Cloud Computing

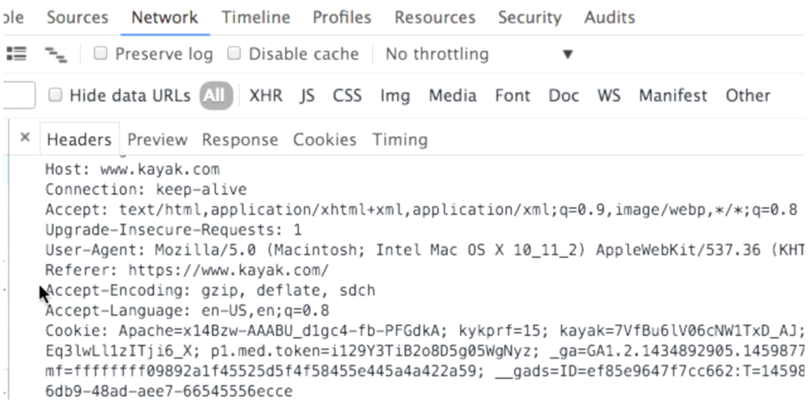
This is CS50. Harvard University. Fall 2015.

Cheng Gong

Table of Contents

Recap

- Our laptop uses DHCP to get an IP address, the IP address of a DNS server, and our default router to send messages to first. And to visit a website, we might use HTTP, as well as TCP to make sure we receive the response.
- HTTPS is the secure version of HTTP, using technologies like SSL or TSL, which just include some kind of public-key cryptography that allows our browser to establish an encrypted connection with the web server.
- With Chrome’s incognito window, our computer won’t store our internet history locally, but the disclaimer text does indicate that "the websites you visit" might.
- Websites also tend to leave cookies, or small text files with some long random identifier, which our browser sends back every time we visit that website again, so that the website might be able to remember that we once logged in.
  - If we open the `Network` tab as we visit a website as we did before, we might see a screen like this:



```
Host: www.kayak.com
Connection: keep-alive
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_11_2) AppleWebKit/537.36 (KHT
Referer: https://www.kayak.com/
Accept-Encoding: gzip, deflate, sdch
Accept-Language: en-US,en;q=0.8
Cookie: Apache=x14Bzw-AAABU_d1gc4-fb-PFGdkA; kykprf=15; kayak=7VfBu6lV06cNW1TxD_AJ;
Eq3lwLl1zITji6_X; p1.med.token=i129Y3TiB2o8D5g05WgNyz; _ga=GA1.2.1434892905.1459877
mf=fffffff09892a1f45525d5f4f58455e445a4a422a59; __gads=ID=ef85e9647f7cc662:T=14598
6db9-48ad-aee7-66545556ecce
```

  # The `Cookie` part of the request shows us the identifier we might be assigned. And if we clear our cookies manually, or use an incognito window, then we would receive a new cookie with a new identifier.

- And IP addresses might be limited to 4 billion possibilities, but those are public IP addresses. Within an organization, there are many private IP addresses for its devices, which means they are essentially sharing the same IP address to the outside world. The router is doing this for us through a mechanism called NAT, network address translation, where the requests come from different private IP addresses but are sent out with the same public IP address, and then the responses are redistributed to the correct private IPs once the router receives them.

- With an HTTPS connection (the green lock icon in the address bar), the internet service provider will still be able to see the final IP address we are trying to reach, but no longer what pages or the content of pages we are on.

  # To have an SSL connection, we also need to go to some third-party such as Verisign, who verifies the legitimacy of our servers, such that our visitors' browsers can trust our website with that green icon.

- And advertisers, who have ads on different websites, will know the websites we have been on, since they too can leave cookies as well as see our IP address when we request an ad that’s included in a website.

- We received this question: Does your IP Address change every time you turn on/off your Wifi or device? Or is there a permanent IP address associated with your device that advertisers / servers could potentially identify you through?

  # And some places, like on campus, you’re probably assigned a new IP address every time. But your home internet connection might be assigned (by Comcast or other provider) what is known as a static IP, which changes much less frequently.

- We take a look at this quick video[1] as a reminder that laptops can be useful tools but hopefully not too distracting in class!

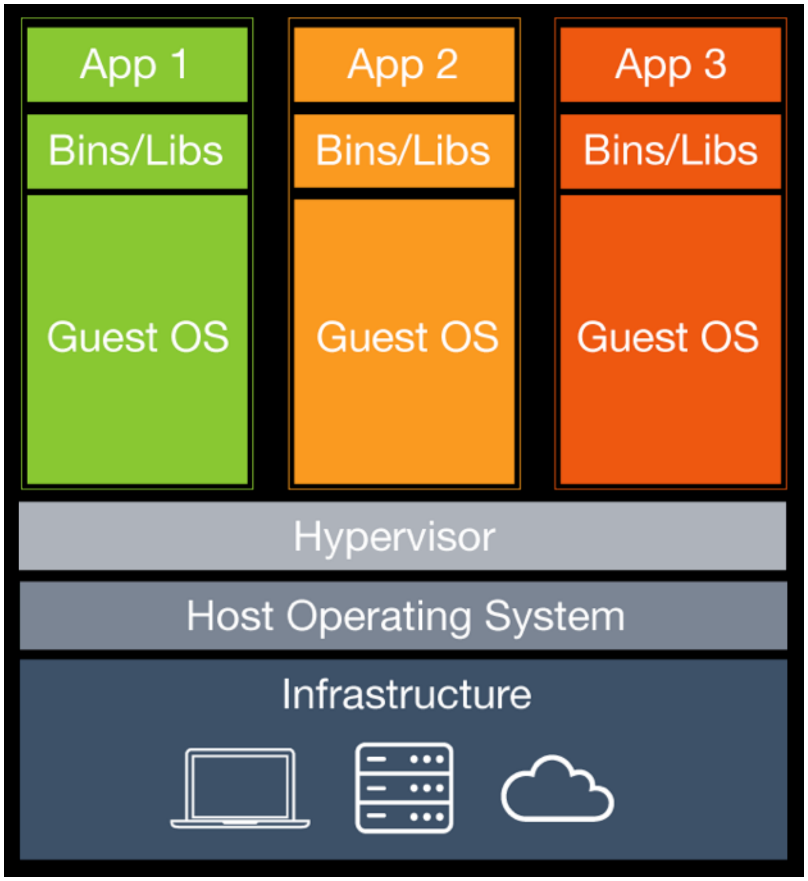
Cloud Computing

- Cloud computing means that we’re using computing resources from someone or somewhere remotely, rather than having our own hardware, data center, etc.

- As hardware gets faster and we have more cores or CPUs on the same physical machine, it makes more sense to split up the computing power among different people, so it feels like everyone has their own machine. This is accomplished through software called hypervisors that run multiple copies of Windows or Linux simultaneous, so we have multiple virtual machines.

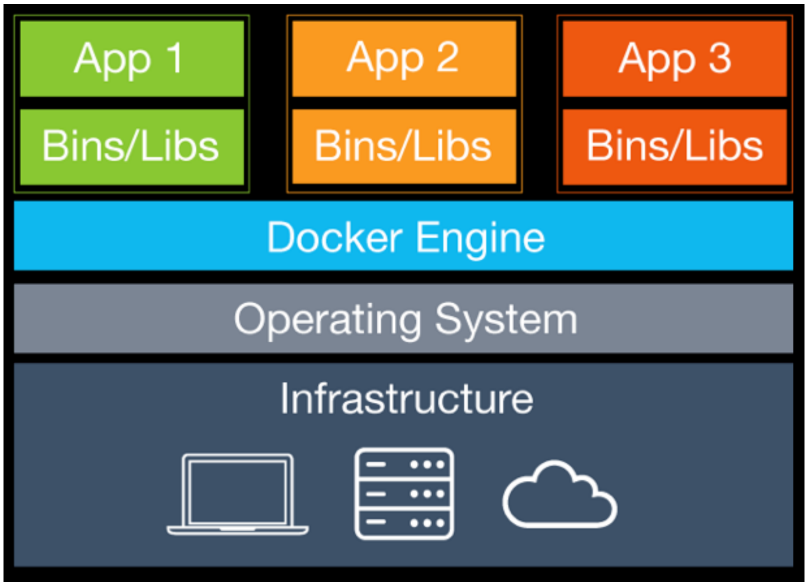
- We can visualize this scheme with a picture like this:

[1] https://www.youtube.com/watch?v=jkRWYC28-Nw



  # Our infrastructure is our hardware, the host operating system is what’s running on the machine, and the hypervisor is the special software made to run three or more virtual machines on top, each of which are separated from each other.

- We can install that software on our own computers, too, to run many operating systems like Windows and Linux simultaneously.
- There’s even fancier technology that allow for this virtual separation of user applications running on the same machine, using containers:



# The advantage of this would be better performance, since we no longer need to run multiple entire operating systems.

- So if we are a startup with lots of customers, and have our own server at our office. Eventually, we'll have too many users to handle with our one server, and each of them will see slow service or or even no service. So we might buy a second server, but that takes time to set up, and we might be wasting the extra resources during non-peak times.
- One advantage to using cloud computing is that we can scale up and down much more easily and quickly, since the provider like Amazon or Google already has those servers set up with hardware and software.
- Once we have many servers, we need to figure out which ones to send users to. We could send each user to the next server in order, or a random server each time, but that could still result in one server having much more load than the rest. So we can have one (or more) machines called load balancers that ask servers periodically how busy they are, and send users to the least busy ones.
- DNS can't solve this problem because it has caching, saving the results so that future requests for the same domain are faster, but that also means that users will be trying the same IP address for some minutes or hours, so it can't be used as a load balancer.
- But if users are sent to different servers each time, temporary information like their shopping cart might not be copied over between servers' memory. So next time we

might need to send users to the same server they visited before. We can accomplish this not with their IP (since many users might share the same public IP) but cookies.

- And if that load balancer fails, no one can access anything, so maybe we could have multiple load balancers. But DNS is still caching the IP of one of those load balancers, so to solve this issue we allow for our load balancers to all share IP addresses. This way, if one of them fails, one of the others can take on its IP and respond to users.
- But now we have another problem of users needing to visit the same servers every time. We could duplicate their hard drives or memory, but as we have more and more servers that would take a lot more time. So we could have yet another server that remembers the temporary data, which we can call the session or state, that all the web servers (which serve the identical parts of the web pages) can talk to.
- But then that server would be the single point of failure, so we might want yet another one. And these challenges are why it is so difficult to have 100% uptime.
- And cloud computing solves the problems of us having to wire new machines ourselves, but rather have some provider do it for everyone 24/7, and worry about how to solve these challenges as best as possible.
- CS50, for example, uses Amazon Web Services with a data center in West Virginia, and not worry too much about failures since the data is indeed backed up perhaps every few hours, and copy that data over to another data center if there is a larger-scale failure.
- IAAS, PAAS, and SAAS are just acronyms for infrastructure, platform, and software as a service, where some providers manage the data centers, virtual machines or containers, or applications for you.
- If we go to `aws.amazon.com`, we can see many services that Amazon provides. EC2, for example, allows us to rent a virtual server, or part of a physical server. And the pricing is per hour, so we can flexibly scale up and down our website.
- Microsoft Azure and Google Compute Engine are other similar services that might be compared.
- Now that we have our web servers set up, we need to store data from our users more permanently. So we'll look at database technologies like SQL and noSQL next time!