so much of life today relies on the Internet — so much so that Internet shutdowns are sometimes used as tools of oppression. Keep that in mind today as we talk about the history of the Internet. Was it originally designed to be this crucial to modern life?

Tracking Internet Shutdowns in 2023

Robbie Mitchell
Senior Communication and Technology Advisor, Internet Society

January 11, 2024

It is increasingly common for governments to shut down the Internet on a national or sub-national level to solve specific problems, including controlling civil unrest, stemming the flow of misinformation, or preventing cheating on national exams.

As of the end of 2023, governments and other actors across 18 countries intentionally disrupted Internet connectivity or blocked access to specific Internet services for their citizens. Of the 124 events Pulse tracked across the year, including four that continued from last year, 55 have been nationwide disruptions lasting from a couple of hours to a week, culminating in more than 2,370 days of disruptions.

how do modules of a system communicate if they’re on separate machines?

Client

Class Browser
(on machine 1)

```python
def main():
    html = browser_load_url(URL)
    ...
```

```python
def browser_load_url(url):
    msg = url # could reformat
    send request
    wait for reply
    html = reply # could reformat
    return html
```

Server

Class Server
(on machine 2)

```python
def server_load_url():
    ...
    return html
```

```python
def handle_server_load_url(url):
    wait for request
    url = request
    html = server_load_url(URL)
    reply = html
    send reply
```
how do modules of a system communicate if they’re on separate machines?

point-to-point links: get a source to talk to a directly-connected destination

link

communication between two directly-connected nodes

examples: ethernet, bluetooth, 802.11 (wifi)
how do modules of a system communicate if they’re on separate machines?

**switches:** help forward data to destinations that are far away

switches do other things, too

**link**

communication between two directly-connected nodes

examples: ethernet, bluetooth, 802.11 (wifi)
how do modules of a system communicate if they’re on separate machines?

As this system grows, we need to think about how to turn this set of **links** into a **network**

<table>
<thead>
<tr>
<th>application</th>
<th>the things that actually generate traffic</th>
</tr>
</thead>
<tbody>
<tr>
<td>transport</td>
<td>sharing the network, reliability (or not)</td>
</tr>
<tr>
<td>network</td>
<td>naming, addressing, routing</td>
</tr>
<tr>
<td>link</td>
<td>communication between two directly-connected nodes</td>
</tr>
<tr>
<td></td>
<td><em>examples: ethernet, bluetooth, 802.11 (wifi)</em></td>
</tr>
</tbody>
</table>
1970s:
ARPAnet

**application**
the things that actually generate traffic

**transport**
sharing the network, reliability (or not)

**network**
naming, addressing, routing

**link**
communication between two directly-connected nodes
examples: ethernet, bluetooth, 802.11 (wifi)

**FIGURE 6.1** Drawing of September 1969 (Courtesy of Alex McKenzie)

https://personalpages.manchester.ac.uk/staff/m.dodge/cybergeography/atlas/historical.html
1970s: ARPAnet

- **application**: the things that actually generate traffic
- **transport**: sharing the network, reliability (or not)
- **network**: naming, addressing, routing
- **link**: communication between two directly-connected nodes
  - examples: ethernet, bluetooth, 802.11 (wifi)

1970s:
ARPAnet

hosts.txt

1970s:
ARPAnet

application the things that actually generate traffic

transport sharing the network, reliability (or not)

network naming, addressing, routing

link communication between two directly-connected nodes

examples: ethernet, bluetooth, 802.11 (wifi)

https://personalpages.manchester.ac.uk/staff/m.dodge/cybergeography/atlas/historical.html
1970s:
ARPAnet

hosts.txt

ARPANET LOGICAL MAP, MARCH 1977

application the things that actually generate traffic

transport sharing the network, reliability (or not)

network naming, addressing, routing

link communication between two directly-connected nodes
equations: ethernet, bluetooth, 802.11 (wifi)

https://personalpages.manchester.ac.uk/staff/m.dodge/cybergeography/atlas/historical.html
1970s:
ARPAnet

hosts.txt

application the things that actually generate traffic
transport sharing the network, reliability (or not)
network naming, addressing, routing
link communication between two directly-connected nodes
examples: ethernet, bluetooth, 802.11 (wifi)

1970s:
ARPAnet

hosts.txt  distance-vector routing

ARPANET LOGICAL MAP, MARCH 1977

application  the things that actually generate traffic

transport  sharing the network, reliability (or not)

network  naming, addressing, routing

link  communication between two directly-connected nodes

examples: ethernet, bluetooth, 802.11 (wifi)

https://personalpages.manchester.ac.uk/staff/m.dodge/cybergeography/atlas/historical.html

Katrina LaCurts | lacurts@mit.edu | 6.1800 2024
1970s: ARPAnet hosts.txt distance-vector routing
1978: flexibility and layering TCP, UDP
early 80s: growth → change

With a **layered model**, we can swap out protocols at one layer without much (or perhaps any) change to protocols at other layers.

https://personalpages.manchester.ac.uk/staff/m.dodge/cybergeography/atlas/historical.html
1970s: ARPAnet
1978: flexibility and layering
early 80s: growth → change
late 80s: growth → problems

| hosts.txt | distance-vector routing | TCP, UDP | OSPF, EGP, DNS |

application: the things that actually generate traffic
transport: sharing the network, reliability (or not)
examples: TCP, UDP
network: naming, addressing, routing
examples: IP
link: communication between two directly-connected nodes
examples: ethernet, bluetooth, 802.11 (wifi)

with a layered model, we can swap out protocols at one layer without much (or perhaps any) change to protocols at other layers

https://personalpages.manchester.ac.uk/staff/m.dodge/cybergeography/atlas/historical.html
With a **layered model**, we can swap out protocols at one layer without much (or perhaps any) change to protocols at other layers.
with a layered model, we can swap out protocols at one layer without much (or perhaps any) change to protocols at other layers.
on the Internet, we have to solve all of the “normal” networking problems (addressing, routing, transport) at massive scale, while supporting a diverse group of applications and competing economic interests