs. Their physical interconnection exists through an Ethernet or glass fiber cable, and other network equipment such as switches or routers in a data center. One such data center or IXP can be made up of several buildings containing thousands of computers and cables, all running 24/7.

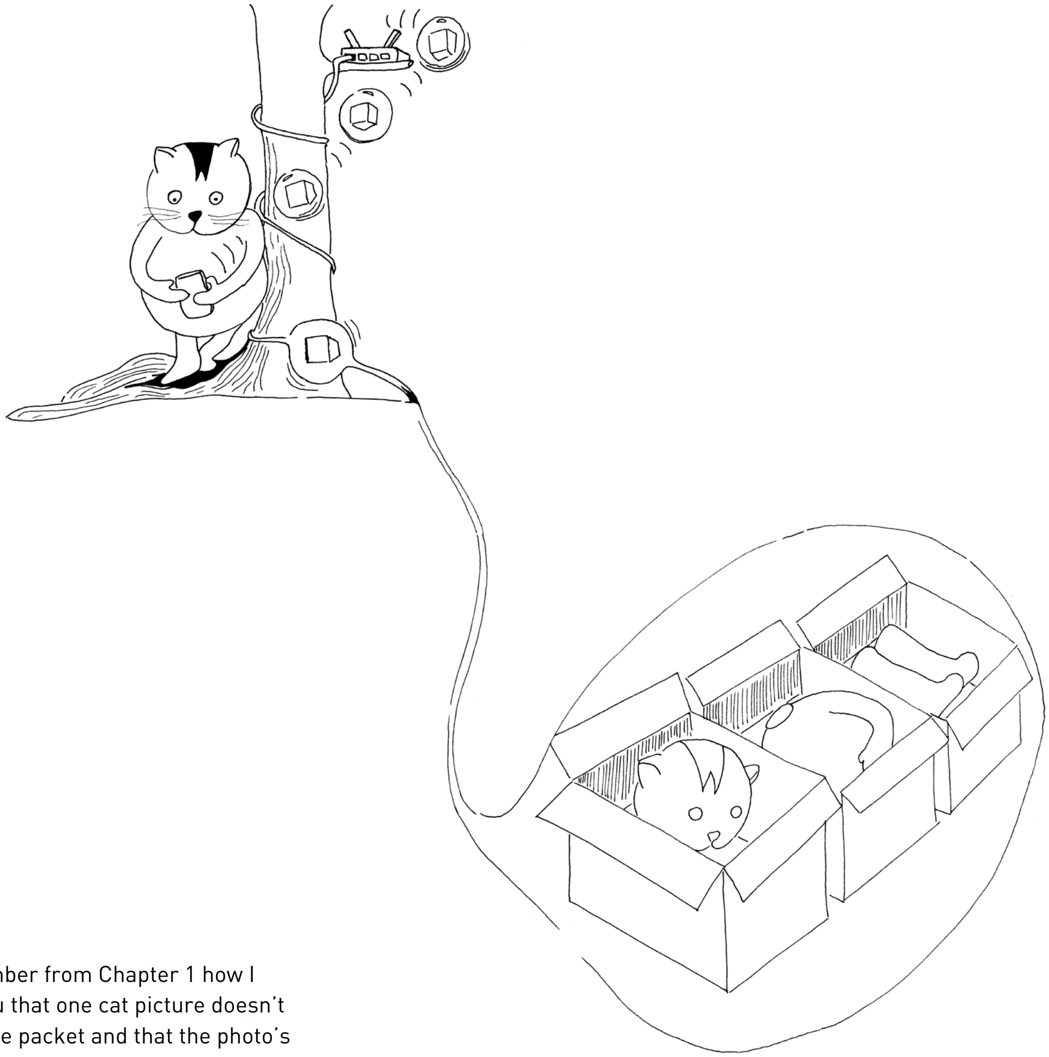


Whenever ASs connect in a data center through a cable, they effectively create a new route or path on the internet, as if they're adding one more connection to a futuristic multidimensional highway exchange with hundreds of ingoing and outgoing street connections.

Agreements over peering and transit are fundamental to AS relationships in an IXP, where members of an IXP are automatically peers. Therefore, IXP interconnections create faster paths to resources across different networks or ASs, giving end users faster access to those resources.

There are more than 1,000 IXPs worldwide. Around 240 are located in Europe and around 340 in North America.[6] The BGP facilitates the exchange of internet traffic across an IXP.

Transport Protocols



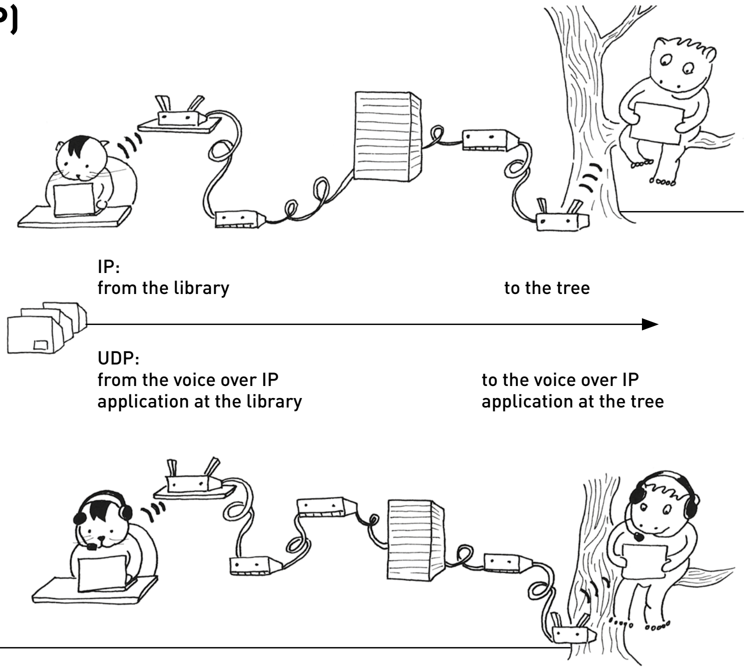
Remember from Chapter 1 how I told you that one cat picture doesn't fit in one packet and that the photo's data needs to be split into smaller pieces? It's the transport protocols that handle splitting the data into smaller pieces on the sender's side and reassembling them on the receiving side. There are several transport protocols, each one useful in its own way for different types of data exchange. We'll focus on three of them: UDP, TCP, and QUIC.

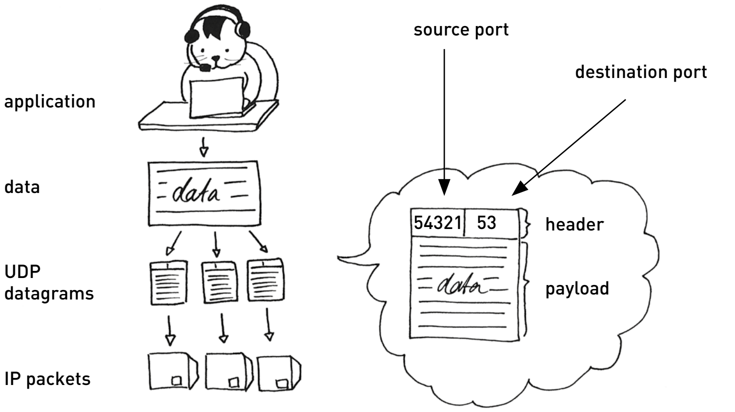
User Datagram Protocol (UDP)

The **User Datagram Protocol (UDP)** is a fundamental protocol of the internet and calls smaller units of data **datagrams**. In short, UDP prioritizes speed at the cost of reliability.

While IP determines how a source transfers packets to a destination address...

UDP adds information to the data transfer to indicate which software should handle the packets' content at the destination address.


IP:
from the library
to the tree
UDP:
from the voice over IP application at the library
to the voice over IP application at the tree

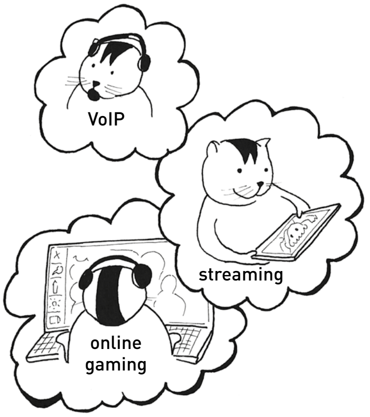


To achieve this goal, UDP adds to each datagram a header containing **port numbers**, numbers associated with a specific software or service. The header contains the port numbers of the source application and the port numbers of the software or service on the destination host. For example, the port numbers for well-known internet services such as DNS lookups use port 53.

Once the data splits, UDP sends these datagrams individually to the IP software, which in turn encapsulates each datagram into a packet and sends the packets over the internet to their destination using the UDP source-to-destination addressing scheme.

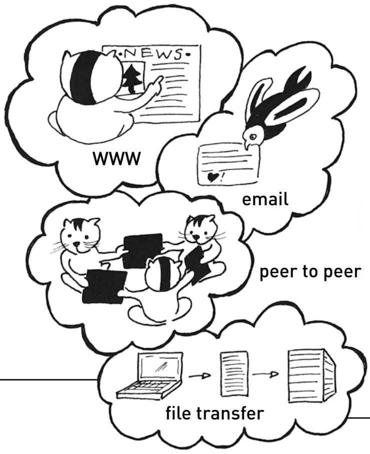
UDP doesn't keep track of whether packets arrive at their destination, arrive in the correct order, or were lost on the way. Applications that use UDP need to implement such error correction themselves.

As a result, we use UDP for cases when waiting for delayed packets or error correction isn't necessary, such as video telephony, video streaming, online gaming, and **voice over IP** applications. A missing datagram simply results in a distorted voice.



Transmission Control Protocol (TCP)

Though UDP is exactly what some applications require, other applications need all of the data they receive to be reconstructed correctly upon arrival. If packets need to arrive successfully, entirely, and in the correct order at their destination, TCP is the best choice for transport protocol.

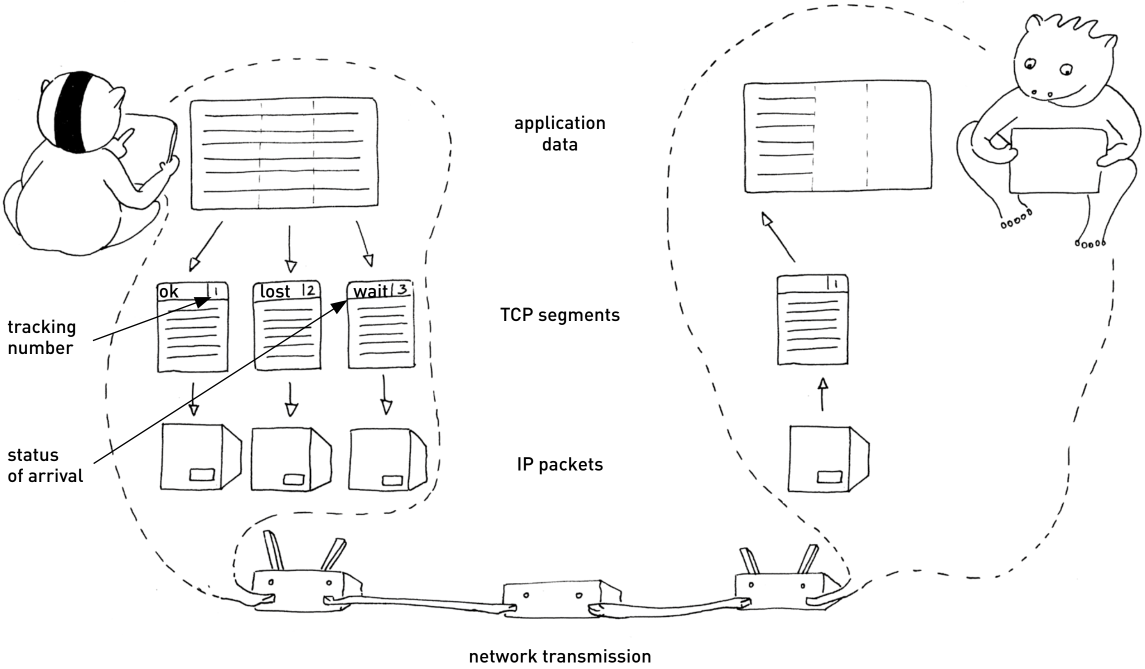


That's why the World Wide Web, email, peer-to-peer applications, file transfer, and many other applications all rely on the **Transmission Control Protocol (TCP)**, a transport protocol that provides error correction, as well as ordered and checked arrival.

When two applications want to send each other packets over the internet through TCP, TCP establishes a communication channel between two of their nodes to allow them to send data in both directions. This communication channel is also called a **pipe** or **stream**. Within the pipe, this is what happens:

On each node, TCP helps the application divide the data it wants to send into smaller segments, and numbers them to track the delivery. Once it divides the data, TCP sends all segments individually to the IP software, which in turn encapsulates each segment into a packet.

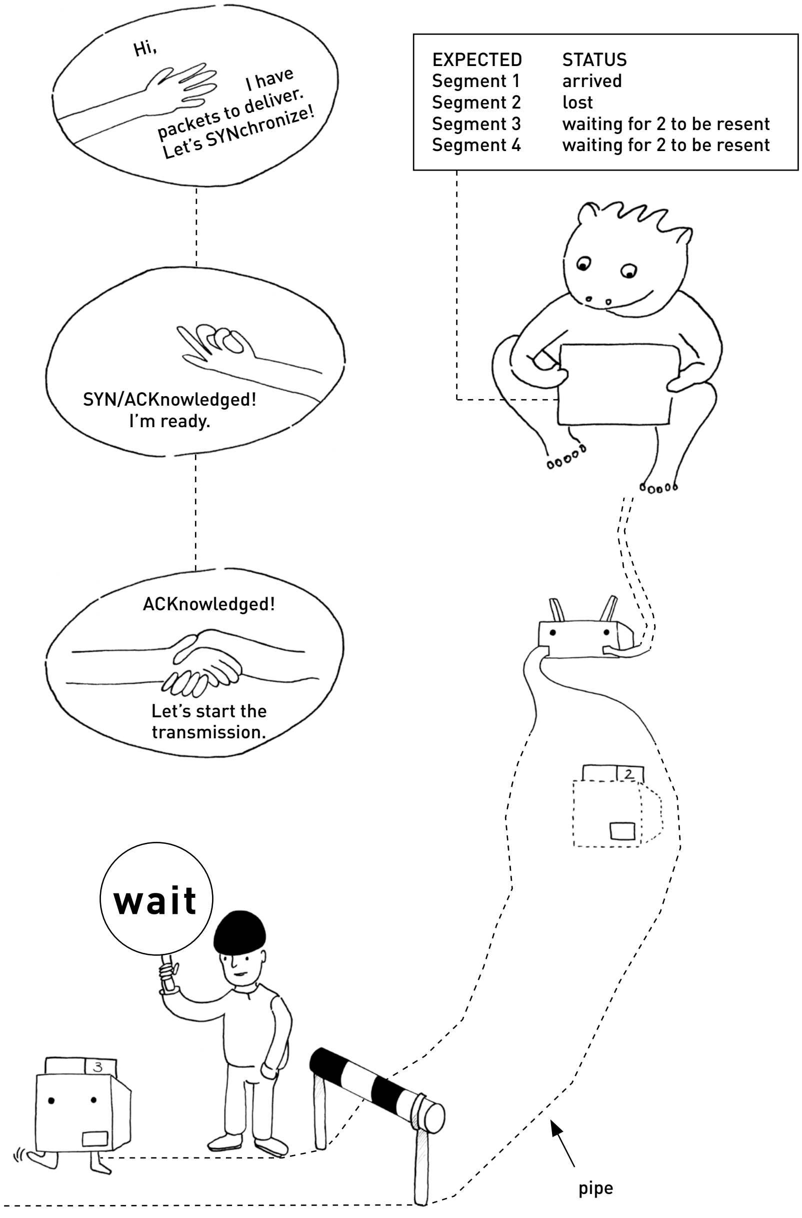
Packet delivery is tracked because TCP requires the receiving application to acknowledge packet receipt by sending back the numbers of the segments it receives.



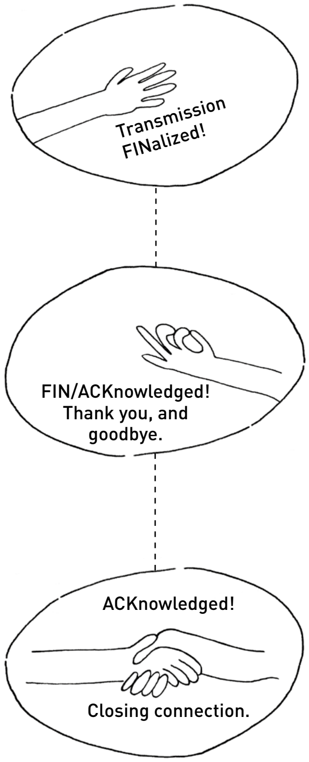
To deliver packets, applications using TCP perform something like a three-way **handshake** with each other. To start the transmission, the applications first send special packets through TCP known as SYN—SYN/ACK—ACK. They then send the packets holding the desired content.

Whenever the receiving application doesn't acknowledge receipt of one of the sent packets, or reports a packet as damaged, TCP on the sender's side sends these packets again. With TCP, if applications have a problem sending or downloading one packet, the other packets being sent must wait in the pipe until the problem is resolved.

Once the sending application delivers all packets over the network, the receiving application unpacks them one by one, puts them back into the correct order, and reassembles all the parts. The applications then end the transmission by sending special packets with the commands FIN—FIN/ACK—ACK.


Hi, I have packets to deliver. Let's SYNchronize!
SYN/ACKnowledged! I'm ready.
ACKnowledged! Let's start the transmission.
wait
3
2
pipe

EXPECTED	STATUS
Segment 1	arrived
Segment 2	lost
Segment 3	waiting for 2 to be resent
Segment 4	waiting for 2 to be resent

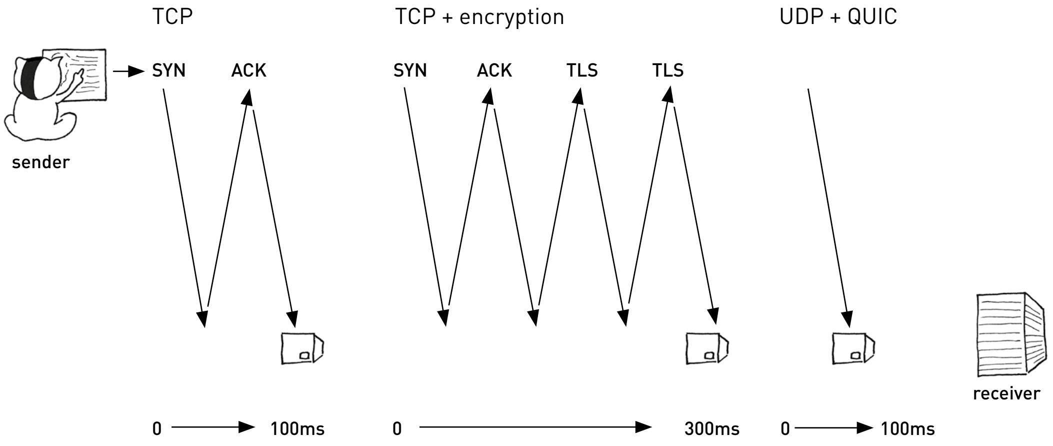

Transmission FINalized!
FIN/ACKnowledged! Thank you, and goodbye.
ACKnowledged! Closing connection.

Every packet delivery over TCP starts and ends with this three-way handshake to ensure that the applications correctly deliver all their packets. With this three-way handshake and numbering, TCP provides reliable, ordered, and error-checked delivery of data.

Quick UDP Internet Connections (QUIC)

While TCP provides integrity and reliability, it's also slow. TCP requires applications to perform a three-way handshake every time they need a session to send and receive data, and if the TCP connection is encrypted they need to perform even more handshakes.

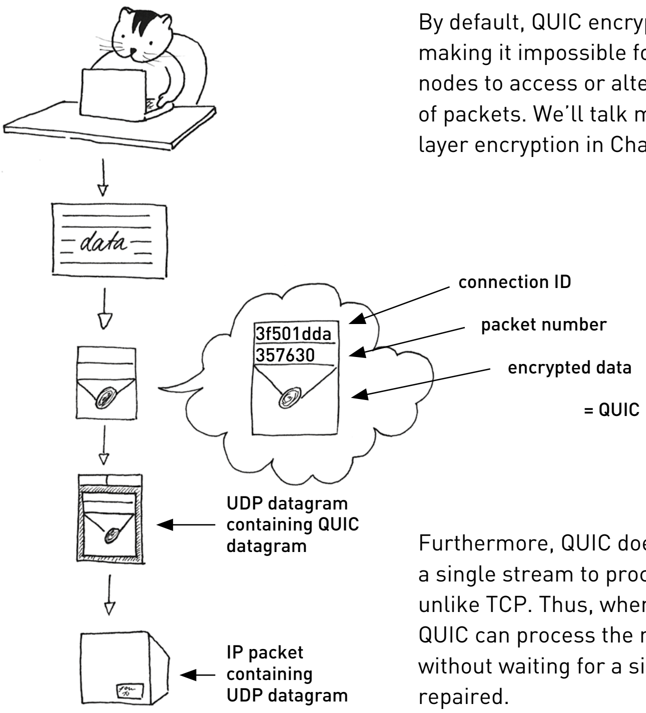
If there were a transport protocol that had both the speed of UDP and the reliability and integrity of TCP, it could make data transmissions on the internet faster overall. **Quick UDP Internet Connections (QUIC)** aims to do that.



QUIC uses the UDP transport protocol to send packets. However, before it hands datagrams over to UDP to put them into IP packets, QUIC prepares its own datagrams in a particular way.

To mimic the ordered delivery of packets as in TCP, QUIC datagrams contain information that the QUIC protocol can use on the receiving end to track and order the received datagrams.

To mimic the reliability of a TCP pipe, QUIC datagrams contain a **connection ID** that allows the sender or receiver to maintain a connection even when it's changing IP addresses, such as when your smartphone switches from a home network to a 3G connection when leaving the house.


data
3f501dda
357630
connection ID
packet number
encrypted data
= QUIC
UDP datagram containing QUIC datagram
IP packet containing UDP datagram

By default, QUIC encrypts its packets, making it impossible for intermediary nodes to access or alter the content of packets. We'll talk more transport layer encryption in Chapter 5.

Furthermore, QUIC doesn't rely on a single stream to process data, unlike TCP. Thus, when errors occur, QUIC can process the rest of the data without waiting for a single error to be repaired.

5

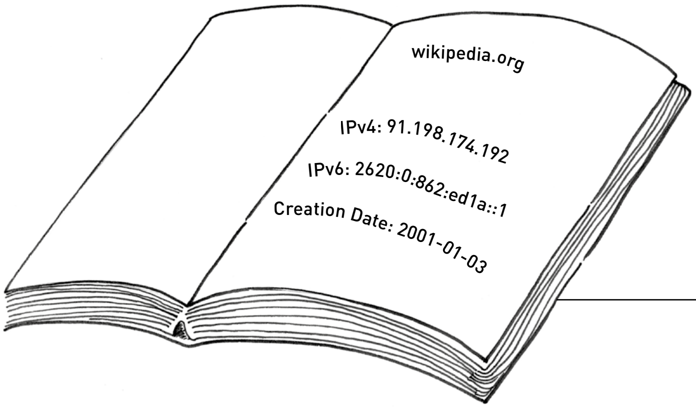
HOW DO PEOPLE RELATE TO INFORMATION ON THE INTERNET?

Domain Name System (DNS)

IP addresses are hard to remember for a human or a cat's brain.

That's why we came up with the **Domain Name System (DNS)**, a public, decentralized database that links a unique name to an IP address, a location, and other data, such as the owner's name and contact information, date of registration and expiry, and other technical details.

We call DNS the "telephone book of the internet," since we can use **domain names**, which consist of unique and memorable words, instead of complicated IP addresses when we want to access a service provided by a server.



DNS relies on UDP (see Chapter 4) to send requests for and responses to domain lookups.

A domain name consists of at least two sections, a **top-level domain**, which is the part to the right of the "dot," as well as a second-level name, which comes first before the "dot."

Example Domain Name	Namespace	Administrator	Zone
en	Hostname	Domain Owners	Subdomain
.wikipedia	Second-Level Name	Registrars and Domain Owners	Domain
.org	Top-Level Name	Registries/TLD Operators	Top-Level Domain (TLD)
.	.	ICANN	Root

This table breaks down the domain name
en .wikipedia .org .

That's the hostname.

That's the second-level name, or domain name.

That's the top-level domain name, abbreviated TLD.

The DNS has a hierarchical structure based on zones of control. Therefore, each level, or namespace, comes with its own **zone** of administrative control:

en
.wikipedia
.org
.

Subdomain zone(s): The owner of a domain can create subdomains, sub-subdomains, sub-sub-subdomains, and so on. They must maintain the records for namespaces at the subdomain level and below. While an administrator may allocate subdomains or **hostnames** for various purposes, no one can purchase a subdomain or hostname from a domain registry or registrar separately.

Domain zone: Anybody can buy a domain name from a **registrar**, an organization or company accredited by the **Internet Corporation for Assigned Names and Numbers (ICANN)** to sell domain names. The person who purchases a domain name is called the **registrant.**

TLD zone: We distinguish **generic top-level domains (gTLD)**, such as .com, .org or .net and **country code top-level domains (ccTLD)**. There are currently more than 290 ccTLDs and 1,200 gTLDs. More gTLDs are being added all the time. Each **top-level domain (TLD)** is maintained and serviced technically by a **registry**.

Root zone: DNS root servers are maintained by the Internet Corporation for Assigned Names and Numbers (ICANN). There are 13 root servers on Earth, which together host registries that account for all top-level domains. The root zone uses multiple autonomous and redundant root servers to maintain resiliency of the DNS.

Registries are entities that have applied to ICANN to control a TLD. National registries, often tied to government entities, maintain ccTLDs. A registry defines what conditions a buyer or registrar needs to meet in order to own or sell a domain under their TLD zone.

Using the example in the table as your guide, try to answer some questions about this domain name: **unique-and.memorable .com**

What is the top-level name?

What is the hostname?

Who are the administrators of this domain name?

What is the zone of .memorable?

Answer: .com

Answer: unique-and

Answer: Domain owners, registrars, registries/TLD operators, and ICANN.

Answer: Domain

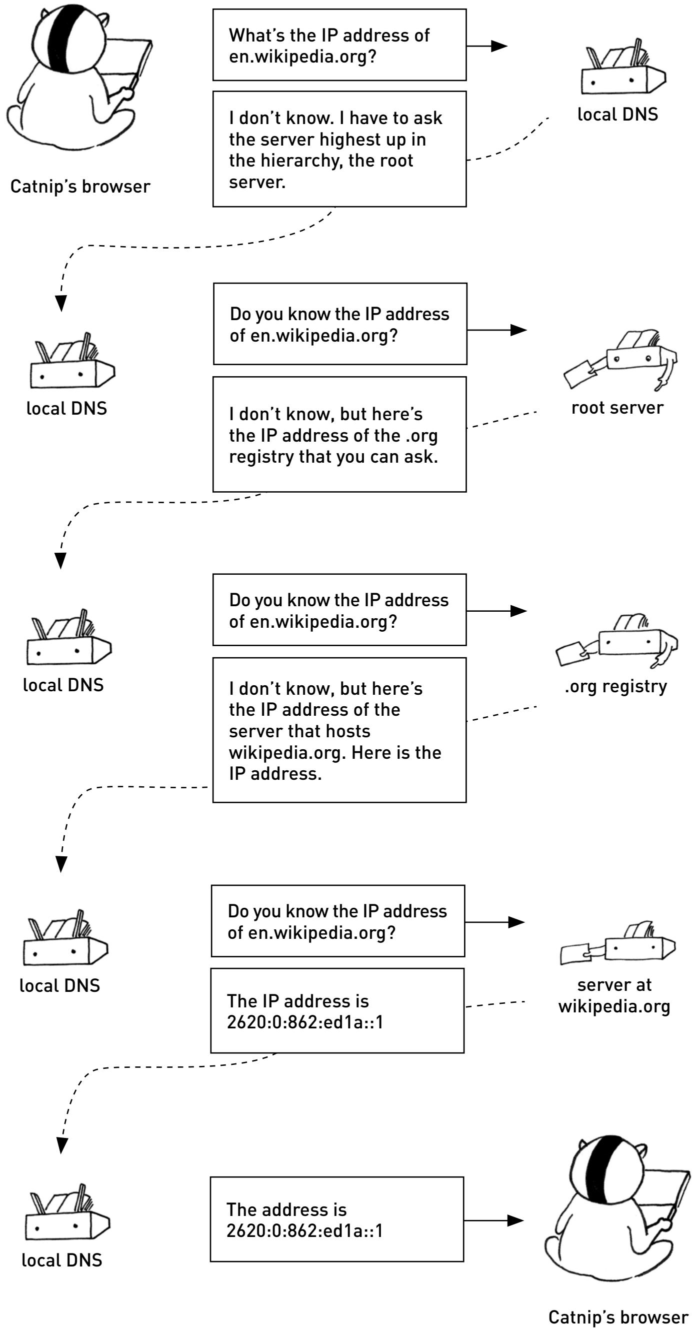
How Does a Domain Name Resolve Back to an IP Address?

When we use a domain name to access a service, DNS **resolves** the name into an IP address, which means that the DNS has a defined protocol for how to look up the IP address of where the service is hosted. Each time we type a domain name (such as en.wikipedia.org) into the address bar of our browser, the browser makes a DNS request to a DNS server.

By default, internet service providers operate DNS servers for their clients that look up a domain name and send back the corresponding address. Our DNS request retrieves the IP address of that name in order for the browser to send our request for the content of en.wikipedia.org to the correct server on the internet.

When your home router assigns and sends your computer an IP address, it also sends the address of a DNS server that can resolve names back to addresses. Sometimes this is the address of the router itself, or **name server** performing the DNS lookup. We call this the **local DNS resolver**.

If Catnip enters a domain name into their browser, their browser asks the local DNS resolver what the domain name's IP address is. If the local DNS resolver doesn't know, it asks the server highest up in the hierarchy: the root server. If the root server doesn't know, it gives the local DNS resolver the IP address of the server of the top-level domain, and if the TLD doesn't know, it gives the IP address of the server of the domain that gives the IP address of the domain name. Finally, the local DNS resolver sends back the IP address to Catnip's browser, letting Catnip connect to the site.



What's the IP address of en.wikipedia.org?

I don't know. I have to ask the server highest up in the hierarchy, the root server.

Do you know the IP address of en.wikipedia.org?

I don't know, but here's the IP address of the .org registry that you can ask.

Do you know the IP address of en.wikipedia.org?

I don't know, but here's the IP address of the server that hosts wikipedia.org. Here is the IP address.

Do you know the IP address of en.wikipedia.org?

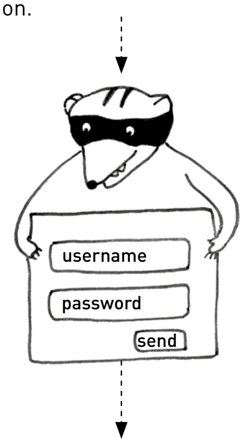
The IP address is 2620:0:862:ed1a::1

The address is 2620:0:862:ed1a::1

As you can see, this process requires many connections, so the first time we make a DNS request, our browser, computer and even the ISP often saves, or **caches**, this information to make future DNS requests faster.

DNS Security Extensions (DNSSEC)

DNS is vulnerable to attacks that hijack and take control of any step of the DNS lookup process. Such attacks can direct users to a malicious website for username and password collection.

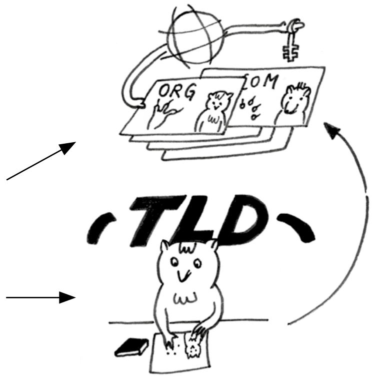


To prevent this, the IETF has made security extensions to DNS on top of the DNS protocol, also known as **DNSSEC**. DNSSEC digitally signs DNS data and provides DNS resolvers with a method to authenticate the data they receive before passing it on.

In order to trust the signature, DNSSEC relies on a trust chain:

ICANN owns the keys that sign the root zone and publishes the keys of the TLD registries.

The TLD zone operator sends their own key to ICANN.

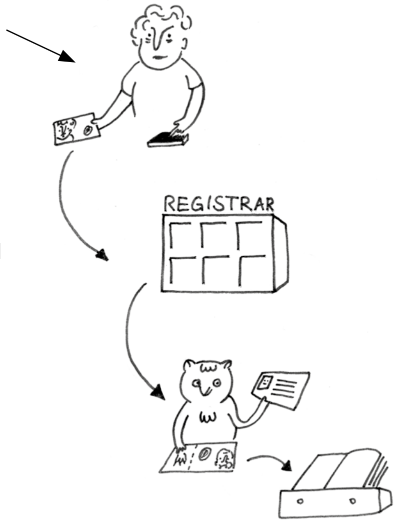


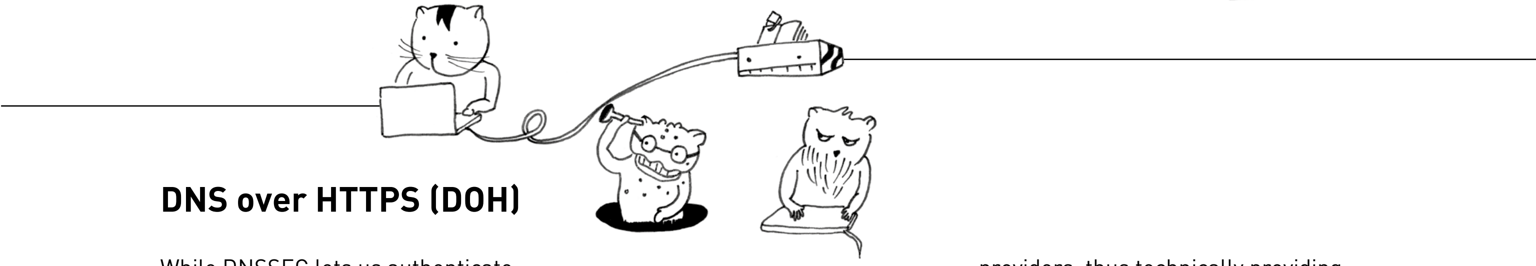
Domain owners generate their own keys, which they upload to their domain name registrar.

The registrar sends the domain owner's keys on to the TLD zone operator.

The TLD zone operator then signs and publishes the domain owner's keys in the DNS.

When we implement DNSSEC at each step of the DNS lookup process, we can verify and trust each lookup.





DNS over HTTPS (DOH)

While DNSSEC lets us authenticate DNS requests by building a trust chain, it doesn't provide information privacy. Every DNS request is observable.

To address privacy, the IETF community developed a new DNS protocol called DNS over HTTPS. HTTPS is the same protocol that secures connections to websites.

Instead of sending DNS requests to the local DNS resolver of the ISP, **DNS over HTTPS (DOH)** uses an encrypted connection over HTTPS (HTTP+TLS) to connect to a DNS resolver of the application's or user's choice. This makes the DNS request unobservable for intermediaries involved in the connection, such as local ISPs, and allows users to identify trusted DNS providers, thus technically providing more privacy to users.

Furthermore, since HTTPS is a widely used protocol, it's harder to implement censorship by blocking domains—but only as long as DOH resolvers are decentralized and do not all belong to the same entity, which could in turn carry out censorship itself.

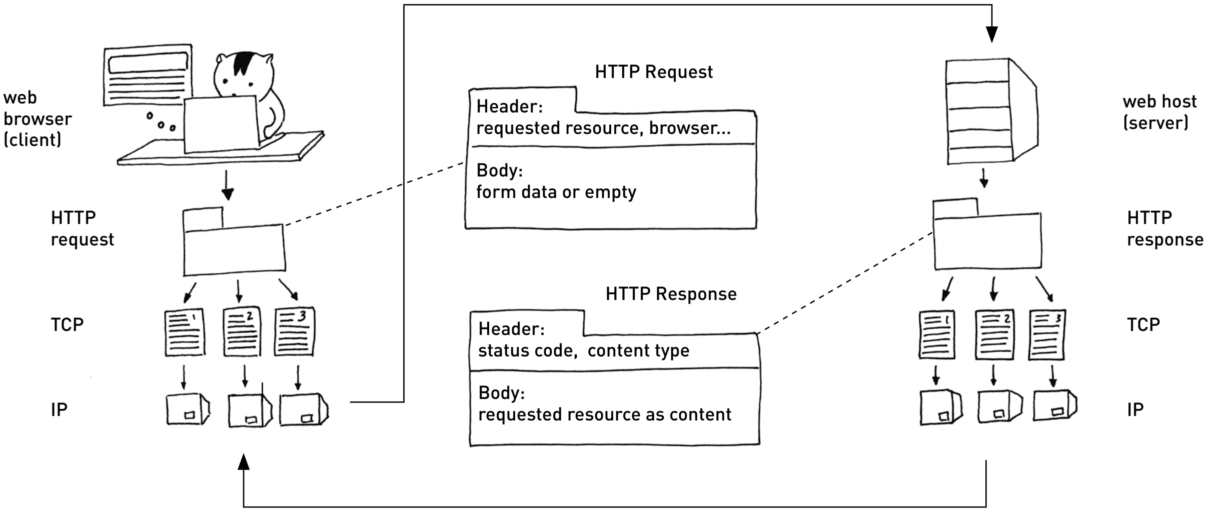
Hypertext Transfer Protocol (HTTP)

The **Hypertext Transfer Protocol (HTTP)** is the protocol for exchanging or transferring **hypertext**, structured text that uses logical links called hyperlinks to refer to other content on nodes. You are using HTTP every day when connecting to websites. HTTP is the foundation of data communication over the World Wide Web.

HTTP is a request–response protocol, relying on a client–server model. When we visit a website, watch a movie, use apps on our smartphones, or upload data using our browser, the client submits an HTTP request message to the server.

The server, which provides resources such as HTML files, images, videos, and other content, returns a response message to the client. The response contains information regarding the request, such as the completion status and eventually the requested content.

HTTP works at the application layer, so it's a protocol that makes a web browser and a web server understand each other. But the actual transport of the data happens through the protocols at the transport layer, usually the **Transmission Control Protocol (TCP)**.



The server responds by sending back HTTP data over the same TCP connection that was used for the request. These HTTP responses return the message body, which contains the requested data or content, whether it's an HTML page, an image, a video, or a file. The responses also contain HTTP headers that describe the data.

HTTP Headers

HTTP stores information about both requests and responses in **HTTP headers**, which the browser and the web server read. These headers aren't visible to users, but they transmit various types of information between client and server, such as what type of data is returned, your device's operating system, the type of browser that you're using, and the type of content being requested. They also transmit an **HTTP status code** that reports on the completion of the request.

HTTP Status Codes

200 = ok (request completed)
301 = moved permanently
(redirection)
404 = not found
418 = I'm a teapot[7]
451 = unavailable for legal reasons
500 = internal server error

The status code 404 is also known as "error 404" and indicates that a requested resource cannot be found on the server.

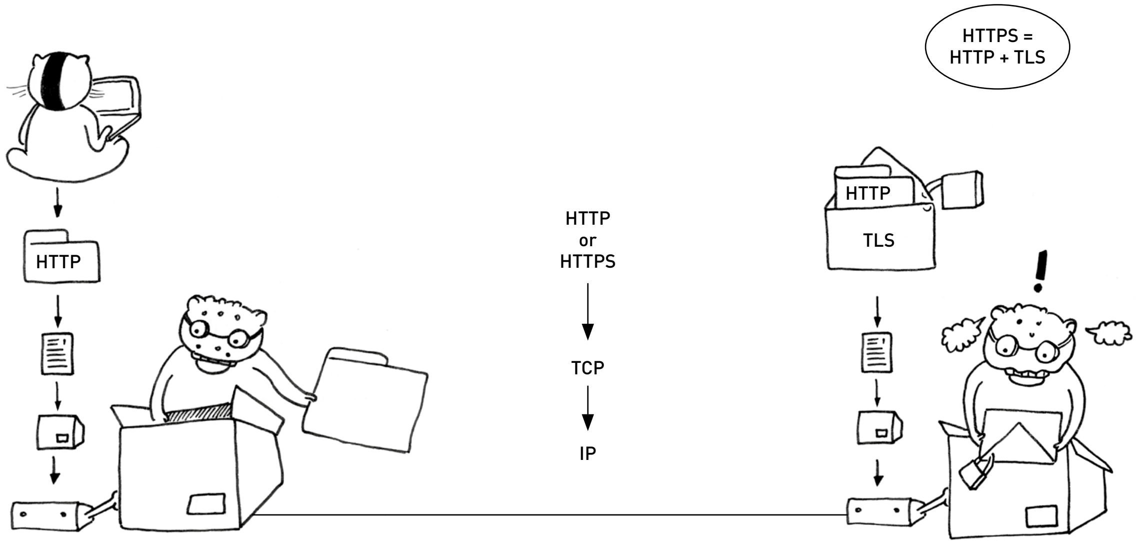
Secure HTTP: HTTPS

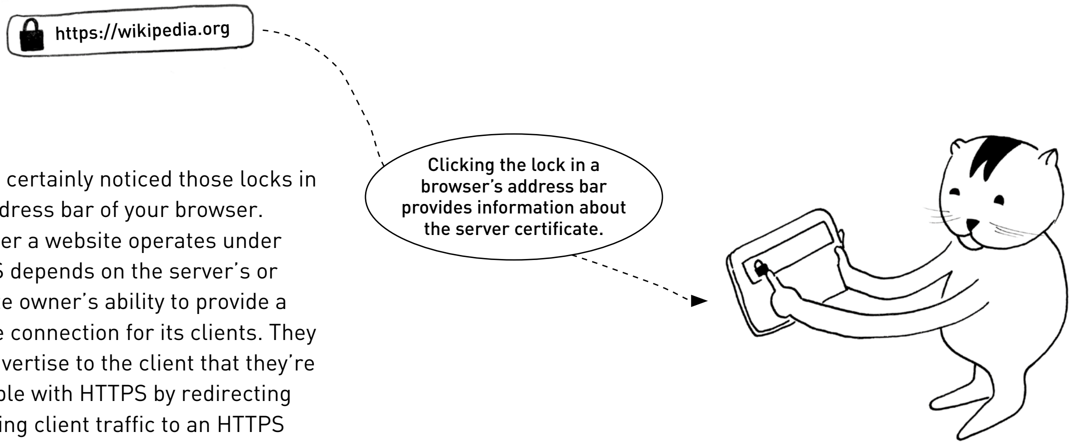
HTTP isn't privacy sensitive. Routers and intermediate devices, such as those controlled by your ISP, and the websites you visit can read and modify all of the information transmitted over HTTP, including the HTTP headers, the originating and destination IP addresses, and even the response data. This is because HTTP itself isn't encrypted.

Privacy and security are what **Secure HTTP (HTTPS)** is for.

HTTPS wraps the HTTP message into encrypted message envelopes before they are transported over the network. HTTPS conceals the message body and HTTP headers, but not the origin and destination IP addresses that indicate which nodes are talking to each other.

Sometimes called Hypertext Transfer Protocol Secure, HTTPS is not actually an acronym.[8] HTTPS is purely defined as securing HTTP connections by means of the cryptographic protocol Transport Layer Security (TLS), which we'll explain next.




https://wikipedia.org
Clicking the lock in a browser's address bar provides information about the server certificate.

You've certainly noticed those locks in the address bar of your browser. Whether a website operates under HTTPS depends on the server's or website owner's ability to provide a secure connection for its clients. They can advertise to the client that they're available with HTTPS by redirecting incoming client traffic to an HTTPS port rather than an HTTP port.

Transport Layer Security (TLS)

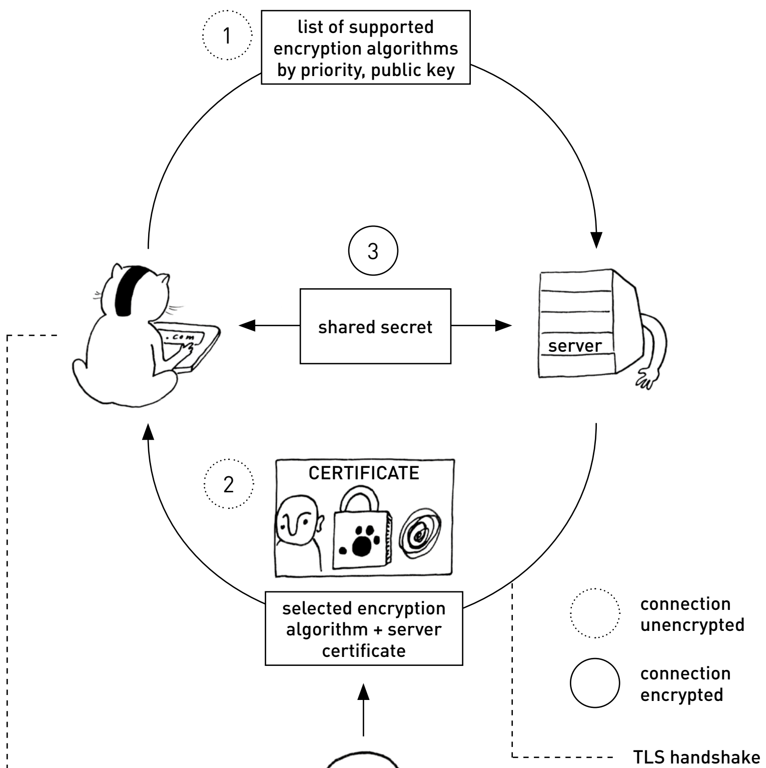
HTTPS can rewrap HTTP messages with encrypted message envelopes because of a cryptographic protocol called Transport Layer Security.

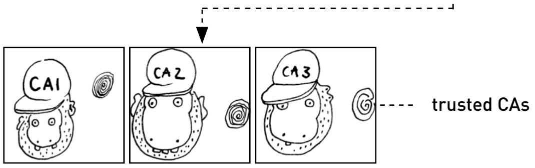
Transport Layer Security (TLS) provides privacy for data in transit, the ability to authenticate the identity of communicating nodes, as well as a message integrity check to prevent undetected loss or alteration of data.

In order to set up a TLS connection between server and client, their communicating nodes establish a shared secret key through a handshake. Once the two nodes set up a connection and a shared secret key, all of the data they exchange is encrypted.

TLS works with websites, email, chat, and many other applications. QUIC and DOH use TLS by default.

TLS relies on trusted third-party organizations called **certificate authorities (CA).** CAs issue digital **certificates** to service operators that attest the ownership of the operator's key. The certificate contains information about the server name, the owner's identity, a copy of the public key, and a cryptographic signature of the CA. Certificates can be issued for single or multiple domain names.


1
list of supported encryption algorithms by priority, public key
3
shared secret
server
2
CERTIFICATE
selected encryption algorithm + server certificate
connection unencrypted
connection encrypted
TLS handshake



Client applications need to make sure that they are communicating with the correct server by verifying the certificate prior to exchanging data. This is done automatically during the handshake by the application, through comparing the signature on the certificate against a list of trusted CAs.

The TLS trust mechanism is weak: CAs can get compromised or forced to issue false certificates, and client applications might be tricked into accepting them.[9]

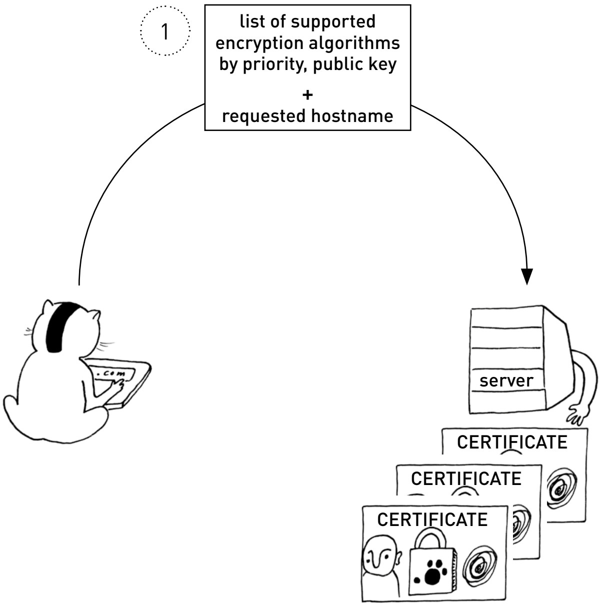
We'll learn more about cryptography and transport encryption on the next pages.

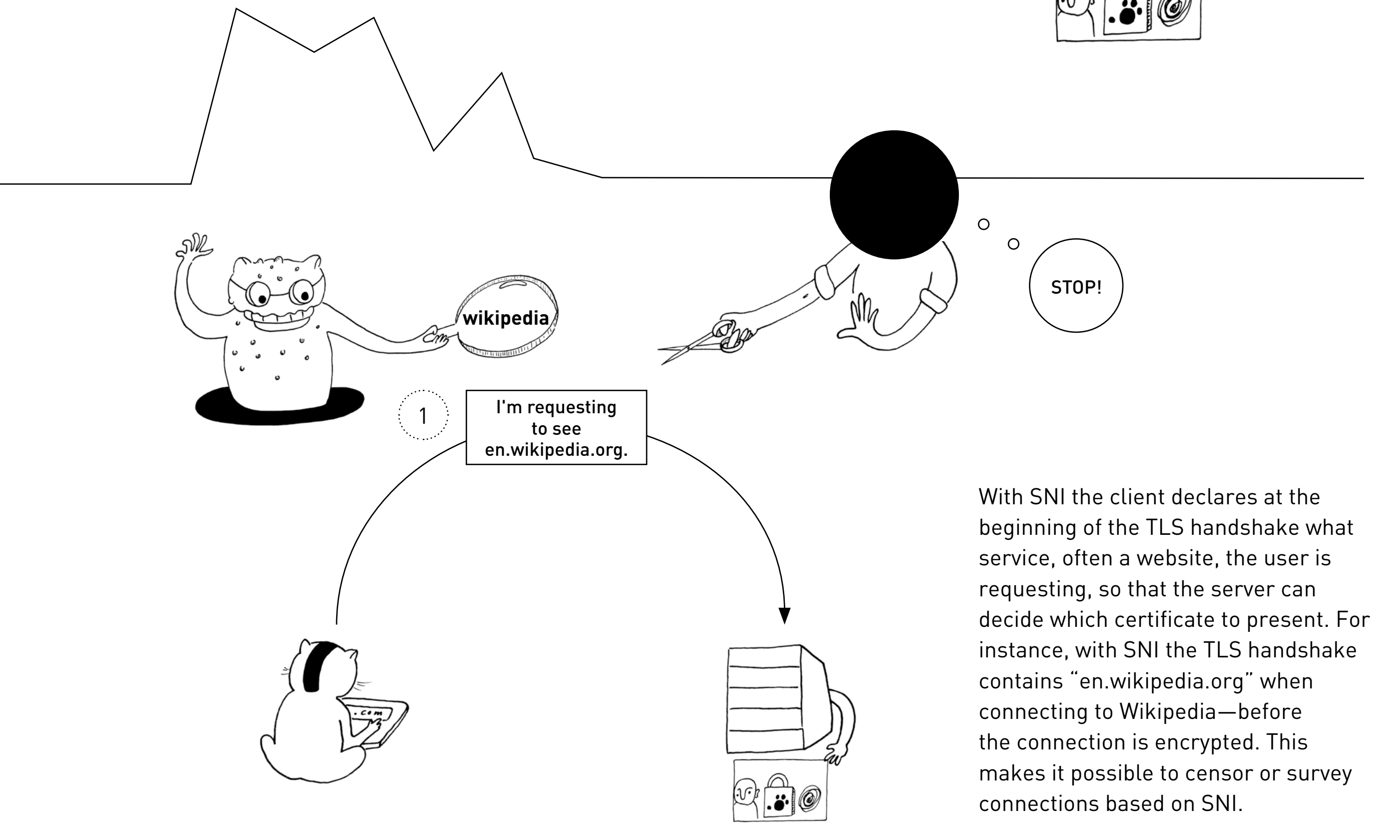
Server Name Indication

It is often the case that a server is hosting multiple websites or services under a single IP address.

Remember the TLS handshake? When using HTTPS (HTTP+TLS), to ensure that the connection is encrypted, client and server need to first accomplish the TLS handshake before they can initiate the HTTP session. But it's not until then that the server knows which website the client is requesting.

The **Server Name Indication (SNI)** is an extension to TLS that allows the client to signal to the server which specific hostname is being requested, in order to allow the server to present the correct certificate for that hostname.


1
list of supported encryption algorithms by priority, public key
+
requested hostname
.com
server
CERTIFICATE
CERTIFICATE
CERTIFICATE



With SNI the client declares at the beginning of the TLS handshake what service, often a website, the user is requesting, so that the server can decide which certificate to present. For instance, with SNI the TLS handshake contains "en.wikipedia.org" when connecting to Wikipedia—before the connection is encrypted. This makes it possible to censor or survey connections based on SNI.

Cryptography

We know from the Snowden leaks that governments and private companies work together to collect and share information about what users do when they're on the internet. The early internet was designed to be completely open and interoperable, without much thought to how this might put users of a global, ubiquitous network at risk. Internet engineers aimed early security efforts at hardening servers and internet infrastructure, not protecting user privacy from threats such as mass surveillance or targeted attacks. Indeed there's an ongoing tension between securing services and protecting user privacy.

Cryptographic Techniques

To secure communication, we can use **cryptography**. Generally there are two cryptographic techniques: signing and encryption.

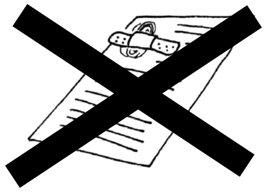
Signing Data

One technique to make sure that data is authentic is to cryptographically sign the data, which is like using one's unique handwritten signature for verification, but with math. This is called **authentication**.

We use the signing data mechanism in many cryptographic procedures where people or machines communicate with each other and want to authenticate data they send or receive. We can cryptographically sign all kinds of data: messages, emails, packets, and even signatures themselves (one signature authenticating another signature).

To authenticate a message or a file, Alice signs the data using her signature by adding a digital fingerprint.

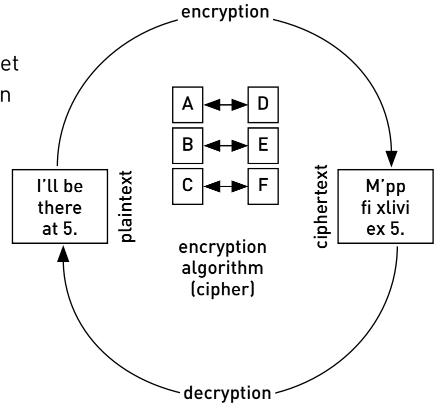
The recipient has a copy of Alice's fingerprint and can compare it to the original.



If the fingerprint has been destroyed or otherwise altered from the original, the recipient can't be sure that the data is authentic.

Encryption

Another cryptographic technique is **encryption**, or the art of writing secret messages. Encryption is an operation whereby we transform human readable plaintext into encrypted text, also called ciphertext. Inversely, transforming ciphertext into plaintext is called **decryption**. The process of encrypting and decrypting a message is called **encryption algorithm** or **cipher**.



Simple encryption algorithms use a mechanism that shifts each letter of the plaintext by a certain number of positions in the alphabet and replaces the original letter by the one obtained through the shift. The shift parameter is what we call a **key**. We need to know the key to perform the inverse operation.

ABCDEFGHIJKLMNOPQRSTUVWXYZ
↕ shift by 4 letters[10] = key
DEFGHIJKLMNOPQRSTUVWXYZABC

If an adversary can guess the key, it is easy for them to decrypt a message. Guessing possible keys can be automated using a computer, until the guess is correct and the key known.

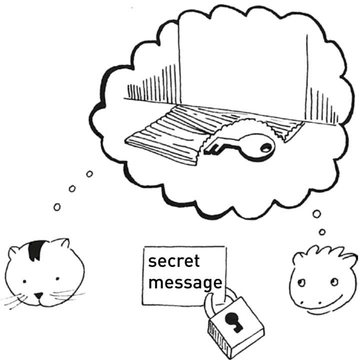
That's why modern encryption algorithms use keys that rely on complex mathematical problems that would take a computer many years to guess.

For example, it is easy to multiply two prime numbers with each other (97 x 13,395 = 1,299,315), but it takes a long time to find the two original prime numbers from the result of the calculation (1,299,315 / *x* = *y*).

There are two types of modern encryption algorithms: asymmetric and symmetric key algorithms. Some encryption algorithms also combine these two types and add additional features.

Symmetric Cryptography

Symmetric cryptography is a mechanism in which the sender and receiver have a copy of the same secret key needed to open a lock.



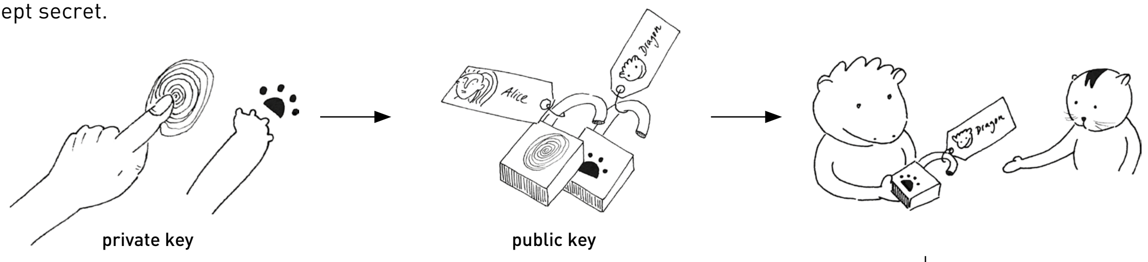
Asymmetric Cryptography

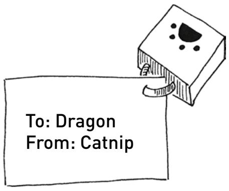
Asymmetric cryptography is a mechanism that always has two types of keys: a private key and a public key. We call this mechanism asymmetrical or **public-key cryptography** because only one party has a private key in their possession.

The **private key** is unique. Only its owner can use it for signing and decrypting data, and it must therefore be kept secret.

The owner can copy the **public key** and give it to those who need to send them encrypted data or messages.

When Catnip wants to encrypt data so that only Dragon can read it, Catnip first needs a copy of Dragon's public key.

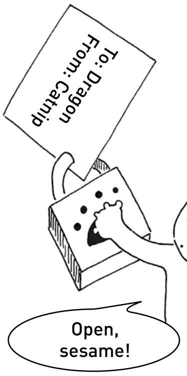




Catnip puts all the data to send to Dragon into an envelope and then closes it with Dragon's public key, the lock.

Catnip can get as many copies of Dragon's public key as needed. Only Dragon will be able to open the locks using his private key.

In some applications using asymmetric cryptography, the receiver is also required to use a **passphrase** on top of using the private key. This ensures that nobody else can use the private key even if they found it somewhere.



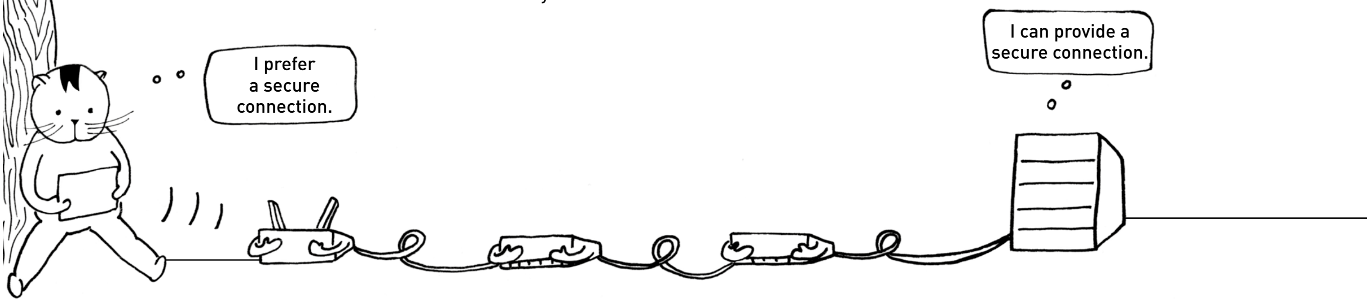
We need encryption and signatures for our data packet exchanges to ensure confidentiality, integrity, and authenticity of the packet data itself as well as for their transport.

Transport Encryption

Transport encryption ensures that our packets travel from one place to another through an encrypted connection, so that no third party can spy on it.

To encrypt network traffic we generally use the cryptographic protocol Transport Layer Security (TLS). TLS encrypts only the connection between the end-point nodes, even if other nodes on the network relay these connections.

It's up to the server or service provider to provide their users with TLS; it's not anything a user can decide to add themselves.

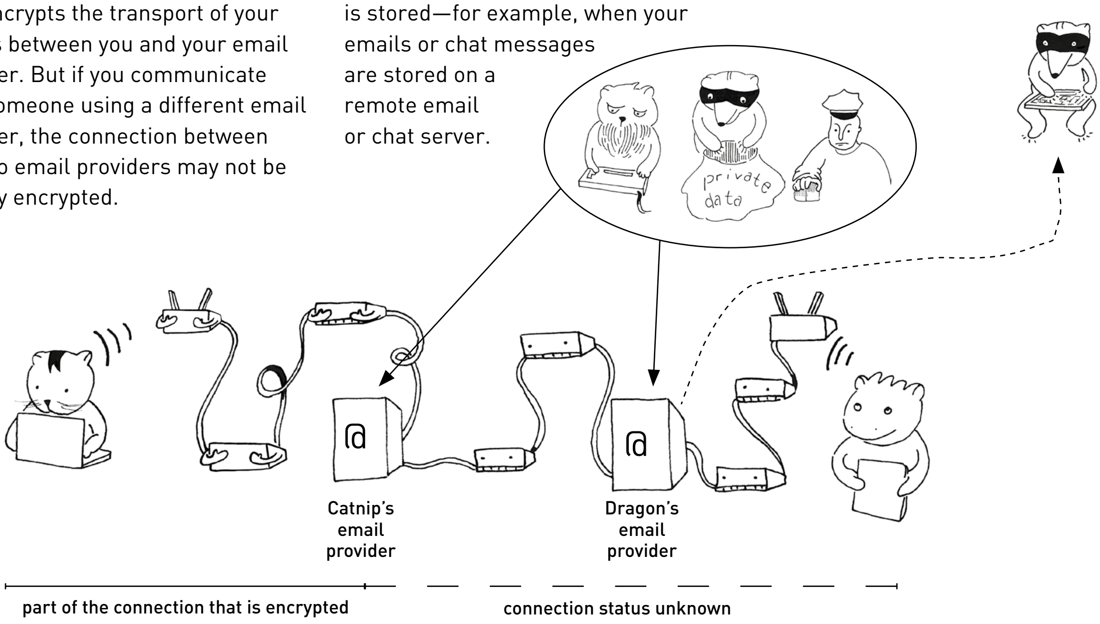

I prefer a secure connection.
I can provide a secure connection.

Limitations of Transport Encryption

Besides the risk of accepting fake certificates from compromised certificate authorities (CA), there are other limitations of transport encryption. One such limitation arises when the sender is not able to control the entire connection. For example, TLS encrypts the transport of your emails between you and your email provider. But if you communicate with someone using a different email provider, the connection between the two email providers may not be equally encrypted.

Another limitation is the storage of data. TLS encrypts the connection that transports the data but does not encrypt the data itself. Thus after the data is transported, server administrators, law enforcement, or an attacker could access it, once it is stored—for example, when your emails or chat messages are stored on a remote email or chat server.

Another limitation is the scenario in which an attacker could use a machine-in-the-middle attack to pretend they're actually the receiving end of your transport connection and read the data in the packets. We'll explain this at the end of this chapter.


private data
@
@
Catnip's email provider
Dragon's email provider
part of the connection that is encrypted
connection status unknown

End-to-End Encryption

Though transport encryption is helpful, it still leaves us open to two vulnerable scenarios:

1) When someone observes our network traffic, though they can't read the content of our data packets, they might learn our whereabouts as well as who is communicating with whom.

2) An attacker, a server administrator, or law enforcement requesting access could read the data stored on the server of the intermediary, such as your email host. For example, when you send an email to a friend, the email lies around on your mail server and your friend's mail server, where an attacker could read it.

To protect ourselves from the first scenario, we need anonymity. We talk about that in Chapter 7. To protect ourselves from the second scenario, we can use privacy-enhancing applications that encrypt the data exchanged between devices, for example of sender and receiver. This is also known as **end-to-end encryption**.

Here are two examples of the more common end-to-end encryption algorithms.

Double Ratchet Algorithm

Used in many modern messenger apps, such as Signal.

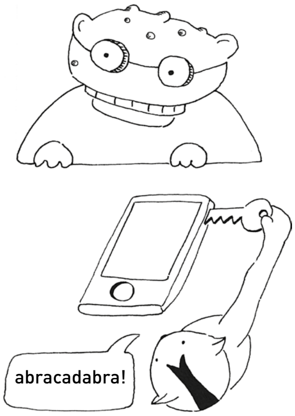
OpenPGP/GPG

Mostly used for email. Also used to encrypt files and folders. When we use OpenPGP to encrypt email, we generally use asymmetric encryption and have to exchange public keys with our communication partners.

Encrypting Data at Rest

When we talk about the internet, we are usually talking about data in transit. But there are good reasons to also think about data when it is not traveling, especially if it is sitting on a server that you don't control, for example "the cloud." Also on your own device: keeping data secret when it is at rest is a common use of encryption.

Cryptographic schemes for encoding and decoding work in various ways, but they essentially all use a key to transform an input into a ciphered output, and sometimes vice versa. To make sure no one but you can read a piece of data, you can use encryption software that generates a unique key that usually requires a password. You can use that same key to decrypt the data when you need to.

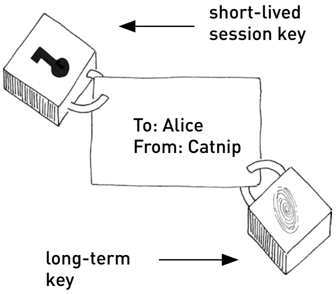


Forward Secrecy

Many modern encryption algorithms feature **forward secrecy**, which ensures that intercepted and stored encrypted packets can't be decrypted in the future, even if the long-term encryption keys of the sender and receiver are compromised.

To achieve forward secrecy, we generate one or several disassociated, short-lived, random session keys for each communication or packet exchange. These keys are good only for the moment of transport, and the encrypted message cannot be decrypted later on, even with these keys.

Forward secrecy is a feature of IPSec, TLS version 1.3, **Off-the-Record (OTR)** chat, Double Ratchet Algorithm, Secure Shell (SSH), and Tor (discussed in Chapter 7).



Limiting Encryption

Cryptography is the study of and process for making encryption and decryption mathematically secure. Despite the absolute certainty of math and numbers, policy regulations, the technical weaknesses of the protocols themselves, programming mistakes, or increases in computational power can limit the security and privacy provided by cryptography.

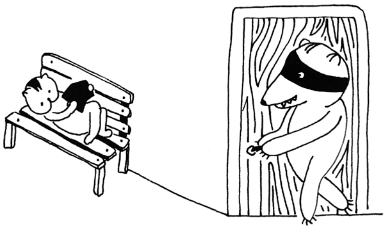
An adversary can try to break encryption and guess the cipher in less time by using more powerful computers.[11]

Several countries have legal restrictions on the domestic use of encryption. More common still are restrictions on the export or import of encryption technologies.

Other tactics focus on weakening the cryptographic protocol. For example, the National Security Agency (NSA) has tried to interfere with encryption protocols and algorithms by trying to make weak encryption a standard.[12] Schemes to weaken encryption are often referred to as **backdoors** because they attempt to get around the strength of cryptographic keys to the proverbial "front door."

The National Institute of Standards and Technology (NIST), the body that sets US national standards, created a backdoor by standardizing a weak algorithm of elliptic curve cryptography.[13]

There are also allegations that the NSA tried to create a backdoor in the IPSec protocol.[14] We know all this from the Snowden leaks.

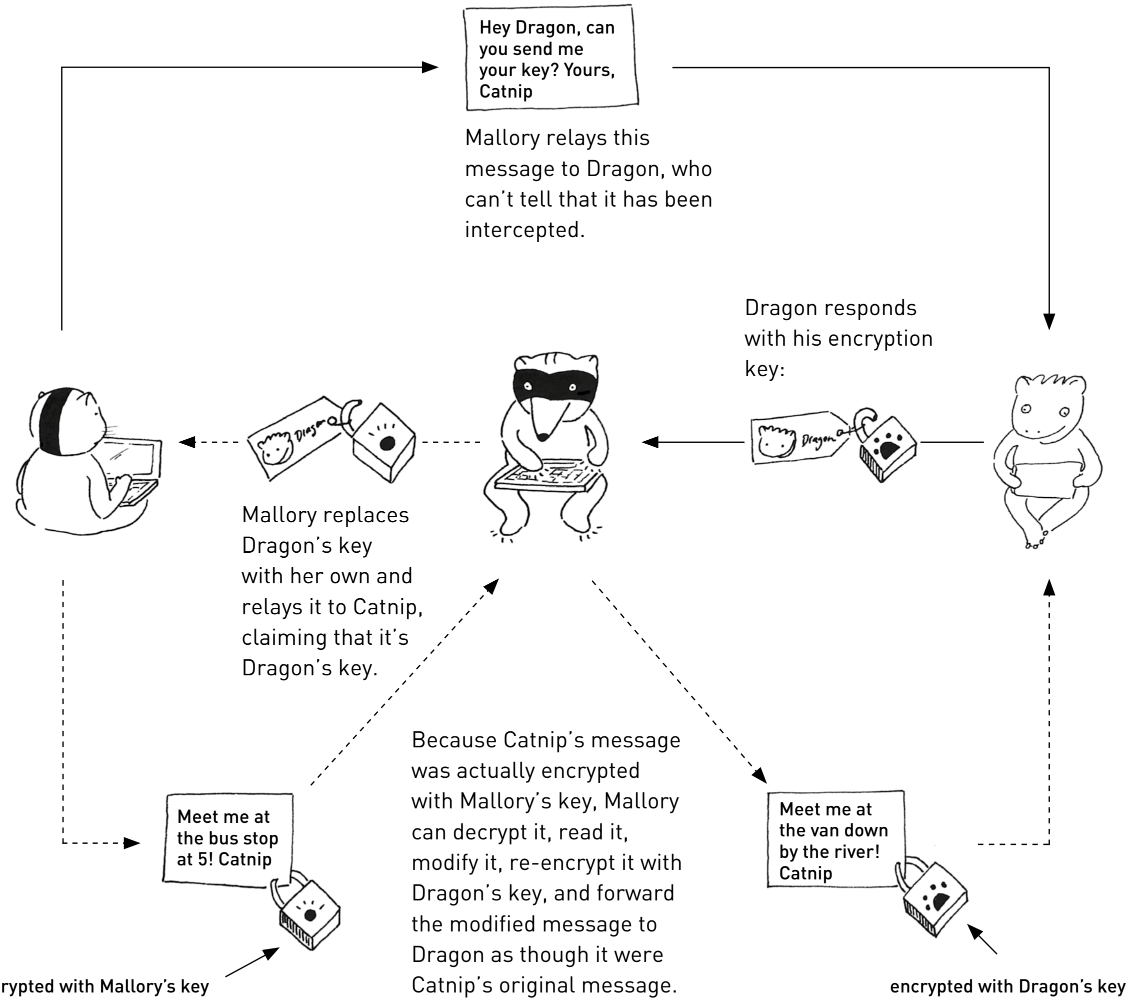


States, banks, and other organizations have reportedly tried to convince software developers to build in backdoors by providing so-called **golden keys**, which would open every lock. Third-party key escrow is another idea with the same risks as backdoor access and weakening encryption. Companies and software developers should refrain from all these methods that weaken encryption—they not only violate user privacy and trust but also compromise user security.

Machine-in-the-Middle

Internet protocols operate in a way that's hidden from most users. An **on-path attacker**, also known as a **man-in-the-middle** or **machine-in-the-middle (MITM)**, exploits these hidden protocol operations to confuse user devices.

Here's the basic idea behind a MITM attack. Catnip sends a message to Dragon (friend), which Mallory (attacker) intercepts:



Hey Dragon, can you send me your key? Yours, Catnip
Dragon
Dragon
Meet me at the bus stop at 5! Catnip
encrypted with Mallory's key
Meet me at the van down by the river! Catnip
encrypted with Dragon's key

Mallory relays this message to Dragon, who can't tell that it has been intercepted.

Dragon responds with his encryption key:

Mallory replaces Dragon's key with her own and relays it to Catnip, claiming that it's Dragon's key.

Because Catnip's message was actually encrypted with Mallory's key, Mallory can decrypt it, read it, modify it, re-encrypt it with Dragon's key, and forward the modified message to Dragon as though it were Catnip's original message.

In the case of public-key cryptography as used in email or chat applications, users can protect themselves from a MITM by verifying the keys of their communication partners prior to a first message exchange.

In the case of a MITM attack on transport encryption, for example when interacting with websites over HTTPS, a manual verification of the server certificate can in some cases protect the user.

Banking operations over the internet protect users by providing a second verification channel, for example by asking to confirm any operation using a **Mobile Transaction Number (mTAN)**. This is called **two-factor authentication (2FA)**.

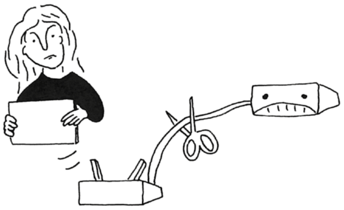
6

WHAT CAN INTERFERE WITH INFORMATION TRAVELING ACROSS THE INTERNET?

Censorship

Remember that networks can deliver packets as long as the packets have correct packet headers, regardless of the packets' contents. The network itself is content agnostic: it doesn't care about content, as long as the packets are routable. This is what we call **network neutrality**.

But sometimes states, institutions, parents, or other authorities want to prevent us from accessing certain content on the internet. **Censorship** methods include blocking, filtering, and throttling.



Blocking renders a website or service inaccessible for a specific set of users and is often put in place with the cooperation of local ISPs, who prevent their end users from being able to make a connection to the website or service.

Filtering is a more general approach to restricting content that seeks to prevent access based on defined characteristics about the content being accessed, such as the use of particular words or image content.

Throttling, also known as degraded or differential service, is used to make access to some services or websites very difficult, slow, or practically impossible for some users.

Those operating the routers, the servers, or the network equipment can filter traffic or block access at the source, during packet delivery, and at the destination.

The reasons for censorship may stem from differing views on morality, free speech, security, religion, politics, or economics.

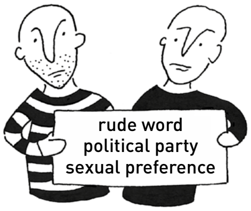
At the national level, ISPs can be forced to apply blocking and filtering to all traffic entering or exiting the country. A regional control of content could rely on the cooperation of several network nodes and can be enacted through autonomous systems. Institutions such as libraries, universities, workplaces, or internet cafes can also put in place blocking and filtering controls on content.

There are many techniques that can occur simultaneously to block and filter content:

IP Blocking

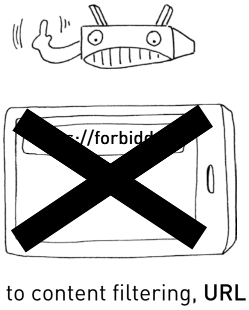
Authorities can block packets based on their source and destination IP addresses. This **IP blocking** technique can block whole ranges of IP addresses and cut off entire regions.

Content Filtering



Whoever controls the router can read the traffic passing through the router. They can read packet headers that include information about the websites we're trying to visit. If the connection isn't encrypted, they can even read the content of websites. This allows public spaces, parents, and routers at the ISP level to use a technique called **content filtering**, which filters all pages that contain certain words.

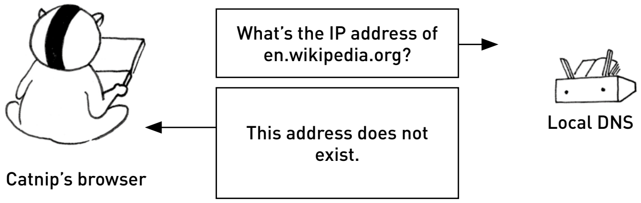
URL Filtering



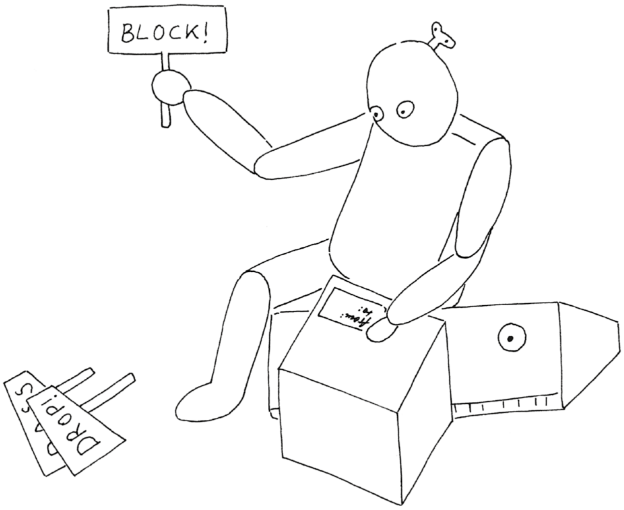
Similar to content filtering, **URL filtering** scans URLs for specified keywords and blocks them.

DNS Blocking

DNS blocking prevents DNS from resolving specified domain names. ISPs implement DNS blocks in the DNS resolvers they control. If a DNS block is in place, when you type a website's address to access it, the ISP's DNS resolver either pretends that it can't find the server, or it returns a different IP address, such as one that hosts a warning message. DNS blocking affects all protocols that rely on DNS, such as HTTP(S), FTP, POP, and SSH.


What's the IP address of en.wikipedia.org?
This address does not exist.
Catnip's browser
Local DNS

State governments very commonly force ISPs to implement DNS block lists to apply their legal and judicial decisions.[15]



Packet Filters

To read packet headers, routers or servers implement **packet filters**, which search for protocol noncompliance, viruses, spam, or intrusions and block outgoing or incoming packets accordingly.

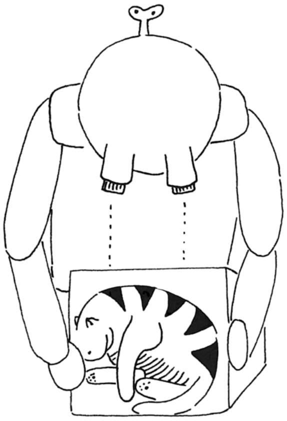
The filter decides if a packet may pass, if the router needs to route the packet to a different destination, or if the router should silently drop the packet.

Packet filters can also protect networks from attacks by filtering out packets that are aimed to attack servers or routers.

Deep Packet Inspection

Deep Packet Inspection (DPI) is similar to packet filtering, but instead of simply looking at packet headers, it also reads the data within the packets.

DPI is data-processing software that can be useful for seeing inside packets to identify, monitor, and troubleshoot network abnormalities, but routers and servers can also use it for data mining, eavesdropping, and internet censorship.

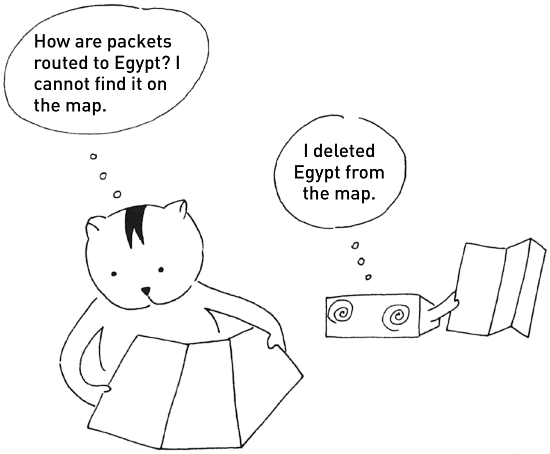


DPI can redirect, tag, block, rate limit, and report, or it can silently drop packets it marks as suspicious.

In order to inspect packets as they go through a key point in the network, DPI covertly and silently copies packets and analyzes them.

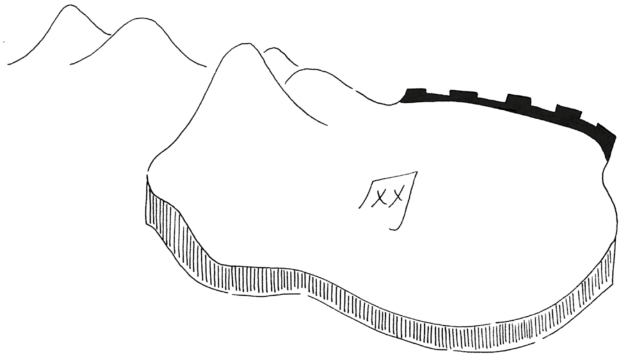
Network Shutdowns

States can easily shut down an entire network by manipulating BGP, the protocol that maps the internet. As we've seen in Chapter 4, BGP routing is simple and powerful, yet easy to get wrong. Operators of BGP routers or Autonomous Systems can publish wrong network routes or unpublish network routes to manipulate BGP in order to cut off entire parts of the internet.


How are packets routed to Egypt? I cannot find it on the map.
I deleted Egypt from the map.

Great Firewall of China

China has a largely state-controlled internet. China limits traffic to some foreign tools and services and forces foreign companies to adjust to national rules and regulations. This set of filters and blocks is known as the **Great Firewall of China (GFC)**. It combines many blocking techniques and man-in-the-middle attacks and is therefore very effective.



Content and Search Removal

Censorship techniques can be enacted at the content's source, not just over the network. Publishers, authors, and service providers have to comply with legitimate requests or applicable laws to take down, unpublish, unlist, or otherwise hide content when required by law or government requests.

Under European privacy law, web publishers also must allow individuals to exclude themselves from search results.

When publishers censor content in these ways, they can pretend that they can't find the content, or, for more transparency, can tell the user that the content exists but that it's blocked—for example, by using the HTTP error status code 451.

The status code 451 is a reference to the Ray Bradbury novel *Fahrenheit 451* and is used when content cannot be served for legal reasons.

7

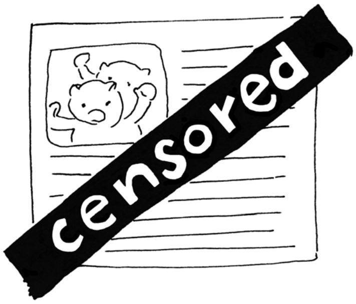
HOW CAN INFORMATION TRAVEL ANONYMOUSLY OVER THE INTERNET?

Now that we've discussed the different ways governments, companies, and organizations censor content on the internet, let's discuss how we can overcome censorship. In this chapter we'll focus on how we find out what is being censored, known as censorship monitoring, and how we can circumvent censorship.

Censorship Monitoring

In order to overcome censorship, we first have to know that it's happening. We can distinguish censorship from temporary outages by globally monitoring internet connectivity. We can conduct **censorship monitoring** in a variety of ways. User reports can reveal whether content has been made inaccessible. Some governments and companies are transparent about the censorship that they enact. Often, though, we can know definitively if internet censorship is happening on a technical level only by testing different types of requests to servers and services from specific locations. We analyze the responses or results of those requests against the responses of requests received on veritably healthy, uncensored connections to see if there are differences between the responses. These tests aim to eliminate the possibility that a response error (such as a 404, explained in Chapter 4) is due to a block or filter in the network and not some other reason originating with the service being requested.

There are several organizations and research projects that try to monitor internet censorship.



Netblocks

→ *https://netblocks.org*

NetBlocks is a civil society group that creates measurements and data visualization tools to help diagnose internet shutdowns, telecommunications blackouts, and politically or economically motivated online censorship.

On top of making censorship visible, their tools are effective at mapping outages due to natural disasters such as earthquakes or hurricanes, as well as cyber attacks on network infrastructure. NetBlocks publishes reports about internet shutdowns worldwide, along with explanations and real-time updates.

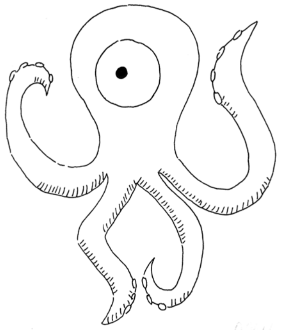
Open Observatory of Network Interference (OONI)

→ *https://explorer.ooni.org*

Open Observatory of Network Interference (OONI) is a global and decentralized observation network for detecting censorship, surveillance, and traffic manipulation on the internet. OONI is free software, so anyone can run an OONI probe to test connections to websites that may be banned. Measurements are published in the OONI Explorer.

Anyone can run an OONI probe to test:

- whether websites are blocked
- whether instant messaging apps (such as WhatsApp and Facebook Messenger) are blocked
- whether censorship circumvention tools (such as Tor) are blocked
- the presence of systems ("middleboxes") in your network that might be responsible for censorship and/or surveillance
- your network's speed and performance



OONI does not protect the privacy of those running probes, and measurements can include identifying information.

Transparency Reports

Companies publish **transparency reports** that detail the nature of demands for censorship from governments, copyright owners, or others and the company's compliance. Transparency reports may also include statistics on requests for user data.

Google, for example, provides anonymized information of requests to unlist search results and content on other Google products such as YouTube or Blogger. Many of these requests are due to legitimate copyright claims, as well as governmental and individual removal requests related to concerns of national security, defamation, privacy and security, drug abuse, or obscenity and nudity.

Google generally classifies the censorship requests they receive into four different categories:

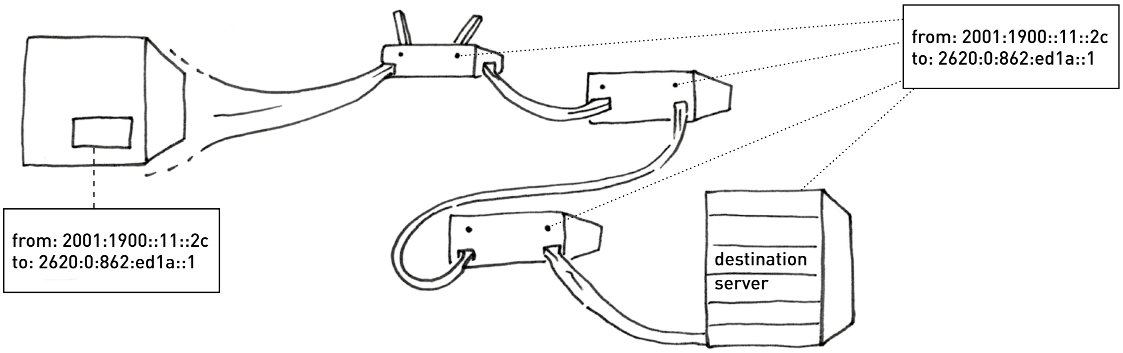
Protected—Google automatically filters and blocks tens of thousands of URLs per week from search results to try to protect users from websites with malware and phishing.[16]

Removed—Requests to remove content because of copyright issues. More than 20,000 private companies and copyright owners have requested Google to remove 4,683,688,889 URLs in total.[17]

Hidden—Search results removed according to privacy laws. Google receives thousands of requests to delist URLs per week; of the requested URLs, Google delists 46 percent.[18]

Censored—Government requests to remove content. Google received about 30,000 within the year 2019.[19]

How Data Travels



Before talking about censorship circumvention, let's quickly review the way our data travels.

Data travels in pieces called **packets**. Each packet has an **address tag**, or **packet header**, that indicates its source and destination address.

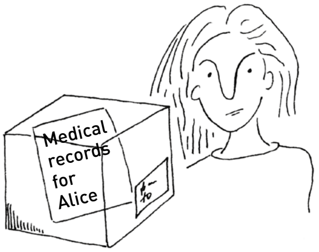
There are no direct connections on the internet. Packets travel through intermediary networks and routers that read the packet header to route the packets to their destination.

By reading the packet tag, these intermediary networks know where they received the packet from and where they're sending it to.

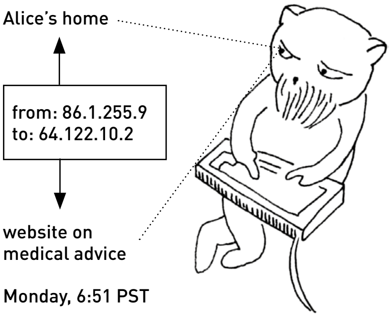
All of these intermediaries can copy, store, or even alter packets.

Anonymity and Pseudonymity

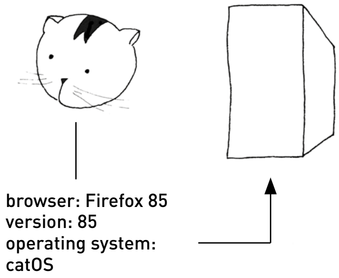
When we send packets without using end-to-end encryption, our packets might contain unencrypted content that gives away information about our real-world identity.



Also, the packets' metadata contains information indicating our real-world location and interests, such as source and destination IP address, that websites and ISPs can easily track.

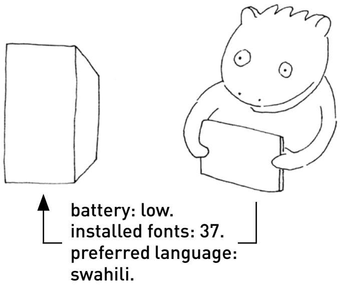


When we visit websites, we send and receive packets that contain more of our information that can be read, extracted, and further analyzed, including the operating system and browser versions used.

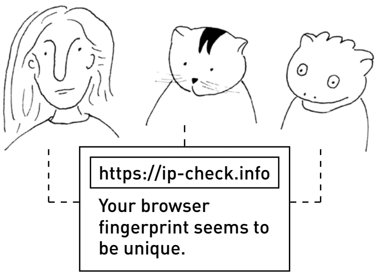


Website operators can extract even more identifying details from our packets when we visit websites programmed with recent web technologies such as JavaScript and HTML5. Operators can know the configured language the browser is using, which fonts are installed on our computer, what screen resolution we're using, and even the battery status of our device.

We call tracking and extracting this kind of information **fingerprinting** because this information is usually unique to a single user.

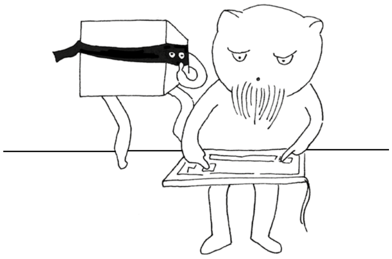


Companies and institutions can use fingerprinting to find out our IP address and other metadata to pinpoint exactly who we are.



When we talk about **anonymity** on the internet, we're talking about the concealment, or linkability, of identifying information. Even when we use applications and services that protect our personally identifiable information, such as transport or content encryption, others can track our packets' metadata, such as our IP address. So we can only be **pseudonymous** online, not anonymous.

To become anonymous, we need to use techniques that conceal our IP information and, therefore, our location.



Censorship Circumvention

Intermediaries can easily alter packets, which becomes a problem when states, corporations, employers, parents, or network operators try to stop you from accessing certain content on the internet.

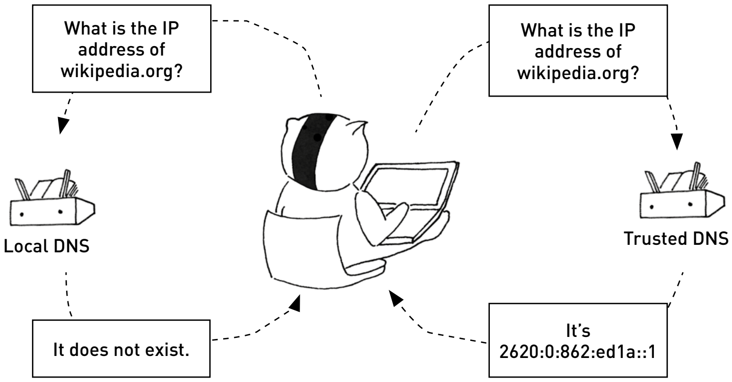
However, attempts to censor content can still happen at any stage: at the source, at intermediate routers, or at the destination. Various parties that operate the network carry out these interventions, which can come in the form of blocking, filtering, or throttling, which are explained in Chapter 6.

There are many good reasons why internet users might want to circumvent censorship or protect their personal data, privacy, anonymity, or pseudonymity online.

However, we'll focus more on the technical implementation of censorship and the ways we can circumvent, or counter, censorship in the form of filtering and blocking.

DNS Proxy

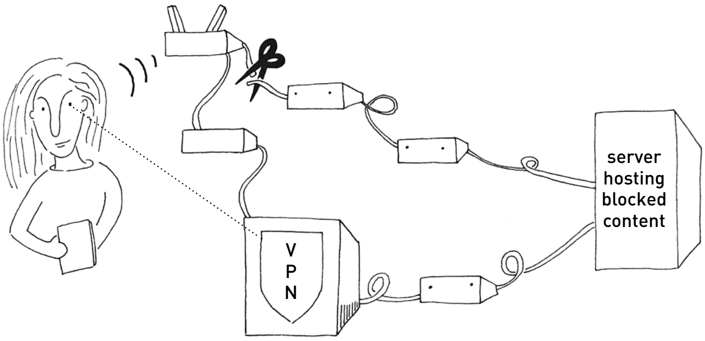
To counter DNS blocking, you can use a DNS server that you trust instead of the one automatically provided by ISPs. Using a **DNS proxy** allows you to bypass DNS filtering or blocking that may be put in place at the level of a local or national ISP.


What is the IP address of wikipedia.org?
What is the IP address of wikipedia.org?
Local DNS
Trusted DNS
It does not exist.
It's 2620:0:862:ed1a::1

Virtual Private Network

To counter surveillance or censorship at your workplace or university, you can connect to a **virtual private network (VPN)** or a **proxy**, which conceals your network traffic and makes and receives DNS requests on your behalf. The VPN provider knows who you are, so a VPN doesn't provide total anonymity.

The VPN provider itself or a strong external adversary can still link your incoming and outgoing traffic to identify you, so make sure you trust your VPN. A VPN simply shifts the burden of protecting your identifying information from you to the VPN provider.



In the United States, circumventing an IP block in order to access a website (for example, using anonymous proxies) is a violation of the Computer Fraud and Abuse Act (CFAA), punishable by civil damages or even jail time for "unauthorized access."[20]

Using Tor to Avoid Censorship

To circumvent censorship, we can also use the **Tor network** to hide our source and destination addresses to anonymize internet traffic. ("Tor" is derived from the original software project name, **The Onion Router**.) Tor is made up of volunteers who route peer user traffic, creating anonymity through "cooperative obfuscation."[21]

How the Tor Network Works

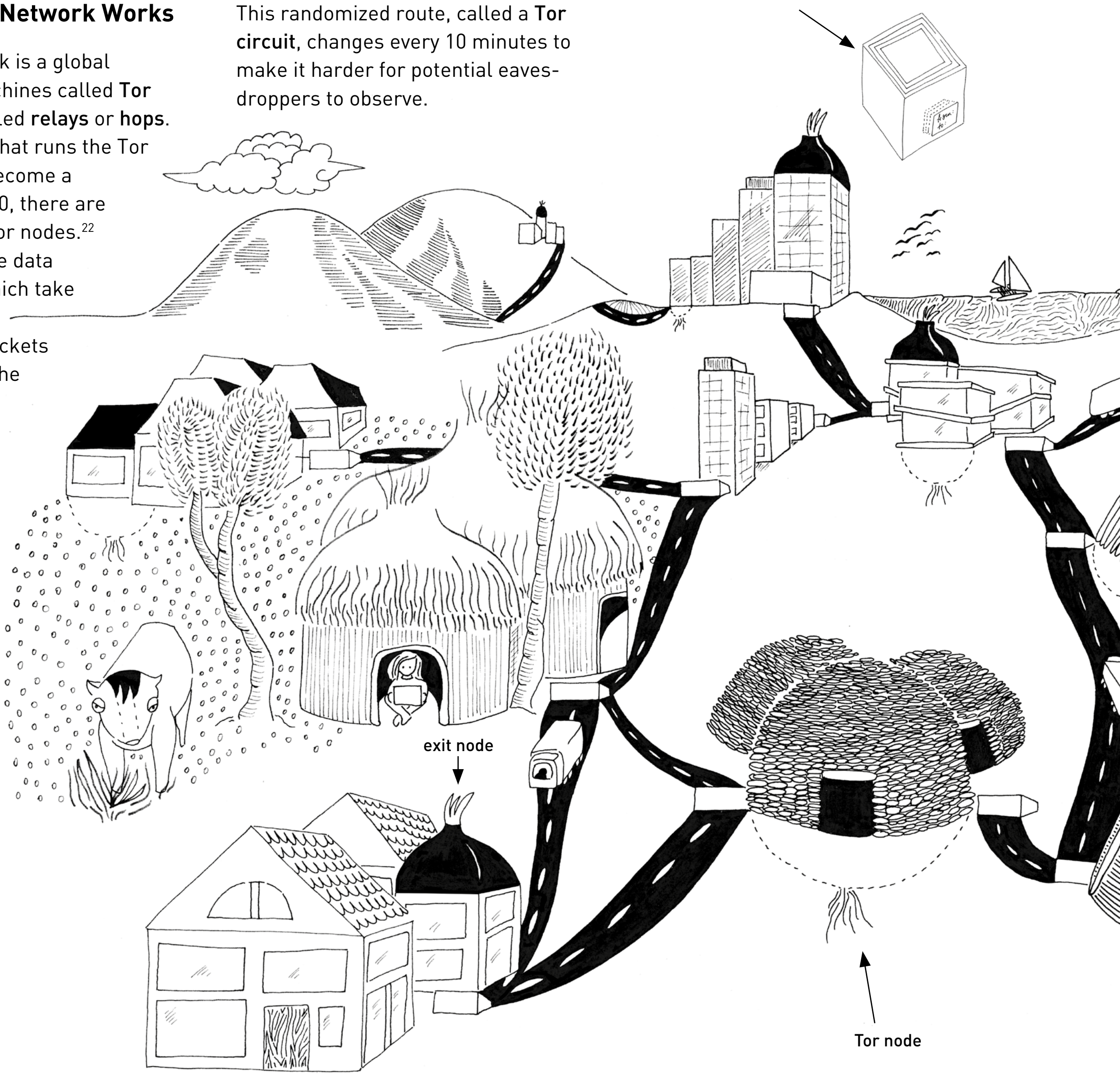
The Tor network is a global network of machines called **Tor nodes**, also called **relays** or **hops**. Any computer that runs the Tor software can become a node. As of 2020, there are around 6,500 Tor nodes.[22] Nodes work like data checkpoints,which take in, treat, and ship out packets traveling over the Tor network.

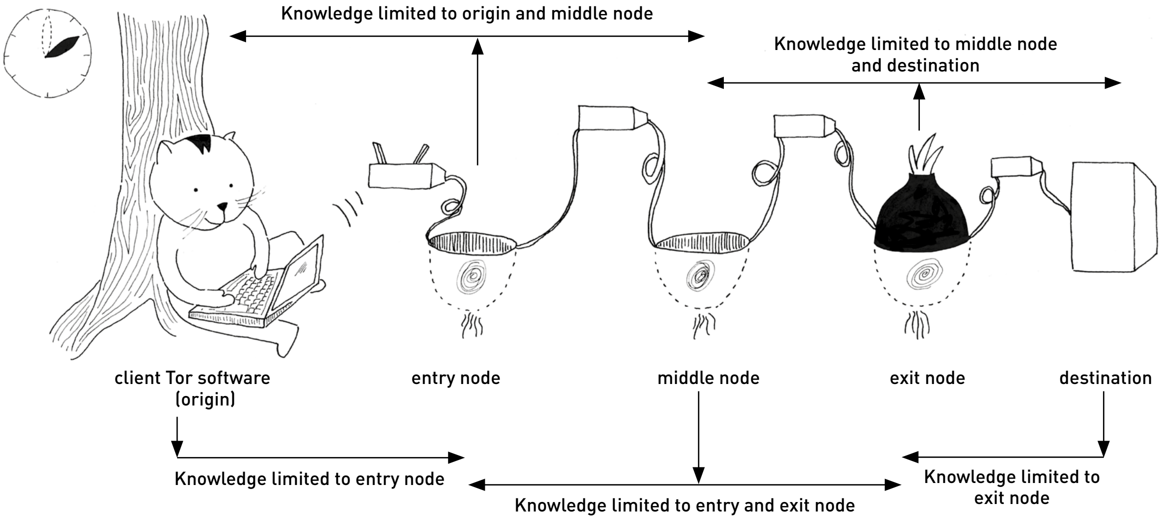
Tor Circuit

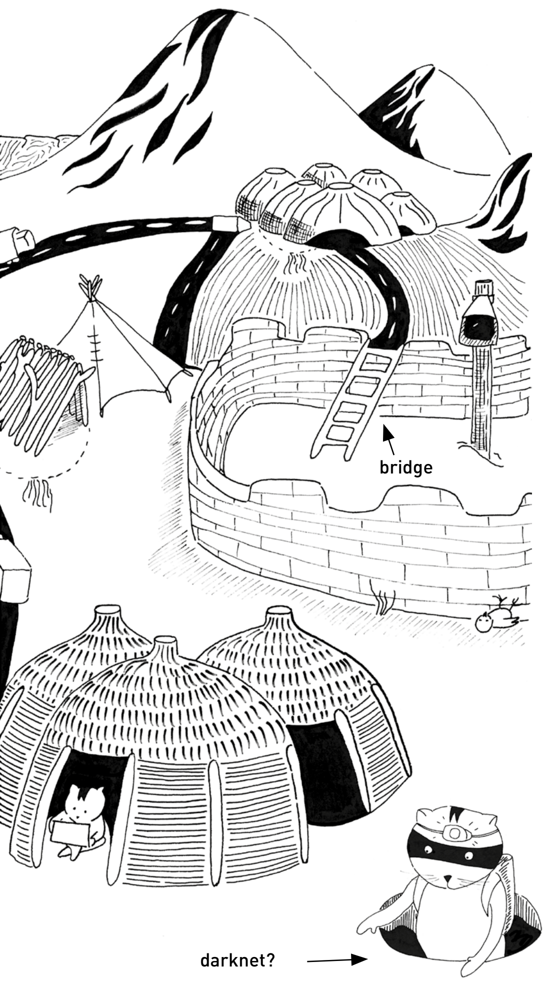
Tor nodes use the same infrastructure that the internet uses to send data packets over the Tor network. However, packets traveling through the Tor network are routed randomly through three different nodes before reaching their final destination. This is so that no single node can know both the origin and destination addresses. This randomized route, called a **Tor circuit**, changes every 10 minutes to make it harder for potential eavesdroppers to observe.

To randomize the path, Tor wraps data packets in three encrypted layers, like an onion.

Each layer contains a dedicated **packet tag**. By encrypting the packet layers, Tor ensures that only a specific node can unwrap its corresponding layer.




Knowledge limited to origin and middle node
Knowledge limited to middle node and destination
client Tor software (origin)
entry node
middle node
exit node
destination
Knowledge limited to entry node
Knowledge limited to entry and exit node
Knowledge limited to exit node



These dedicated packet tags only contain a partial route, meaning none of the three nodes knows the packet's entire path.

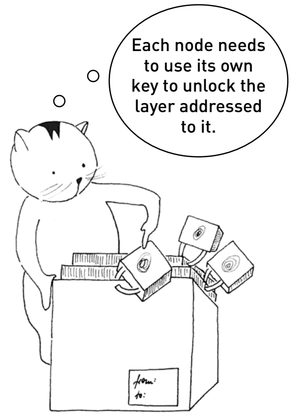
Each node peels off a single layer and sends the packet to the node written on the tag of the newly exposed layer. To anyone without full knowledge of the Tor circuit, it just seems like obscure cargo being sent from place to place.

This obscurity is why the Tor network is often pejoratively called the **darknet** (contrasted with the "clearnet"). In truth, there are many darknets, and most parties who use them simply value the privacy they provide.

Because each intermediary node knows only the location of the node preceding it and the location of the next node, the packet's sender remains anonymous.

Packets destined for a website or email server on the clearnet leave the Tor network after the last node. To an outside observer, these packets seem to originate from this **exit node.**

As of this writing, there are about 1,200 exit nodes on the Tor network.[23]


Each node needs to use its own key to unlock the layer addressed to it.
from:
to:

It's important to note that if the initial request wasn't encrypted using transport or end-to-end encryption, the package's contents would be visible to the exit node.

Blocking Tor

On some networks, known entrances to the Tor network are censored. When entrances are blocked, you can use a bridge as an entry point. A **bridge** is a Tor node that isn't listed in the public Tor directory. Note that if a censor learns of a bridge address, the censor can block it.

Censors can also block exit nodes, preventing any packets sent over the Tor network from reaching websites or servers on the clearnet.

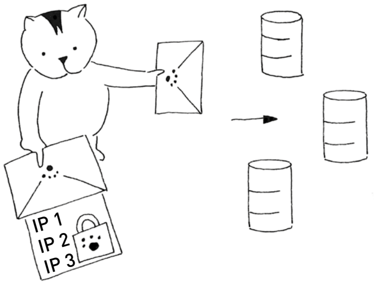
Onion Services

The Tor network provides services that keep traffic end-to-end encrypted. Like Tor end users, an **onion service** is hidden within the network, obscuring the server's location. These services use the top-level domain **.onion**. Let's walk through an example of how a **.onion** service works.

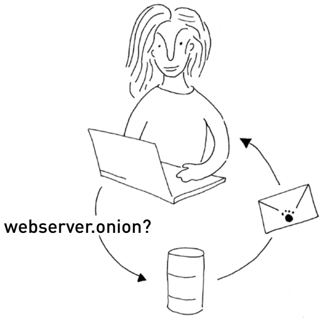
Say Catnip proposes an onion service for a web server, webserver.onion.

Catnip's Tor software randomly picks three nodes, builds a Tor circuit out of them, and makes them **introduction points**.

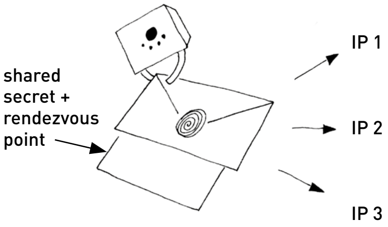
Catnip's Tor software then prepares a **descriptor message** that contains the names of the introduction points and the public key belonging to webserver.onion. It signs the descriptor message and sends it to a (distributed) database within the Tor network.



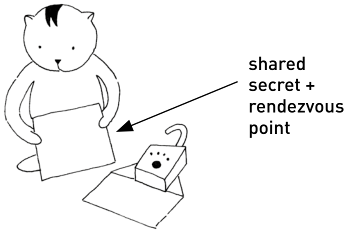
Tor user Alice has read about webserver.onion and wants to visit it. Her Tor software sets up a **rendezvous point** within the network and then asks the database for a signed message from webserver.onion.



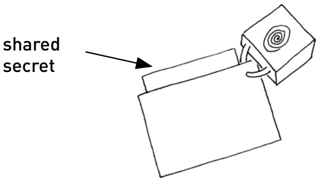
When she receives the message, her software creates an **introduction message** which contains a secret message encrypted with webserver.onion's public key. She then sends the introduction message to one of the introduction points, which relays the message to Catnip.

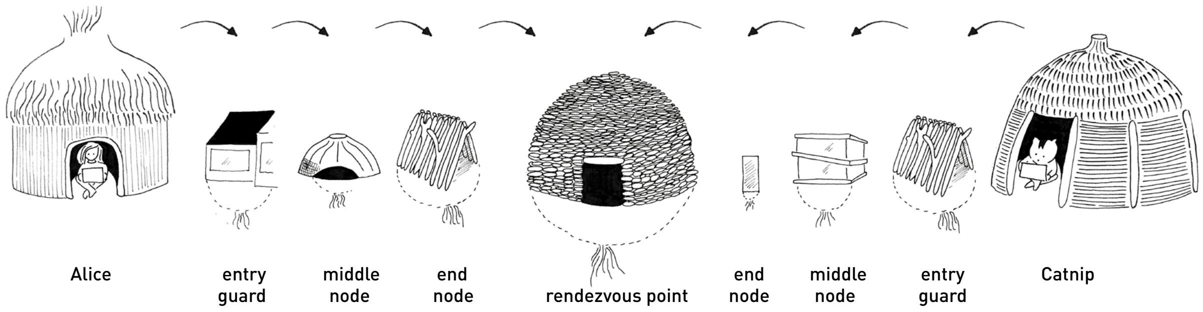


The message arrives to Catnip through a Tor circuit of three nodes, none of which knows the entire path, keeping both Catnip and Alice anonymous. Catnip's onion service then decrypts Alice's introduction message and finds the address of the rendezvous point and the secret message.



Catnip connects to the rendezvous point and includes Alice's secret message in their first rendezvous message. The rendezvous point informs Alice's Tor client when this happens. Now Alice and Catnip can communicate by using their own three-node Tor circuit to the rendezvous point, which relays their end-to-end encrypted messages.





Limitations of Tor

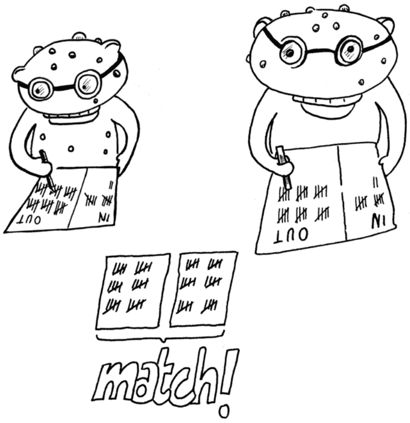
Because Tor currently uses TCP as its packet transport protocol, applications based on UDP, such as video calls and file sharing through torrents, don't work over Tor. All UDP packets are rejected when using Tor.

Especially because Tor is a privacy and anonymity-enhancing tool, you should have a clear understanding about what Tor can't do or when it won't protect your identity or your data.

First, keep in mind that anyone running the Tor software on their server can choose to become a node in the Tor network. This means that there might be eavesdroppers or other malicious entities running Tor nodes. If the same entity controls more than one node in a Tor circuit delivering your packets, it could know enough about you and your traffic that your anonymity would be broken.

Having access to an exit node can sometimes be enough for an eavesdropper to know who you are if you log into a website without using encryption (such as HTTPS), for example. It's critical that you always use encryption alongside Tor.

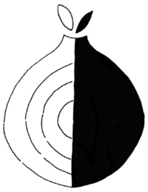
Lastly, particularly savvy malicious parties can infer the contents of encrypted packets by closely monitoring the size, timing, and quantity of ingoing and outgoing packets at both the origin and destination. This is called **targeted traffic analysis**, and Tor can't protect you against such attacks.



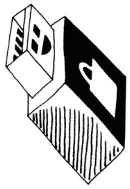
Using the Tor Network

You can access Tor two ways. Either you can manually configure a device or router to send and receive traffic through the Tor network, or you can use one of several software applications:

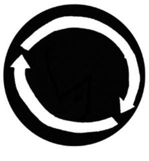
TorBrowser prevents others from tracking you as you browse the web or the dark web.



TorBrowser and Orbot let you use Tor on an Android smartphone.



Tails is a live operating system that makes all network traffic automatically go through the Tor network.



OnionShare lets you share files anonymously.

8

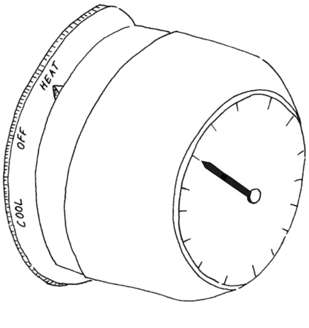
WHAT CONTROL DO MACHINES HAVE?

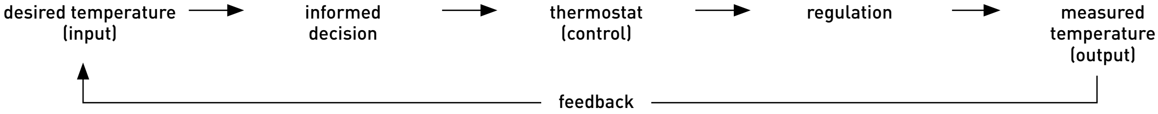
We have explained the standard ways in which devices on a network communicate with one another. It's time to go deeper into how those devices themselves operate. In this chapter we will explain computer decision making, specifically cybernetics, algorithms, and automation.

Cybernetics

In 1948, Norbert Wiener coined the notion of cybernetics as the art of governing and communication. **Cybernetic systems** are systems that use data communicated within the systems to regulate and optimize themselves.[24]

An example of a simple cybernetic system is a heater thermostat. If the temperature drops below a certain temperature, it should automatically turn on. When the temperature rises above a certain temperature, it should automatically turn off. The user can change the settings so that the heater aims to keep a certain temperature.



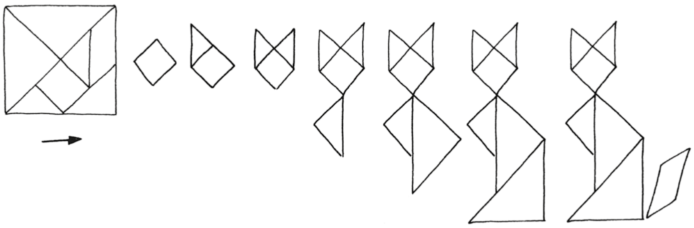


Algorithms

An **algorithm** is an unambiguous set of instructions describing how to solve a problem or a class of problems.

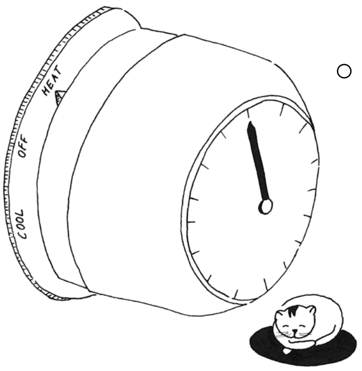
Algorithms represent a core feature of today's and tomorrow's informational ecosystems.

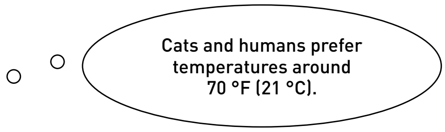
The word comes from a cacography of the name of Iranian mathematician and astronomer Muhammad Ibn-Mūsā al-Hwārizmī.



Software Algorithms

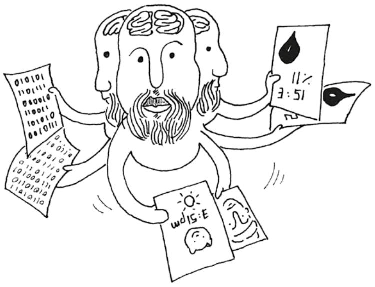
Algorithms are an important part of software in computer systems. **Software algorithms** follow logical steps, and—similar to cybernetic systems—aim to produce an output based on an input. You learned basic algorithms in mathematics to solve a problem like $x + 2 = y$. When the input, x, is 40 then the output, y, is 42. The quest to solve more complex problems with computing requires more complex algorithms and more input data.





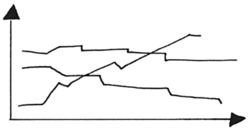
The heater thermostat adjusts temperatures according to pre-obtained data and the knowledge that humans and cats express discomfort above certain temperatures.

In the era of big data, companies and organizations use algorithms to search through huge amounts of data. They assign levels of relevance to each piece of data and classify them to produce valuable information.



Based on aggregated and evaluated data, algorithms can:

- identify patterns and trends
- cluster similar data points



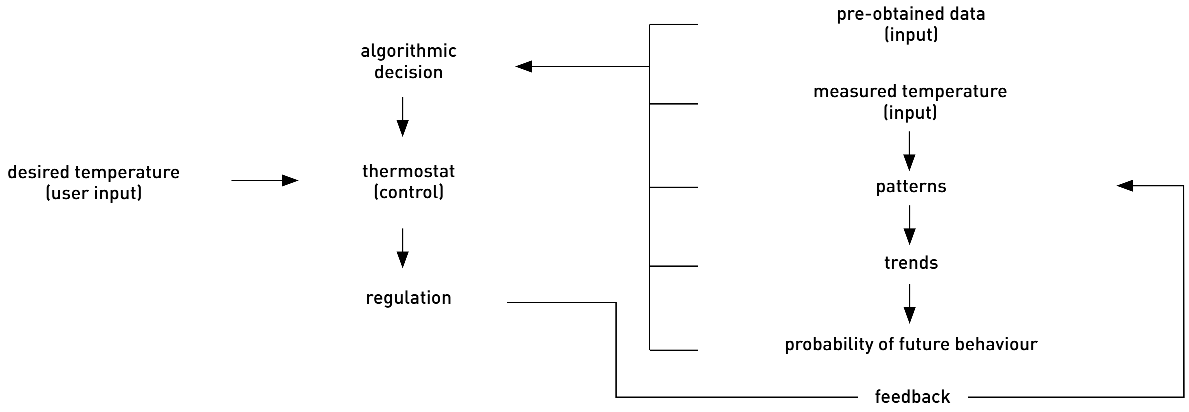
The more data they collect, the more data they can use to compare against other data.

The heater thermostat collects and evaluates data to adjust the temperature based on additional data points, such as humidity, time of day, height and weight, personal preferences of cat and human, and times of presence at home.

We can also program algorithms to:

- make decisions
- guess future behavior

Generally, Catnip leaves the house at 7 AM and comes back at 8 PM, so the thermostat learns to lower the temperature during Catnip's absence.



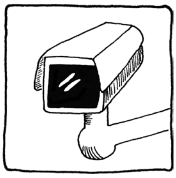
Some examples of how companies and organizations use algorithms[25]:

search engines

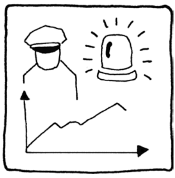
recommendation and scoring systems (Netflix, Amazon, Spotify)

targeted advertising

content filtering (spam protection, child content filters)

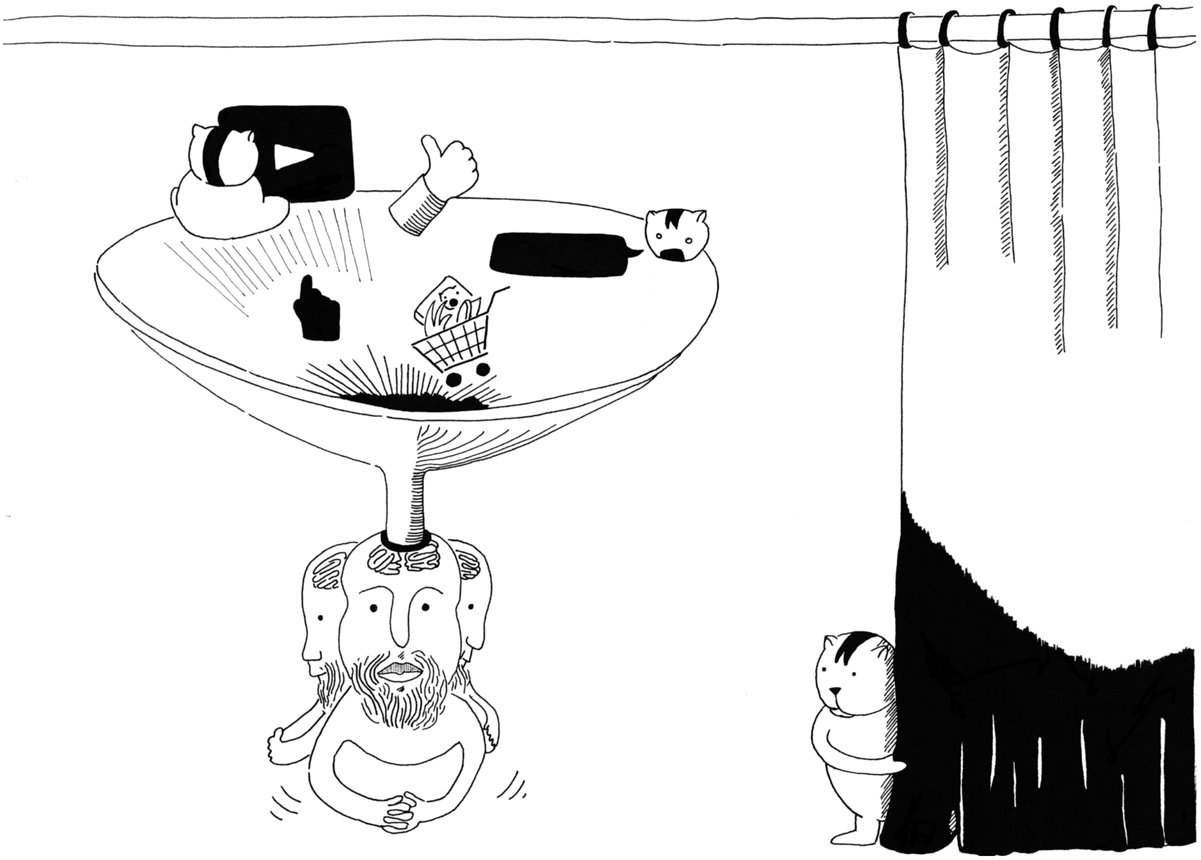


surveillance



predictive policing

algorithmic journalism (automatic production of content)



All of these algorithms automatically select and assign relevance to data.[26] In some systems, users influence this process by cross-referencing, linking, liking, clicking, commenting, watching, and consuming content.

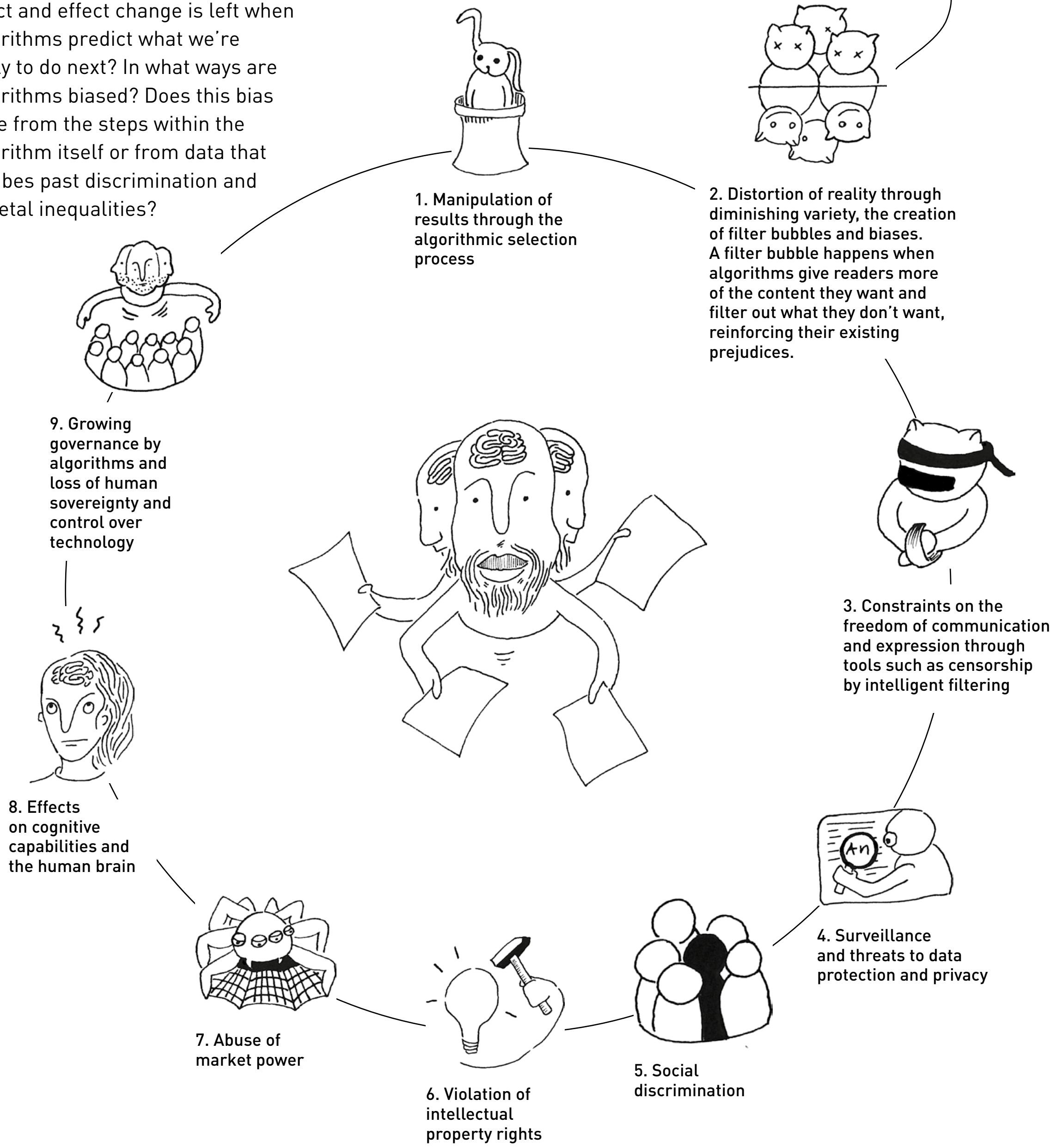
This modern phenomenon is shifting the work of selecting and evaluating content from journalists and editors in traditional media to a quasi-unlimited number of users and consumers who preselect content for algorithms through highly fragmented and small tasks.

However, algorithms are mostly invisible computational structures, which are often subject to business secrets and patents (such as the Google PageRank or the Amazon recommendation algorithm).

Risks of Algorithmic Decision Making

If we want to use algorithms to make the right decisions, we first need to ask many important questions. How do algorithms evaluate data? Does the data accurately represent our multiple and varying identities? How much individual responsibility to act and effect change is left when algorithms predict what we're likely to do next? In what ways are algorithms biased? Does this bias arise from the steps within the algorithm itself or from data that imbibes past discrimination and societal inequalities?

There are nine categories of risks involved when we delegate the selection and classification of valuable information to algorithms:[27]



In the case of the "filter bubble," Catnip wouldn't be able to actually set the thermostat. Household members could only express comfort or discomfort.

1. Manipulation of results through the algorithmic selection process

2. Distortion of reality through diminishing variety, the creation of filter bubbles and biases. A filter bubble happens when algorithms give readers more of the content they want and filter out what they don't want, reinforcing their existing prejudices.

3. Constraints on the freedom of communication and expression through tools such as censorship by intelligent filtering

4. Surveillance and threats to data protection and privacy

5. Social discrimination

6. Violation of intellectual property rights

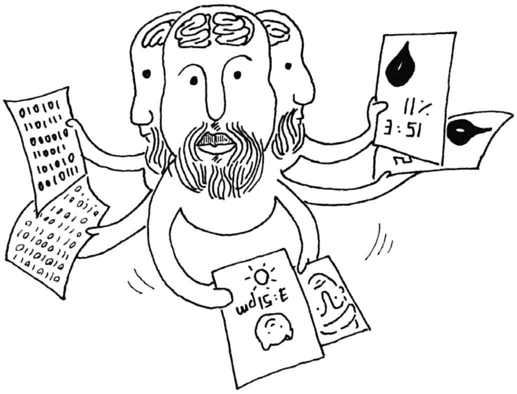
7. Abuse of market power

8. Effects on cognitive capabilities and the human brain

9. Growing governance by algorithms and loss of human sovereignty and control over technology

Levels of Automation

Algorithms can do a number of tasks that would be impossible to perform manually, such as crawling the web to index all of its content.[28] We can call this the **first level of automation**.



However, algorithms can also autonomously interpret the results of their tasks and automate decision making based on these results and on expected outputs. We call this the **second level of automation**. An example is a web search engine's decision to unlist or de-prioritize links to web content that is recognized as spam.

Through the growing use of **machine-learning techniques**, self-learning algorithms can even assist the birth of other algorithms.

This development poses the question of control and authority between algorithms and humans. What level of the autonomy of algorithms should humans be comfortable with?

Governance over Algorithms

As we´ve seen, powerful algorithms already govern our daily lives. But how we govern these algorithms through institutional regulation, laws, and state intervention is still subject to debate.

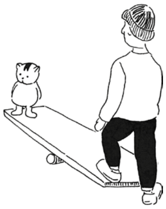
With missing transparency on how specific algorithms operate, how they learn, or which biases they exacerbate, we cannot only rely on the ethics of programmers or of content and service providers who may choose to preemptively self-regulate. Self-regulation means, for example, to conceive and to market a product as "privacy by design": search engines could rely on algorithmic intelligence without collecting, keeping or selling user data, for example.

However, the economic model of many companies is based precisely on the availability of huge amounts of personal user data for algorithmic interpretation—in exchange for a free service.

Just like states create security standards for the construction industry or in the medical field[29], they may create regulations for the use of algorithms.

To name just a few, intervention instruments like these could be used to regulate algorithmic selection[30]:

- capacity building to promote the public´s and private sector's knowledge and ability to analyze risk
- financial incentives or penalties for companies related to the compliance with international human rights standards
- provisioning public services
- ensuring fair competition.



Some of these instruments already have legal frameworks, but the extent to which we should apply existing frameworks to algorithmic selection is a controversial topic.

The European General Data Protection Regulation (GDPR)[31] is a good start. It regulates the processing of personal data within the European Union, the transfer of this data outside the EU, and aims at giving people agency over their personal data.

Although criticized by many for not being up-to-date with regards to the pervasiveness of algorithms, the GDPR inspired the California Consumer Privacy Act (CCPA) in the United States.

But institutional or state regulation also can't easily target risks like bias, **heteronomy** (when decisions are taken by a machine without human control or intervention), and the effects of algorithmic selection on humans´ cognitive capabilities. That's why we need to further raise consumers' awareness and knowledge of such risks.[32]

9

HOW DOES THE INTERNET BUILD ON PREVIOUS TECHNOLOGY?

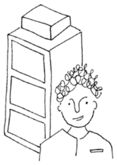
The Layers of the Internet

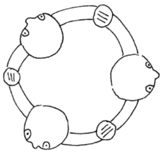
The internet is made of several layers, one on top of the other and interacting with each other. Now that we have learned about the concrete functions and components of the internet, it's time to conceptualize the internet as a whole system, including the people and institutions that operate it. Each layer operates on the layer below it and serves the layer above it.

Social Layer

The most relatable layer of the internet is made up of the entities that use it and the human relationships that govern it.

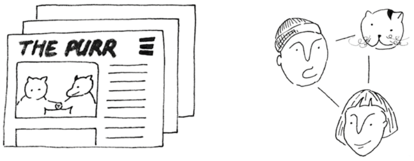
citizens


companies


nonprofit organizations

Content Layer

The content layer, what data is accessible and available over the internet, is perhaps the most recognizable for users.

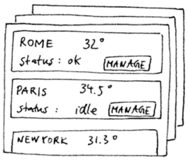


websites (news sites, social media, blogs)

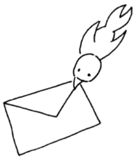
entertainment

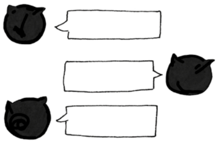
Application Layer

Applications are the ways content is served.



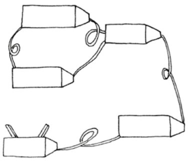
web-based platforms


email

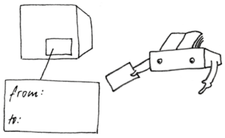

instant messaging

Logical Layer

The logic of the interoperable internet, or its standard protocols, support the connections between devices and the applications running on them.


networking

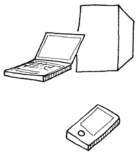
routing

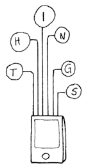


addressing
(Domain Name System)

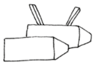
Infrastructural Layer

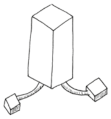
Internet infrastructure is the material basis of the IP network, or the physical components across which the logical layer can send information from one place to another.


computers
(servers, personal computers,
mobile devices)

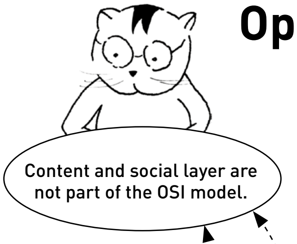


internet of
things (IoT)


routers
(gateways,
switches)

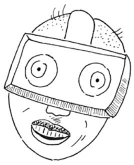

data centers

Open Systems Interconnection (OSI) Model



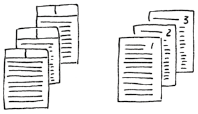
In 1984, the International Organization for Standardization (ISO) published the **Open Systems Interconnection (OSI) model**, a conceptual model that characterized and standardized the technical communication of a telecommunication or computing system. As a concept, in which any layer can take for granted underlying layers, it enables the interoperability of the internet as a diverse communication system across its governance space.

governments



virtual Reality (VR)

voice over IP



traffic controls

telecommunication cables, Wireless networks, Satellite networks

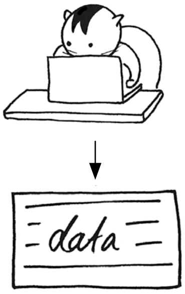
Application Layer

Layer 7: Application

This layer is the layer with which the user directly interacts using client applications, for example a web browser.

Layer 6: Presentation

This layer translates and transmits data between applications, such as text encoding or audio/visual compression.



Logical Layer

Layer 5: Session

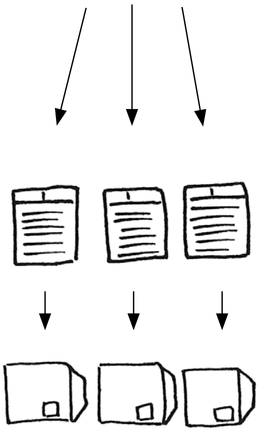
This layer controls the connections between computers by opening the connection, or session, and closing it, according to various session layer protocols.

Layer 4: Transport

This layer ensures the reliable transmission of packets, grouped in so-called messages, segments (TCP), or datagrams (UDP).

Layer 3: Network

This layer defines an addressing scheme and how packets are routed over the network.



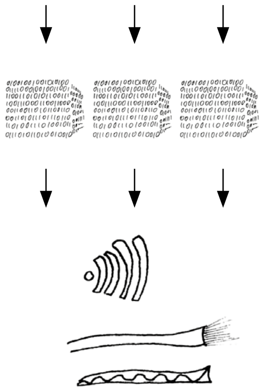
Infrastructural Layer

Layer 2: Data-Link

This layer defines the transmission of data frames between two nodes directly connected in the physical layer.

Layer 1: Physical

This layer defines electrical and physical specifications of the data connection. At this layer, physical media (electrical cable, optical glass fiber, radio frequency spectrum) send and receive raw bit streams. Network adapters, repeaters, and modems operate only at this low level.

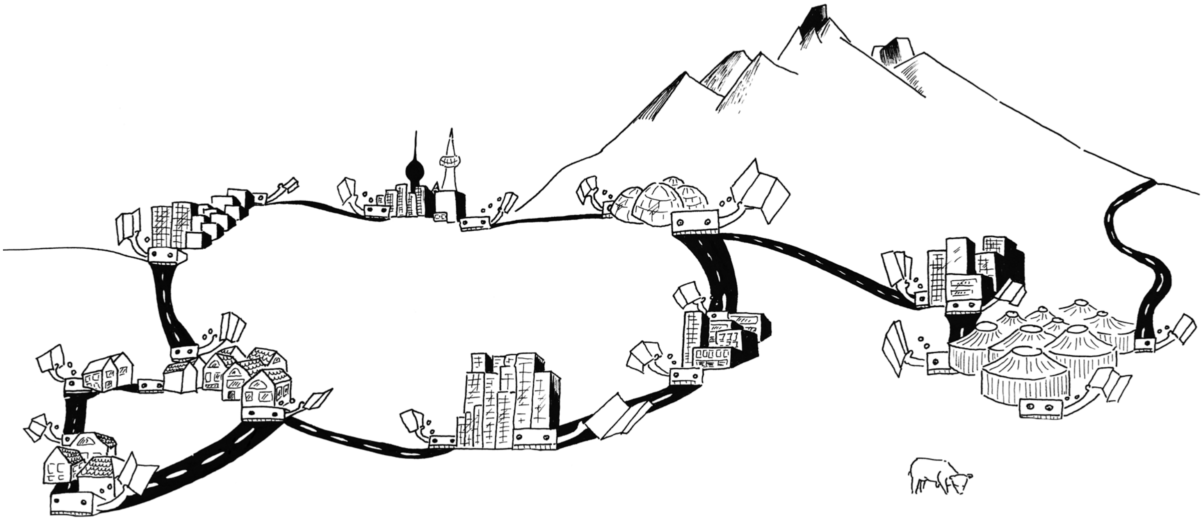


10

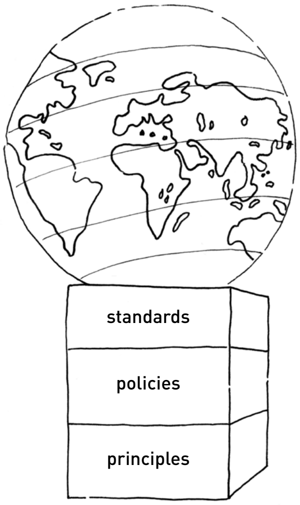
WHO CONTROLS THE INTERNET?

Internet Governance

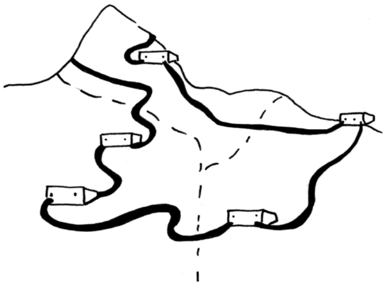
As we've seen, the internet is a globally distributed network made up of many voluntarily interconnected autonomous systems that interoperate through protocols, hardware, and software.[33]



The development, coordination, and management of the internet across a broad range of principles, policies, and technical standards are what govern the internet and ensure that it operates and evolves over time.

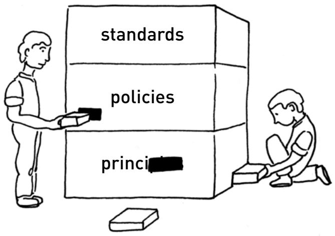


Since the internet moves information across virtually all sovereign nations and since many different public and private entities own and operate its physical parts, the internet has no central governing body to manage this coordination.

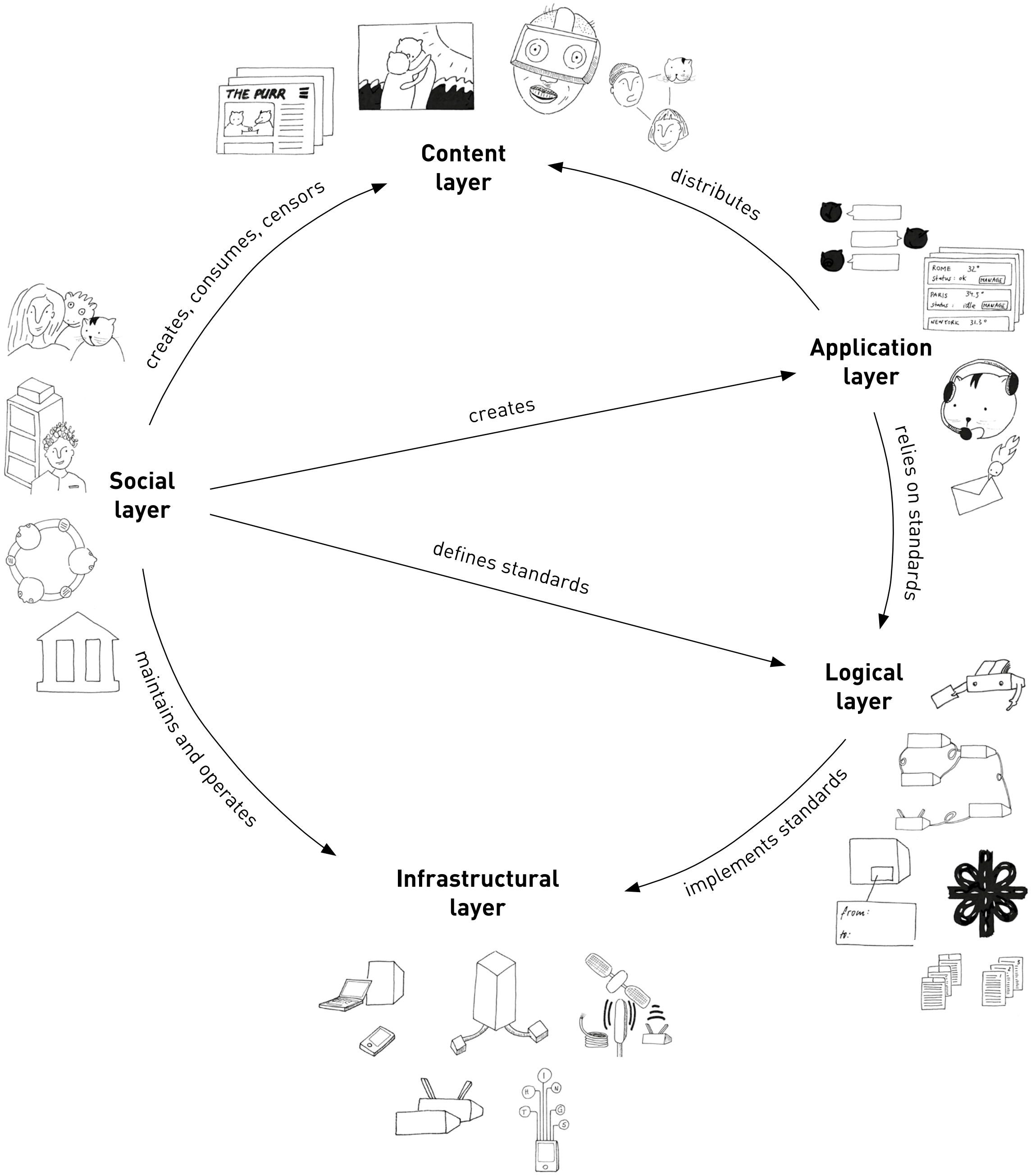


Instead, **internet governance** is a patchwork of organizations and actors who are dedicated to developing and maintaining different aspects of the internet's global interoperability.

Policy changes on one aspect will have a direct impact on others.



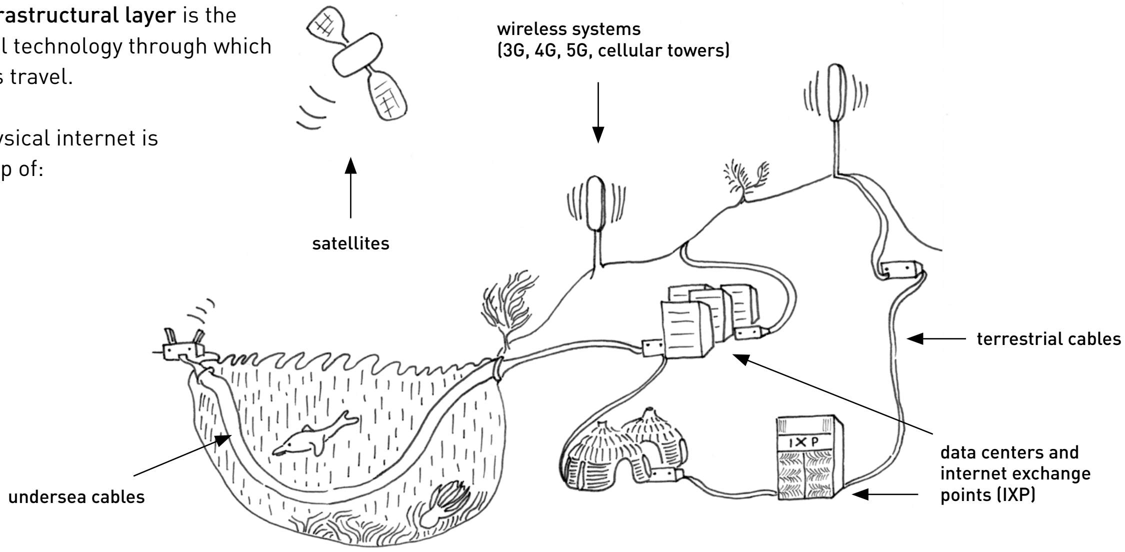
While not an exact mapping, a simplified model of the five layers of the internet—social, content, application, logical, and infrastructural—can give us an understanding of internet governance. In this chapter, we'll look at each layer and the internet governance processes that concern themselves with those layers.



Infrastructural Layer

The **infrastructural layer** is the physical technology through which packets travel.

The physical internet is made up of:


wireless systems
(3G, 4G, 5G, cellular towers)
satellites
terrestrial cables
data centers and internet exchange points (IXP)
IXP
undersea cables

Internet Engineering Task Force (IETF)

The **Internet Engineering Task Force (IETF)** develops and promotes a wide range of internet standards, among other standards, including the Internet Protocol suite (TCP/IP). The IETF produces technical documents called **Request for Comments (RFCs)** that outline such standards. Although RFCs aren't compulsory, they influence the way people design, use, and manage the internet. IETF is an open standards organization, with no formal membership. All participants are volunteers, but their employers or sponsors usually fund their work. (We discuss this more in the next chapter.)

→ *https://www.ietf.org*

Internet Research Task Force (IRTF)

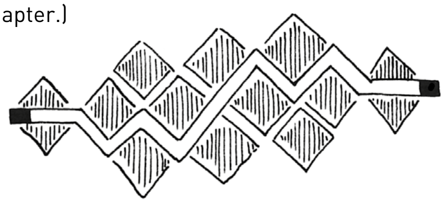
The **Internet Research Task Force (IRTF)** promotes research on the evolution of the internet by creating focused, long-term research groups that study topics related to internet protocols, applications, architecture, and technology.

→ *https://www.irtf.org*

Internet Architecture Board (IAB)

The **Internet Architecture Board (IAB)** oversees the technical and engineering development of the IETF and IRTF.

→ *https://www.iab.org*



IETF and IRTF are parallel organizations. While IETF focuses on short time research, IRTF focuses on long term developments.

This infrastructure is largely privately owned by telecommunication companies all over the world.

The infrastructural layer's technology—hardware and software—implements the standards defined at the logical layer (such as the Internet Protocol) in order to function interoperably. A router that does not speak the Internet Protocol properly or uses a proprietary communications protocol can't reliably function within the internet.

The patchwork of organizations that are responsible for ensuring the operation and interoperation of the internet at the infrastructural layer specify standardized hardware, define internet protocols, and coordinate governance across global and regional internet authorities. The work of organizations at this layer is often invisible to most internet users, yet crucially important for a resilient and interoperable internet.

Internet Society (ISOC)

The **ISOC** is a US nonprofit organization whose mission is "to promote the open development, evolution, and use of the internet for the benefit of all people throughout the world." ISOC supports and promotes the work of IAB, IETF, and IRTF. ISOC is also the parent organization of IETF—all IETF RFC documents, including those describing "Internet Standards," are copyrighted by ISOC.

→ *https://www.internetsociety.org*

Internet Corporation for Assigned Names and Numbers (ICANN)

The **ICANN** coordinates the internet's system of unique identifiers—IP addresses and top-level domain space (DNS root zone). The Internet Assigned Numbers Authority (IANA), the organization we discussed in Chapter 3, is a department of ICANN.

The five regional internet registries (RIR) manage the allocation and registration of IP addresses within geographic regions of the world:
Africa: *https://afrinic.net*
Asia Pacific: *https://apnic.net*
Canada & United States: *https://arin.net*
Latin America & Caribbean: *https://lacnic.net*
Europe, the Middle East and parts of Central Asia: *http://ripe.net*

→ *https://icann.org*
→ *https://iana.org*

Institute of Electrical and Electronics Engineers (IEEE)

The **IEEE** develops a large range of international standards for modern telecommunications hardware, such as standards for networks that allow devices to connect to the internet, including Ethernet, Bluetooth, and Wi-Fi. IEEE also works on standards for robotics, smart cities, and artificial intelligence. IEEE works together with the International Organization for Standardization (ISO).

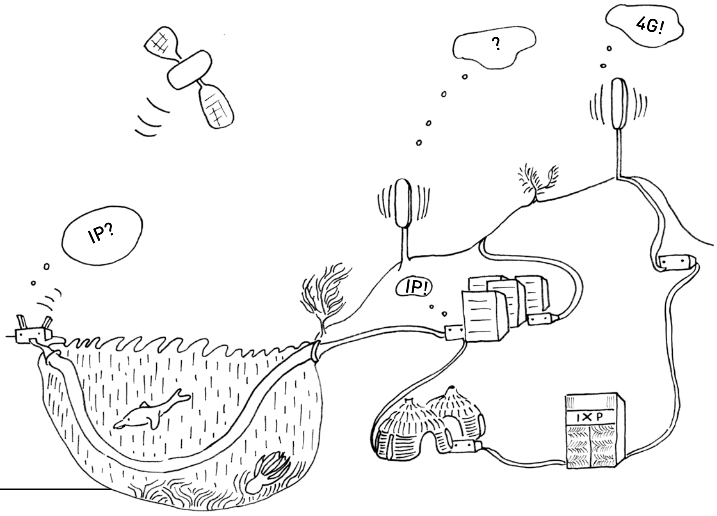
IEEE's structure is complicated, but it has separate regional and technical columns, each containing various units that are based on a technical topic, such as local area networking or artificial intelligence. Anyone can join a working group and contribute to building standards.

→ *https://www.ieee.org*

Logical Layer

In order to interoperate, networks need to “speak the same languages,” that is, follow the same standards and procedures. The definition of these standards constitutes the **logical layer** of the internet.

The logical layer makes the internet tick. The logical layer is a set of procedures that ensure that all the processes necessary to make the internet function actually work.



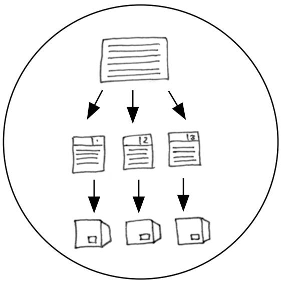
Here are some clear examples of where the logical layer is governed:

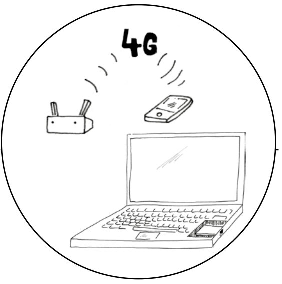
Internet protocols and standards, such as TCP/IP, are developed and defined at the IETF.

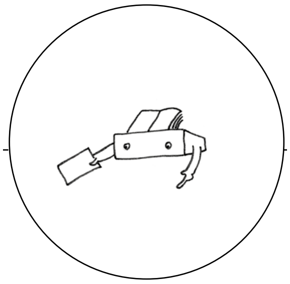
Hardware and Wi-Fi standards are developed by the Institute of Electrical and Electronics Engineers (IEEE).

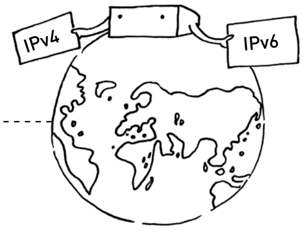
Internet services such as the Domain Name System and the management of the root zones are managed by ICANN.

IP addresses are globally assigned and distributed by IANA and the regional internet registries (RIR).









International Telecommunication Union (ITU)

Much of the coordination on the physical layer is facilitated by the **ITU**, an agency of the United Nations responsible for coordinating the global use and access to the radio spectrum and cellular networks. The ITU also promotes international cooperation in terms of satellite orbits and coordinates the development of technical standards related to telecommunications.

The ITU’s standardization work has become the vehicle for repressive regimes to limit internet freedom in their jurisdictions. Regimes can influence the standards-making process of next-generation technology to embed features that facilitate surveillance and tracking.

Content and Application Layer

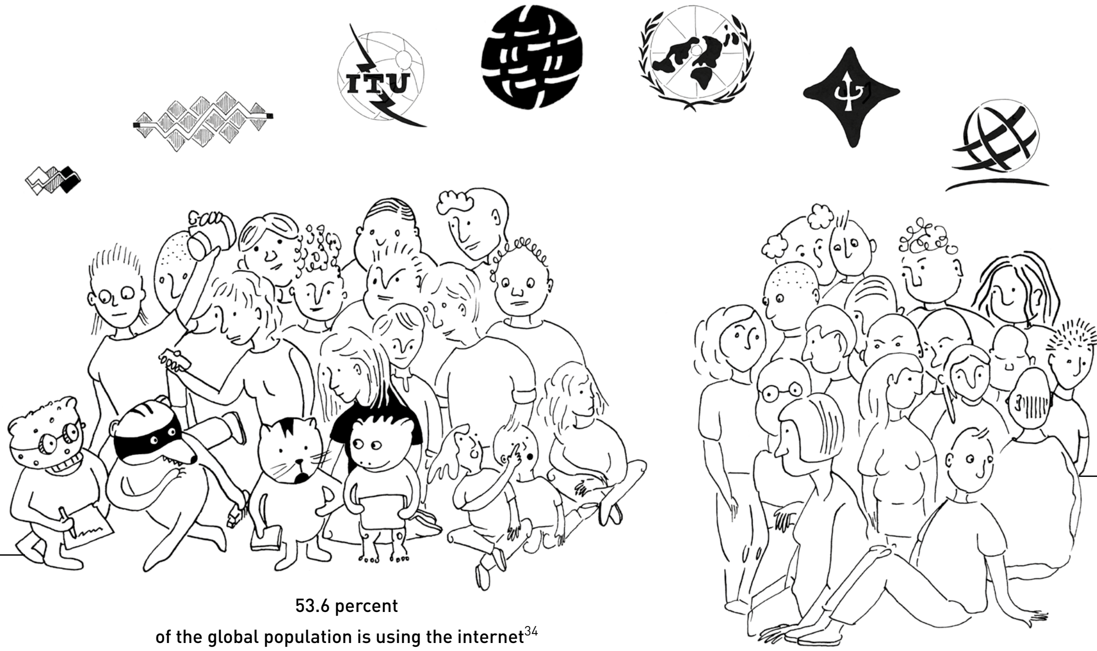
The **content layer** is the heart of the political and public debate on internet governance. Many of the most discussed issues—privacy, encryption, freedom of speech, human rights, and intellectual property—arise from the software and applications occupying this layer. Many of these discussions take place through traditional policy instruments such as state regulation or private–public agreements.

Internet Governance Forum (IGF)

→ *https://www.intgovforum.org*

The **IGF** was created to serve as a global forum for governments, private companies, and civil society to discuss the issues that dominate the content layer.

Social Layer



53.6 percent of the global population is using the internet[34]

It's important not to forget that the internet is created and used by people. In our own expression of the layers of the internet (see previous chapter), we define a **social layer** because it gives people their rightful place in the stack.

For the purposes of this book, we take a departure from the often-cited OSI model of the internet, which is still useful for engineers, by adding critical, nontechnical layers to describe the internet. Our reason is that we see the internet as fully integrated into the daily lives of people around the world, even those who are not yet fully connected to it.

The social and economic aspects of the internet are driven mostly by:

- states and governments
- private bodies and businesses
- policy organizations operating at the logicial layer (such as the IETF)
- citizens

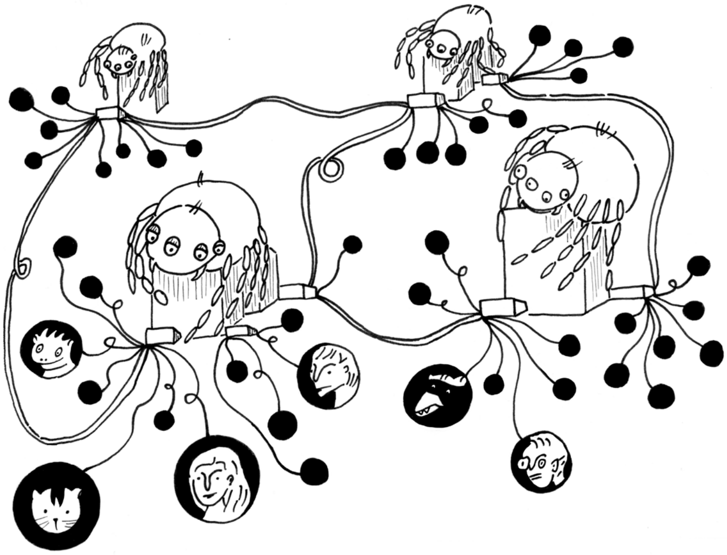
11

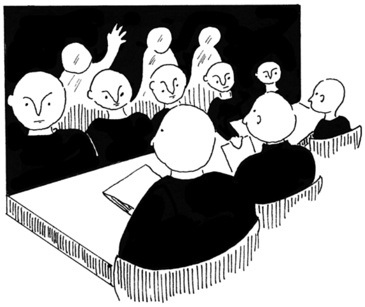
HOW IS POWER DISTRIBUTED OVER THE DECENTRALIZED INTERNET?

While the vision of the early internet was a utopic network with equally distributed power,

...the current reality is that the internet controls content through consolidated services.

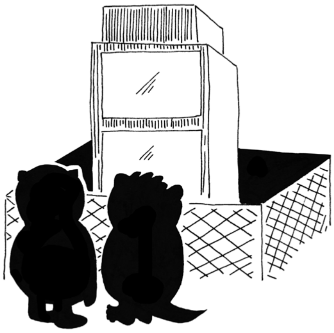
The consolidation of power over the internet is a problem for many reasons.



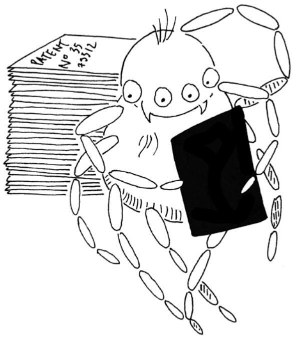


First, only a few big companies from developed countries determine how the internet works. The results of their decision making risk excluding everyone else.

Second, the private sector shapes internet standards without the involvement of civil society, possibly overlooking their needs.



Third, private companies aren't governed democratically. It's a challenge for civil society, in particular, to govern the decentralized internet across borders because it's hard to guarantee that international companies will respect laws and international human rights norms to protect privacy, freedom of expression, and freedom of association.

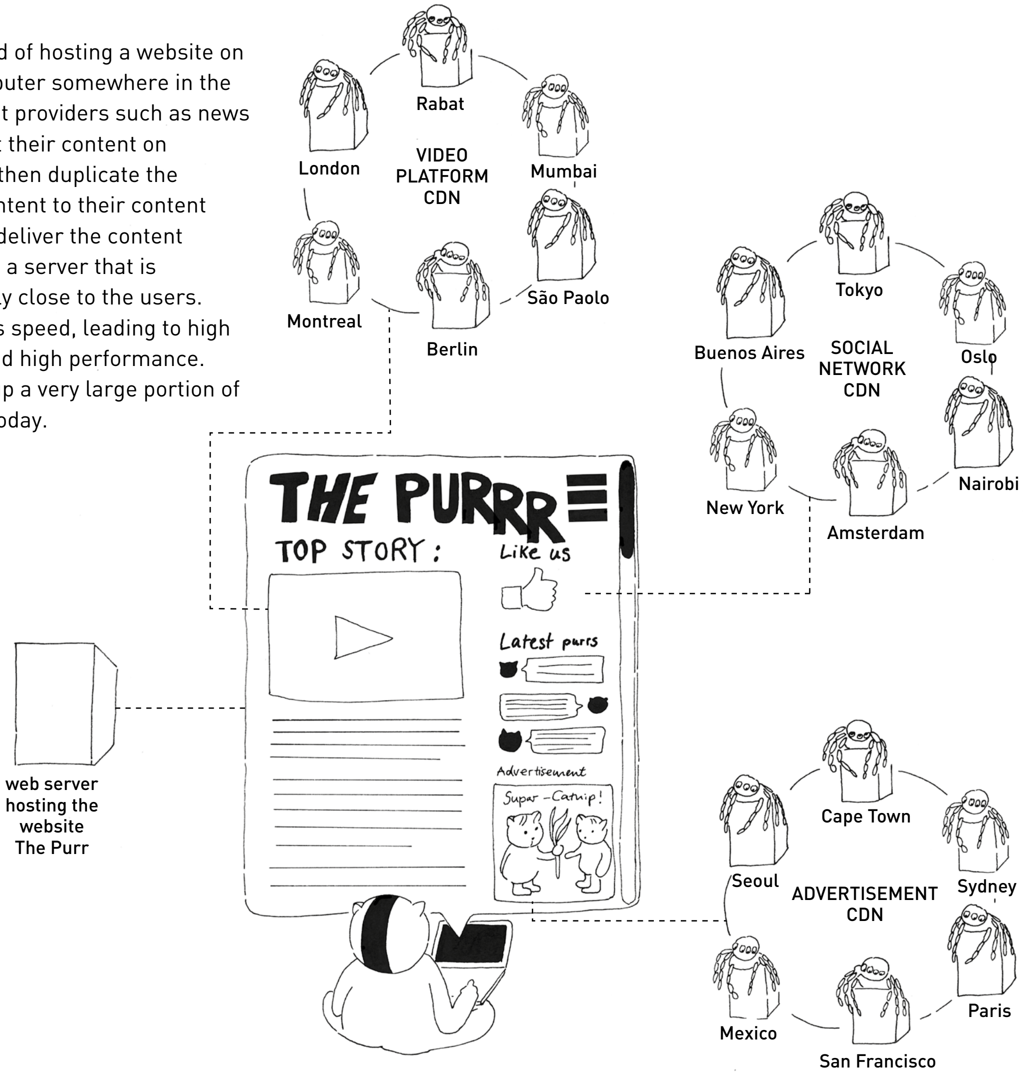


Lastly, a centralized internet leads to the proliferation and concentration of patents and intellectual property within a few privileged groups and regions.

Content Delivery Networks

Content delivery networks (CDN) are geographically distributed networks that host and deliver content—like images, videos, or other streaming media—to users.

Today, instead of hosting a website on a single computer somewhere in the world, content providers such as news websites host their content on CDNs. CDNs then duplicate the providers' content to their content network and deliver the content to users from a server that is geographically close to the users. This improves speed, leading to high availability and high performance. CDNs make up a very large portion of the internet today.



When I visit a website, I might be loading content from many different servers. Although this makes the internet faster, it introduces another intermediary that if untrusted, risks user privacy since CDNs can see who loads which content. However, hosting content on CDNs can mitigate censorship since by design it makes the availability of content more resilient.

Because of the rapidly growing video streaming traffic on the internet, telecommunication service providers have even started to launch their own CDNs, commonly referred to as **telco CDNs**.

Some of the major CDNs and content providers and their services include:

- Cloudflare
- Akamai Technologies
- Amazon CloudFront
- Microsoft Azure
- telco CDNs

We can surmise that every CDN, like nearly all internet intermediaries, has been asked to comply with court orders for user data. Here's a deeper analysis of some of these CDNs.

Cloudflare

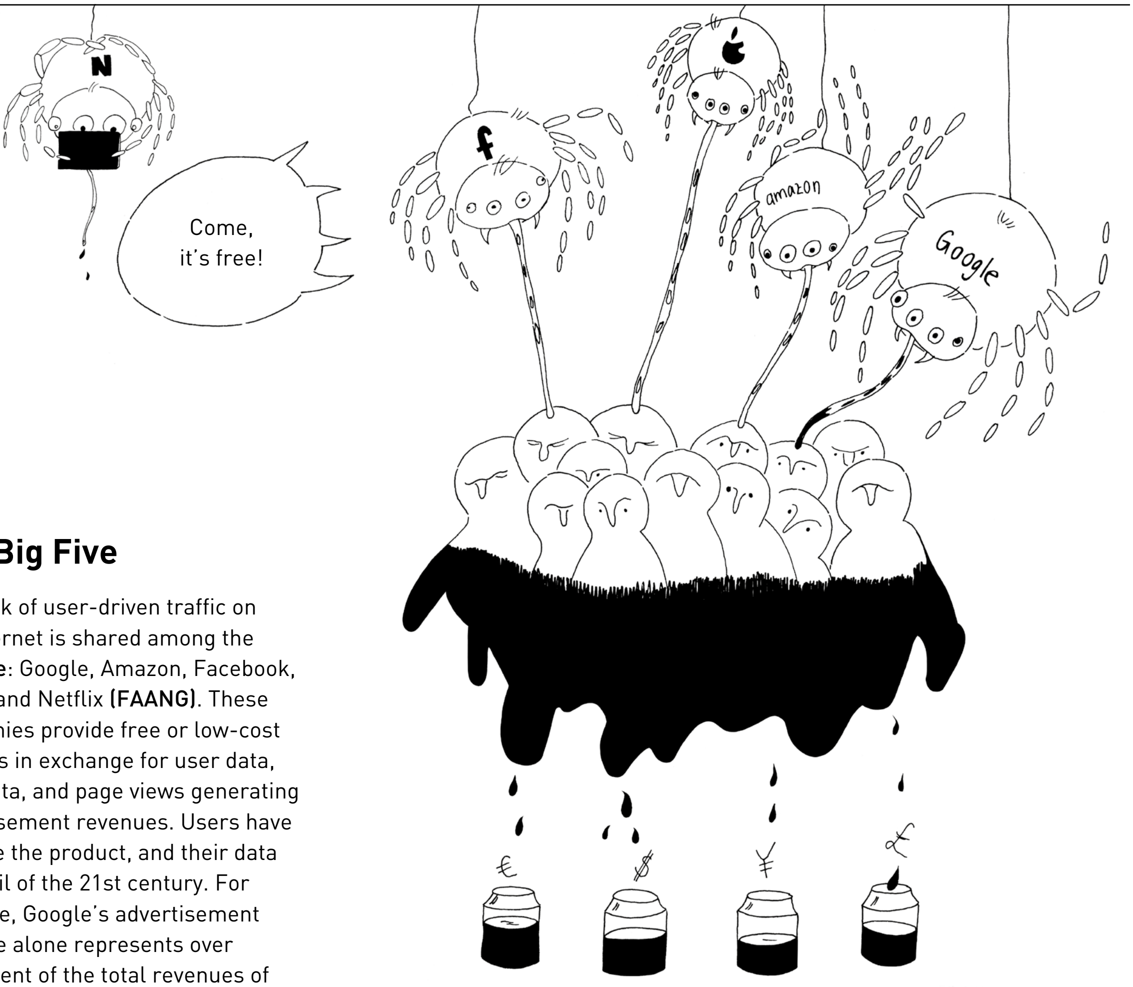
Cloudflare is a US company that focuses on web performance by offering companies cybersecurity, DDoS protection, and services related to resilient DNS and content provision. Cloudflare also hosts many phishing websites and has been criticized for its arbitrary decisions to not take down discussion forums of Daesh (ISIS) but on the other hand choosing to remove 8chan.

Akamai

Akamai is a "cloud service provider," which means it sells its distributed content delivery network to large websites. Akamai's CDN serves between 15 percent and 30 percent of global web traffic! The NSA and FBI have reportedly used Facebook's Akamai CDN to collect information on Facebook users.[35]

Telco CDNs

The same company that is supplying your broadband might also now provide the distribution power of a content delivery network. This is to save costs on the increased use of broadband for streaming video. If one telco subscriber streams a popular video from across the world, a telco CDN would cache that content locally in case another subscriber wants to stream it, too.



The Big Five

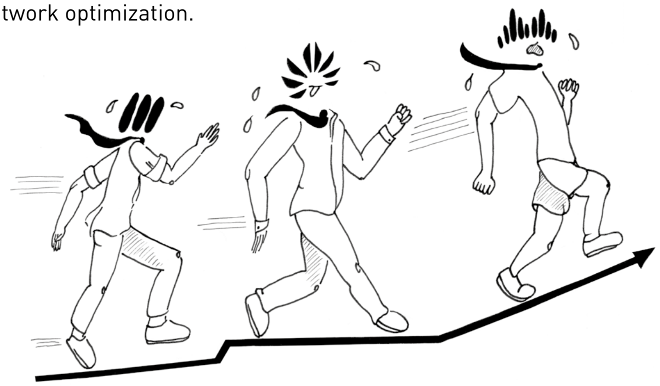
The bulk of user-driven traffic on the internet is shared among the **Big Five**: Google, Amazon, Facebook, Apple, and Netflix **(FAANG)**. These companies provide free or low-cost services in exchange for user data, metadata, and page views generating advertisement revenues. Users have become the product, and their data is the oil of the 21st century. For instance, Google's advertisement revenue alone represents over 90 percent of the total revenues of the company.

Physical Centralization of Power

Even on the physical layer we can observe the **centralization** of power over the internet. The companies dominating the hardware industry include Ericsson, Huawei, and Cisco.

Ericsson Mobile Communications is a Swedish multinational corporation that develops wireless technologies, mobile phones, and network services such as network technology deployment, network transformation, and network optimization.

Huawei Technologies Co., Ltd. (華為) is a Chinese multinational corporation and the largest telecommunications equipment manufacturer in the world.



Cisco is a San Francisco–based multinational corporation that produces network equipment for data centers, telephone products, and applications. Cisco has partnered with Akamai to create network device deployment tools for data centers.

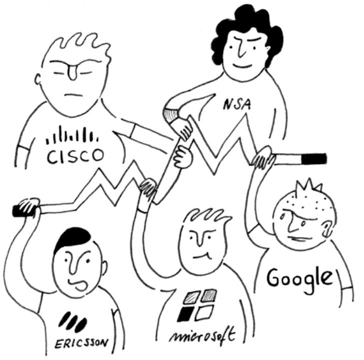
Cisco has partnered with Ericsson "to develop a cloud and IP architecture together, and [...] to build a network management system that covers both companies' network gear."[36]

Cisco has partnered with Apple for the Cisco Security Connector, an application for iPad and iPhone that delivers more security and privacy for Apple iOS devices.

Political Centralization of Power

The consolidation of power and influence is also apparent in the logical layer. The multistakeholder model (discussed in Chapter 12) has not yet brought about the openness in internet governance that many had intended. A key reason is that membership in internet governance organizations consists largely of employees of big companies. As members of these organizations, the employees work to standardize their companies' work product and thereby give it legitimacy. By these means, their companies gain a competitive edge.

Consolidation and Influence at the IETF



IETF meetings take place four times a year, around the globe. On paper, every individual may attend and participate remotely in the writing of RFCs and standards, but in practice flying to each of the meetings and paying the entrance fee of US $875 isn't within the reach of people who aren't sponsored by a company. This skews the creation of IETF standards toward the interests of big companies that can afford to attend IETF meetings.[37]

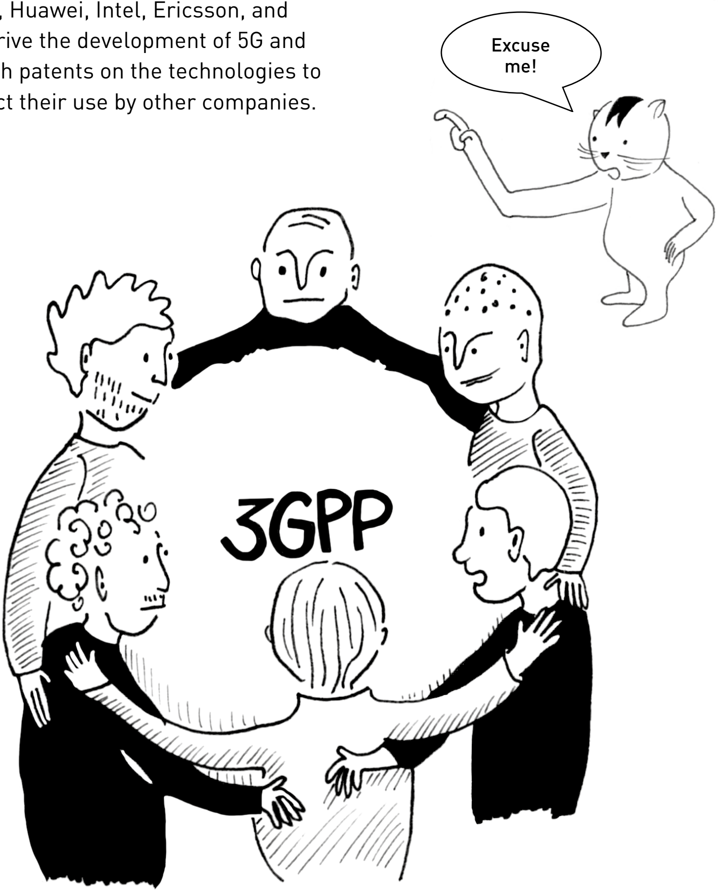
ICANN: An Industry Expo

ICANN is a nonprofit organization and partly generates its income by collecting licensing fees from creators of generic top-level domains (gTLDs). This system was meant to be an open process that promoted competition and freedom of expression online, but in reality it works to harmonize and govern industry players who own gTLDs. Therefore, ICANN's decisions tend to follow US intellectual property laws and favor business interests. The long process of applying for new gTLDs requires strong English skills and a minimum cost of US $185,000, resulting in very few applications from the Global South.

The Rise of 5G at the ITU

The wireless standard **5G** provides increased bandwidth and access to broadband speeds even in remote areas. With its increased data rates, improved coverage, more efficient and reliable connections, and lower cost, this technology will likely replace wired connections in urban areas. Service providers of landline internet won't be able to compete in terms of price and cost.

Because 5G could become the primary way that individual users connect to the internet in the future, several companies are heavily lobbying issues related to 5G at the ITU in order to compete over future market share. Companies such as Nokia, Huawei, Intel, Ericsson, and ZTE drive the development of 5G and publish patents on the technologies to restrict their use by other companies.



The development of standards for 5G is specifically happening in the **3rd Generation Partnership Project (3GPP)**, a standardization organization that's contribution driven, like the IETF. However, though anyone can contribute, most of the participants are employees of big telecommunication companies: China Mobile, Verizon, AT&T, Vodafone, Deutsche Telekom, and América Móvil. Standards developed at 3GPP are approved by the ITU because they concern the mobile spectrum.

Private companies have significant influence at the ITU since they often have more financial and time resources to engage in ITU's policy procedures. Their influence is disproportionately greater than that of nongovernmental members, so civil society engagement in ITU needs to increase to counterbalance commercial influence.

12

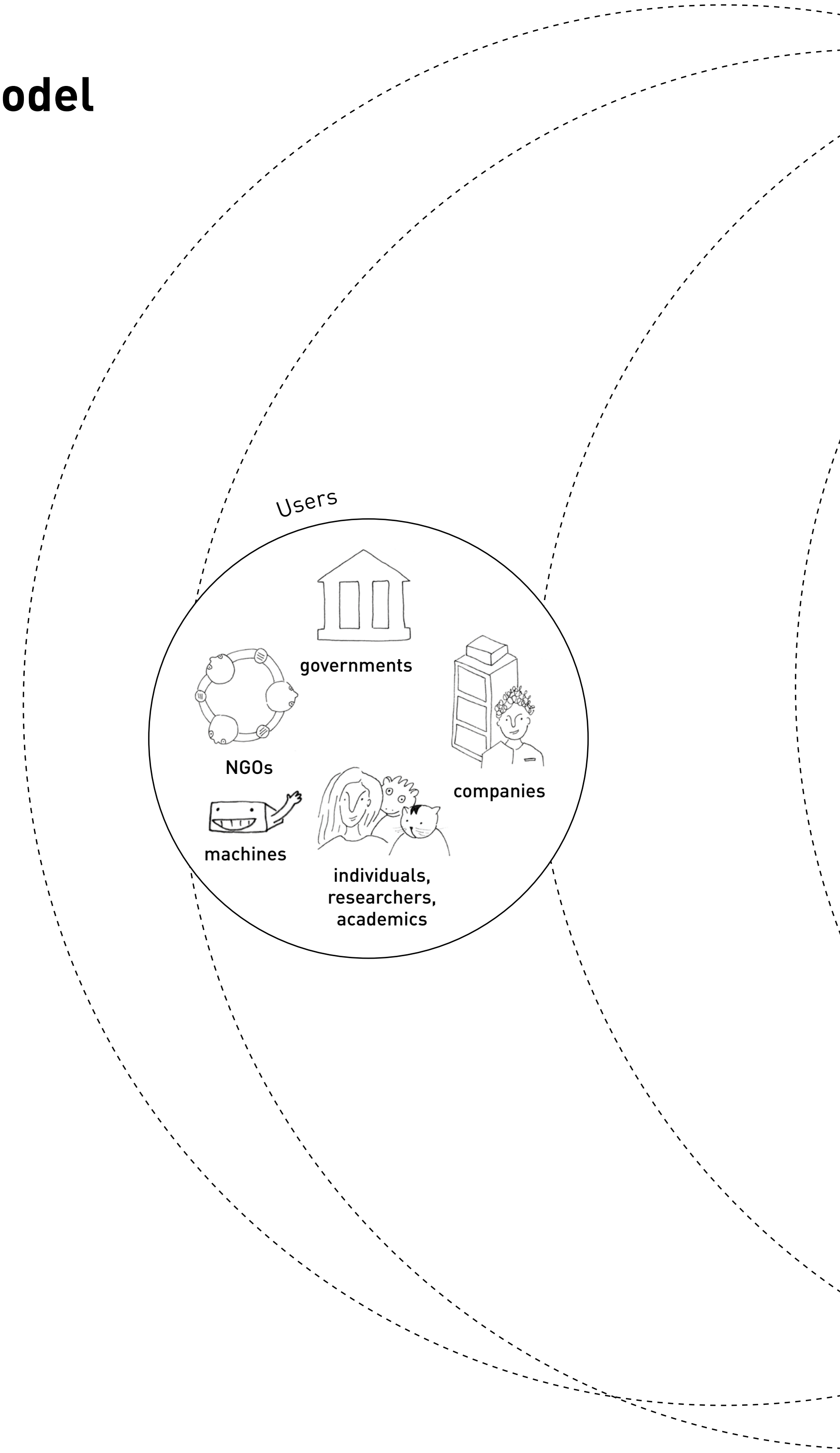
HOW CAN CIVIL SOCIETY ENGAGE IN INTERNET GOVERNANCE?

Internet governance is unique. It departs from traditional multilateral governance in that it allows civil society organizations (CSOs) to directly partake, on relatively equal footing with other stakeholders, in the creation of the policies that govern the internet.

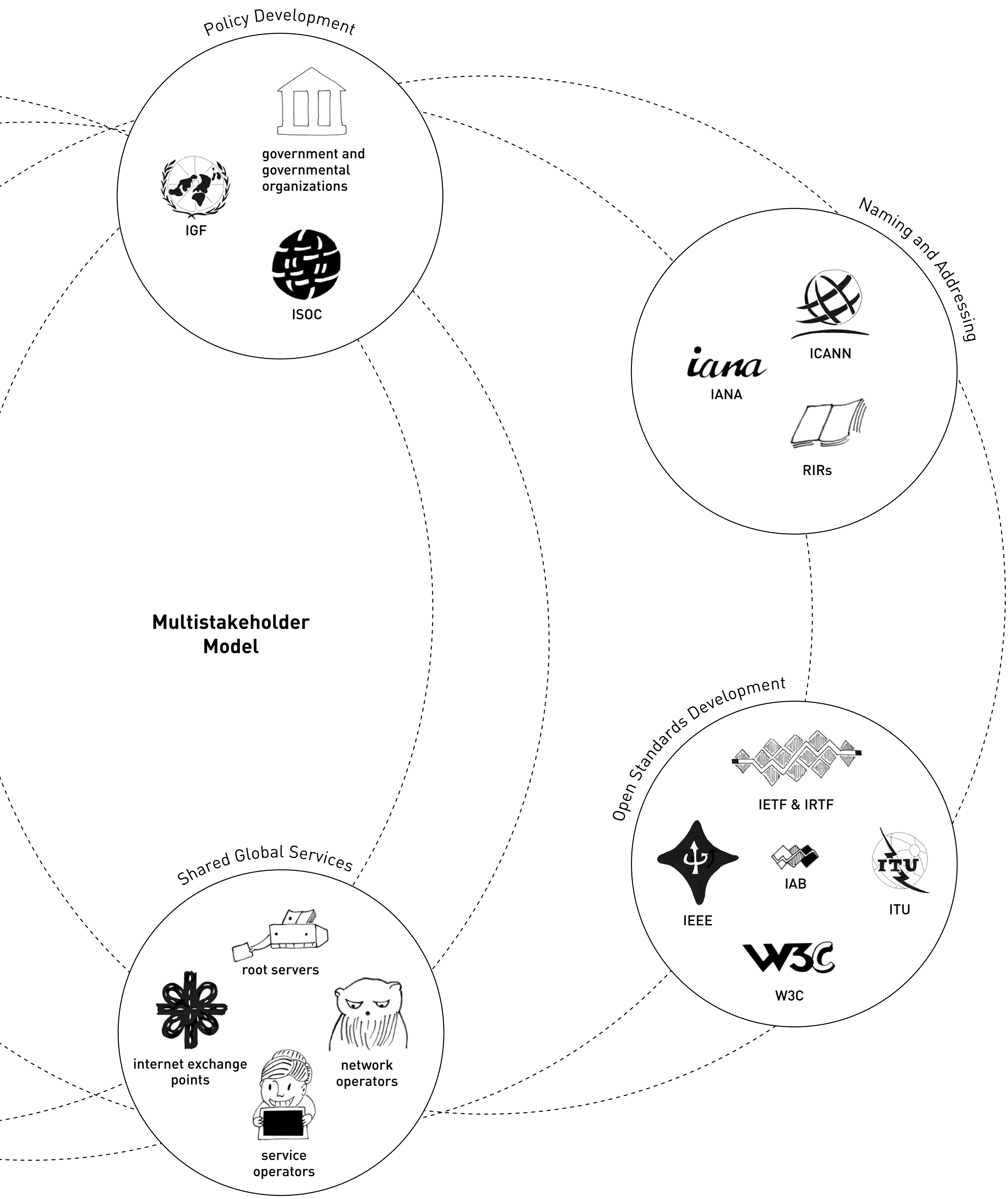
The Multistakeholder Model

While previous worldwide telecommunication technologies such as the telephone or the telegraph were largely controlled and regulated by governments or government-controlled monopolies, the internet's governance is based on a unique multistakeholder model.

This **multistakeholder model** entails the participation of governments, the private sector, civil society, and anybody who cares to show up and do the work. The participation of multiple parties allows civil society actors to directly discuss and help determine the policies that shape the internet on a global scale instead of being confined to lobbying.

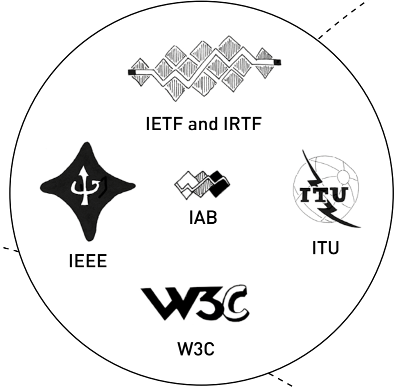


However, multistakeholder governance is still in its early stages, and there are legitimate concerns about whether power is equally balanced among the various stakeholders (see Chapter 11). Nevertheless, there are still many venues and mechanisms where actors can practice mulitstakeholder internet governance, most of which are open processes. In this chapter we'll discuss these venues and mechanisms and how you can be involved in them.


Policy Development
government and governmental organizations
IGF
ISOC
Naming and Addressing
iana
ICANN
IANA
RIRs
Multistakeholder Model
Open Standards Development
IETF & IRTF
IEEE
IAB
ITU
W3C
Shared Global Services
root servers
internet exchange points
network operators
service operators

Organizations Where You Can Engage in Internet Governance

Given the aspirations for multistakeholder input in global internet governance, diverse and informed stakeholders can, and must, engage in the decision-making bodies to ensure that the internet remains open, pluralistic, and democratic. From standards-setting bodies to debate forums, not all internet governance organizations address all internet governance issues. Furthermore, it is critical to understand and enforce the mandate of these bodies to ensure strong engagement and reduce overlap.



Open Standards Development

There is a constellation of organizations that set global standards for hardware, protocols, and other internet-related technologies.

IETF

The IETF sets low-layer internet standards. It can be a confusing place, considering its use of specialized jargon, unique procedures, and technical subject matter. The IETF celebrates its quirks,[38] such as using "humming" to determine consensus. At the same time, its earnest influence on the shape and capacities of the internet requires participants to focus on such issues as human rights, privacy, security, and access. The participation of civil society is crucial to ensuring that the IETF develops internet technology with regard for the security and privacy of the most vulnerable and threatened users.

There are many opportunities to get involved with IETF:

- Join a research group as an individual participant.
- Take part in the mentoring program of the IETF.
- Organize a "Birds of a Feather" (BoF) informal meeting with like-minded individuals to determine whether there's a need to set up a new formal working group.
- Suggest a topic for the IETF hackathons, which focus on solving real-life problems and take place the weekend before the IETF meetings.
- If you self-identify as a woman, join the Syster mailing list and meetings aimed at improving the gender balance of participation at the IETF.

→ You can find IETF's website at: *https://ietf.org*

IEEE

IEEE is a global professional association for technology and engineering with a distributed organizational structure. It's open to suggestions for new technologies and the establishment of working groups. To get engaged with IEEE, you can:

- Join a local section, chapter, student branch, or affinity group.
- Join an existing technical standards group at the IEEE.
- Join the Global Initiative for Ethical Considerations in the Design of Autonomous Systems or one of its several working groups.

→ You can become a member at IEEE through their website: *https://www.ieee.org/membership_services/membership/join*

ITU

Civil society participation in the oldest international governance body, the ITU, is difficult. The ITU is a multilateral institution, meaning only nation states can be members, though observer-member status is now open to nongovernmental organizations. Despite its lack of open participation, the ITU plays a crucial role in developing the standards and infrastructure for telecommunications access, especially in the Global South. To get involved with ITU, you can:

- Join ITU as a contributing member, either by joining a national delegation or by becoming a nongovernmental sector observer-member.
- Identify civil society actors or academic institutions that are ITU members and work with them to set a collective agenda.
- Participate in the ITU conference that occurs every four years to decide the next four-year roadmap of the organization.

→ You can find ITU's website at: *https://www.itu.int*

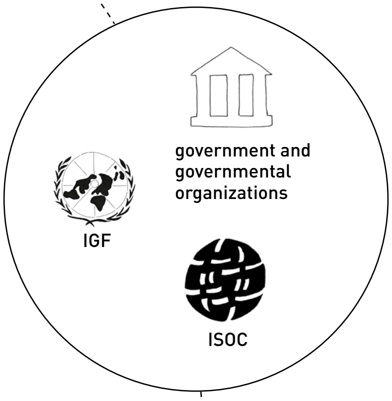
Policy Development

The internet is ubiquitous, and so is internet-related policy setting; yet for discussion of global issues, one major multistakeholder space stands out.

Internet Governance Forum

The **Internet Governance Forum (IGF)** is an open forum where anyone can submit and organize discussions and workshops in yearly meetings. During these meetings, diverse stakeholders come together to form working groups called **Dynamic Coalitions** to discuss specific issues. For example, the Dynamic Coalition called "Freedom of Expression and Freedom of the Media on the Internet" examines issues related to its name. The Dynamic Coalition called "Community Connectivity" focuses on how we can use community networks to improve connectivity and access to information for low-access areas.

→ To participate in IGF meetings, go to: *https://www.intgovforum.org/multilingual/content/participate-in-igf-meetings*



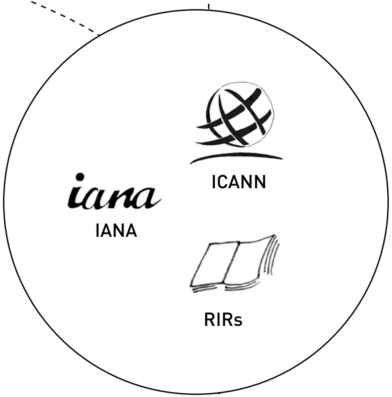
Naming and Addressing

There is a need for global governance over the distribution of the finite resources of domain names and IP addresses.

ICANN

Participation at ICANN to govern the domain space is open to anyone willing to volunteer time. ICANN provides many tutorials about their work and how you can get involved on their website:. *See https://www.icann.org*

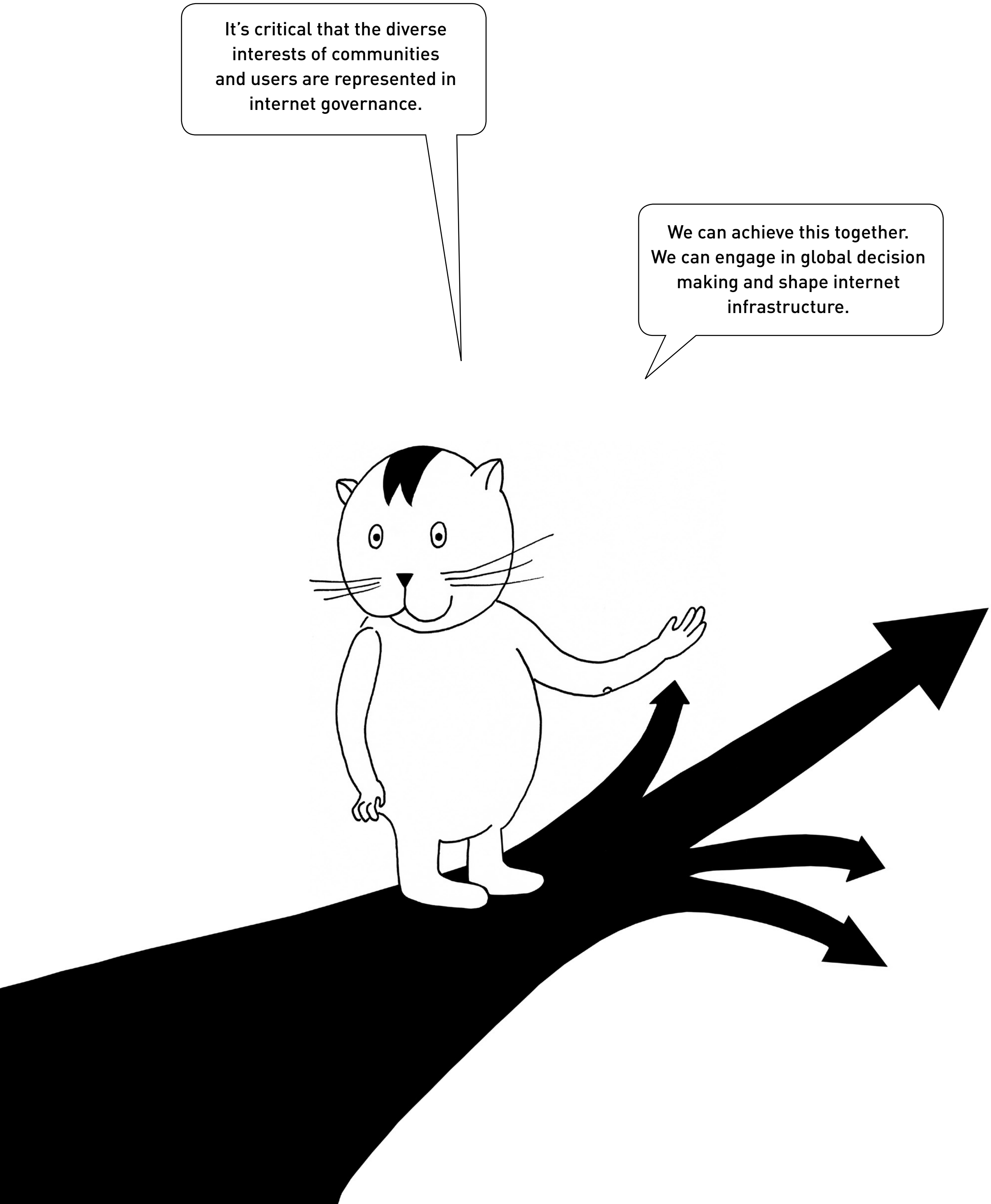
A great place to start is an ICANN affiliated organization, the **Noncommercial Users Constituency (NCUC)**, which helps set gTLD policy. You can find their website at:
→ *https://www.ncuc.org*
Since 2016, ICANN's bylaws stipulate that whenever ICANN makes policy decisions, they must conduct an assessment on how that policy might impact human rights. You can find such assessments here:
→ *https://icannhumanrights.net*



While internet governance might recognize CSOs as an important stakeholder group in the creation of the policies that govern the internet, few actually participate in setting critical internet policy at the global level.

Civil society organizations engaging in technical standards development are even more rare. Just remember that any civil society organization can actively shape these policies itself. The governance of the internet will always encounter difficult decisions and values tradeoffs that pit interests against one another.

And civil society is not monolithic: the public interest can be represented in a variety of ways.[39]


It’s critical that the diverse interests of communities and users are represented in internet governance.
We can achieve this together. We can engage in global decision making and shape internet infrastructure.

NOTES

1. On mobile phone networks, instead of a MAC address devices need to have an International Mobile Equipment Identity (IMEI) identifier.

2. James Bamford, "Edward Snowden: The Untold Story" (August 2014), *https://www.wired.com/2014/08/edwardsnowden*

3. Bruce Schneier, "Wi-Fi Hotspot Tracking" (October 10, 2019), *https://www.schneier.com/blog/archives/2019/10/wi-fi_hotspot_t.html*

4. DHCP is the most common way to receive a network address. IP addresses can also be manually configured.

5. According to research and mapping of Patrick Maigron, Associate Professor at Telecom SudParis, France, the total number of autonomous systems worldwide was 97,004 as of June 2020. Source: Regional Internet Registries Statistics, "RIR Delegations & RIPE NCC Allocations," *https://www-public.imtbs-tsp.eu/~maigron/RIR_Stats/RIR_Delegations/World/ASN-ByNb.html*

6. As of June 2020, Packet Clearing House—the international organization responsible for providing operational support and security to critical internet infrastructure, including internet exchange points—counted 1,061 IXPs worldwide. Source: Packet Clearing House, "Internet Exchange Directory," *https://www.pch.net/ixp/dir*

7. The HTTP status code 418 indicates that the server refuses to brew coffee because it is a teapot. It was added to the HTTP protocol in 1988 as an April Fools' joke.

8. For details, see Internet Engineering Task Force, "HTTP over TLS" (May 2000), RFC 2818, *https://tools.ietf.org/html/rfc2818*

9. Seth Schoen, "New Research Suggests That Governments May Fake SSL Certificates" (March 24, 2010), *https://www.eff.org/deeplinks/2010/03/researchers-reveal-likelihood-governments-fake-ssl*

10. This is known as the Caesar cipher.

11. Famous for this are the code breakers of Bletchley Park, who during World War II managed to break the encryption algorithms of the Nazi ciphering machines Enigma and Geheimschreiber (secret teleprinter). In the future, a quantum computer could break the cryptographic systems in use today. However, a quantum computer would also allow for building and using quantum cryptographic systems.

12. The NSA spends US $250 million per year to insert backdoors into software and hardware, according to "Secret Documents Reveal N.S.A. Campaign Against Encryption," *New York Times*, September 5, 2013. The cryptographic algorithm Dual Elliptic Curve Deterministic Random Bit Generator (Dual_EC_DRBG) has been known to contain several backdoors that the NSA knew about. In 2017 the International Standards Organization (ISO) did not approve two encryption algorithms created by the NSA: SIMON and SPECK.

13. Veronika Stolbikova, "Can Elliptic Curve Cryptography Be Trusted? A Brief Analysis of the Security of a Popular Cryptosystem" (May 1, 2016), *https://www.isaca.org/Journal/archives/2016/volume-3/Pages/can-elliptic-curve-cryptography-be-trusted.aspx*

14. Wikipedia, "Alleged NSA interference: IPsec" (September 7, 2013), *https://en.wikipedia.org/wiki/IPSec#Alleged_NSA_interference*

15. There are many examples of DNS blocks: China blocks websites such as *https://torproject.org*. Europe blocked The Pirate Bay in 2017. Germany's Federal Department for Media Harmful to Young Persons has blocked a list of 3,000 domain names. At the request of subscribers, UK ISPs are required by law to prevent persons under age 18 from accessing hundreds of thousands of sites. Europol implements the Child Sexual Abuse Anti Distribution Filter (CSAADF) to block images of child sexual abuse.

16. In the first week of May 2020, Google detected 59,557 unsafe websites and 871 malware sites. On May 17, 2020, Google's Safe Browsing feature deemed a total of 1,915,195 sites dangerous. See graph at Google Transparency Report, "Google Safe Browsing," *https://transparencyreport.google.com/safe-browsing/*

17. On June 29, 2020, a total of 4,683,688,889 URLs have been requested to be delisted from Google's search results. Source: Google Transparency Report, "Content Delistings Due to Copyright," *https://transparencyreport.google.com/copyright/reporters/.* These removal requests were made by 21,407 copyright owners. Source: Google Transparency Report, "Explore the Data: Copyright Removal Request Data," *https://transparencyreport.google.com/copyright/explore*

18. For example, between May 3 and May 17, 2020, Google received requests to delist 6,982 URLs. In June 2020, Google delisted on average 46.5 percent of the requested URLs from its search results. Source: Google Transparency Report, "Requests to Delist Content Under European Privacy Law: Delisting URLs from Google Search for Privacy," *https://transparencyreport.google.com/eu-privacy/*

19. Between January 1 and June 30, 2019, a total of 16,947 government requests were recorded. Between July 1 and December 31, 2019, a total of 13,354 government requests were recorded. This adds up to a total of 30,301 requests for the year 2019. Source: Google Transparency Report, "Government Requests to Remove Content: Removal Requests by the Numbers," *https://transparencyreport.google.com/government-removals/*

20. David Kravets, "IP Cloaking Violates Computer Fraud and Abuse Act, Judge Rules" (August 28, 2013), *https://www.wired.com/2013/08/ip-cloaking-cfaa*

21. Brunton and H. Nissenbaum, "Vernacular Resistance to Data Collection and Analysis: A Political Theory of Obfuscation," *First Monday* 16, no. 5 (2011), doi:10.5210/fm.v16i5.3493.

22. On July 4, 2020, the Tor network counted 6,451 Tor nodes. Within the first half of the year 2020, the number of nodes fluctuated between 6,000 and 7,000. Source: The Tor Project, Tor Metrics, "Servers," *https://metrics.torproject.org/networksize.html*

23. Ibid.

24. Norbert Wiener, *Cybernetics: Or Control and Communication in the Animal and the Machine* (Cambridge, MA: MIT Press, 1948).

25. Florian Saurwein, Natascha Just, and Michael Latzer, "Algorithmische Selektion im Internet: Risiken und Governance automatisierter Auswahlprozesse" (2017), kommunikation @ gesellschaft, 18, 1–22, *https://nbn-resolving.org/urn:nbn:de:0168-ssoar-51466-4*. Also see Michael Latzer, Katharina Hollnbuchner, Natascha Just, and Florian Saurwein, "The Economics of Algorithmic Selection on the Internet" (2014), doi: 10.5167/uzh-100400, p. 6.

26. Latzer et al., p. 2.

27. Latzer et al., p. 20.

28. See Francesca Musiani, "Governance by Algorithms" (2013), *Internet Policy Review* 2 (no.3), doi: 10.14763/2013.3.188, p. 3.

29. Comparison made by Christopher Wylie, the whistleblower who revealed the Cambridge Analytica data scandal, in "Man kann das Internet reparieren," interviewed by Moritz Honert, *Tagesspiegel*, June 14, 2020.

30. This list is inspired by Latzer et al., p. 29.

31. General Data Protection Regulation (GDPR) of the European Union, EU 2016/678, *https://gdpr-info.eu*

32. Latzer et al., p. 30.

33. Parts of this chapter are derived from and inspired by Corinne Cath, Niels Ten Oever, and Daniel O'Maley, *Media Development in the Digital Age* (March 2017), *https://www.cima.ned.org/publication/media-development-digital-age-five-ways-engage-internet-governance/*

34. ITU estimates that at the end of 2019, 53.6 percent of the global population was using the internet. *https://www.itu.int/en/ITU-D/Statistics/Pages/stat/default.aspx*

35. See documents released alongside Glenn Greenwald's *No Place to Hide: Edward Snowden, the NSA, and the U.S. Surveillance State* (Metropolitan Books, 2014), *http://glenngreenwald.net/#BookDocuments*

36. Mike Robbuck, "CEOs of Ericsson and Cisco Promise an End-to-End Networking Wonderland" (November 9, 2015), *https://www.sdxcentral.com/articles/news/ceos-of-ericsson-cisco-promise-an-end-to-end-networkingwonderland*

37. You can see a diagram of the company affiliation of the authors of IETF standards here: *https://www.arkko.com/tools/rfcstats/companydistr.html*

38. The Tao of IETF introduces the reader to the "ways of the IETF": it conveys the inner workings of IETF meetings and Working Groups, discusses organizations related to the IETF, and introduces the standards process. The Tao is not a formal IETF process document but an informal and informational overview. For details, see Internet Engineering Task Force, "The Tao of IETF: A Novice's Guide to the Internet Engineering Task Force" (November 8, 2018), *https://www.ietf.org/about/participate/tao*

39. The website Public Interest Tech—edited by American cryptographer, computer security professional, and privacy specialist Bruce Schneier—provides resources on public-interest technology: a definition, links to writings, and links to organizations that are engaged in the field of public-interest technology, such as internet governance, as well as links to educational resources. See *https://public-interest-tech.com*

KEYWORD INDEX

The Internet, Demystified

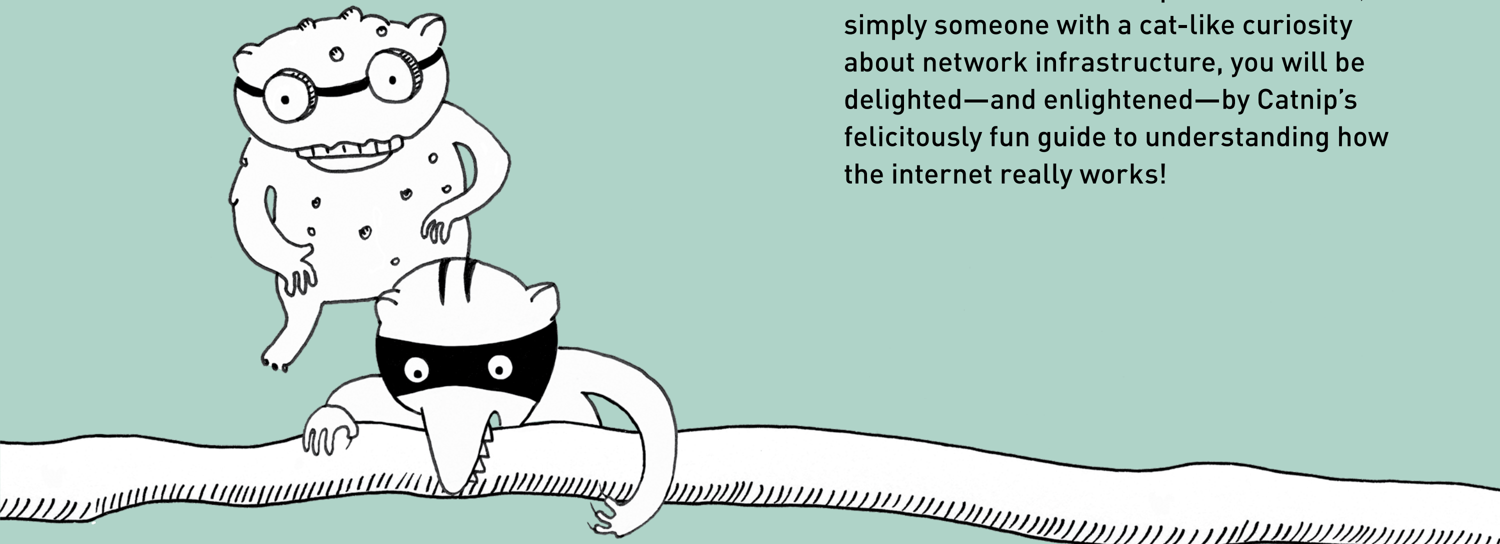
The internet has profoundly changed interpersonal communication, but most of us don't really understand how it works. What enables information to travel across the internet? Can we really be anonymous and private online? Who controls the internet, and why is that important? And . . . what's with all the cats?

How the Internet Really Works answers these questions and more. Using clear language and whimsical illustrations, the authors translate highly technical topics into accessible, engaging prose that demystifies the world's most intricately linked computer network. Alongside a feline guide named Catnip, you'll learn about:

- The "How-What-Why" of nodes, packets, and internet protocols
- Cryptographic techniques to ensure the secrecy and integrity of your data
- Censorship, ways to monitor it, and means for circumventing it
- Cybernetics, algorithms, and how computers make decisions
- Centralization of internet power, its impact on democracy, and how it hurts human rights
- Internet governance and ways to get involved

This book is also a call to action, laying out a roadmap for using your newfound knowledge to influence the evolution of digitally inclusive, rights-respecting internet laws and policies.

Whether you're a citizen concerned about staying safe online, a civil servant seeking to address censorship, an advocate addressing worldwide freedom of expression issues, or simply someone with a cat-like curiosity about network infrastructure, you will be delighted—and enlightened—by Catnip's felicitously fun guide to understanding how the internet really works!



no starch press

THE FINEST IN GEEK ENTERTAINMENT™

www.nostarch.com

$19.95 US / $25.95 CDN

ISBN 978-1-7185-0029-7

51995

9 781718 500297