Reverse Proxies and Content Delivery Networks

COMP3220 Web Infrastructure

Dr Heather Packer – hp3@ecs.soton.ac.uk
Reverse Proxy

- A reverse proxy requests network resources on behalf of a client from one or more servers
- Resources are then returned to the client and appear to originated from the reverse proxy
- Reverse proxies does not require any client-side configuration
- A proxy is associated with clients
- A reverse proxy is associated with servers
Uses of Reverse Proxies

• Security – Reverse proxies can:
  • hide the existence and characteristics of origin servers
  • implement an application firewall can have a single point to monitor and handle attacks, and find and remove malware
  • add HTTP access authentication to a web server

• Load - Reverse proxies can:
  • cache content- compress content
  • distribute requests to several servers
  • provide large content to a client on time
Risks of Reverse Proxies

• Decrypt and re-encrypt https traffic
• Has access to TLS certificates' private keys
• Single point of failure to provide access to web servers
Motivation Scenario

Stream video content to 100,000+ simultaneous users

• You could use a single large “mega-server”
  • Single point of failure
  • Point of network congestion
  • Long path to distant clients
  • Multiple copies of video sent over outgoing link

• This solution doesn’t work in practice
Content Delivery Network

• A geographically distributed network of proxy servers (edge nodes)
• Hosts static content (such as images, CSS and JS)
• Data travels to user via the shortest path (reduced latency)
Commercial CDNs

- Limelight Networks
- Level 3 Communications
- Akamai Technologies
- Amazon CloudFront (try cloudping.info to see latencies)
- CloudFlare
Motivational Scenario

Streaming video to 100,000+ simultaneous users

• Working Web solution: store/serve many copies of video at multiple geographically distributed sites (CDN)

• Two strategies:
  1. Push CDN servers deep into many access networks
     • Close to users
     • Placed near ISP
     • Used by Akamai, 1700 locations
  2. Place larger clusters at key points in the network near internet exchanges
     • Internet exchanges where network providers connect their networks to each other
     • Dedicated high speed private networks are used to connect the clusters together
     • Used by Limelight
Motivational Scenario

Streaming video to 100,000+ simultaneous users

• Working Web solution: store/serve many copies of video at multiple geographically distributed sites (CDN)

• Two strategies:

  1. **Push CDN servers deep into many access networks**
     
     Better latency and better network performance.
     
     Harder to maintain because there are many more servers in the CDN.

  2. **Place larger clusters at key points in the network near internet exchanges**
     
     Higher latency and lower performance for the end user
     
     Easier to manage less servers and less locations.
CDN: Simple content access scenario

- A CDN has to be able to tell clients where to find resources
- A client will request a file, with one URL but retrieve it from another server

http://video.netcinema.com/6Y7823V
1. Requests URL for video
http://video.netcinema.com/6Y7B23V
2. Resolves Domain name via local DNS
   video.netcinema.com
3. Resolves video.netcinema.com
   As CNAME kingcdn.com

4. Resolves Domain name
   KingCDN.com

5. Returns IP of KingCDN
6. Contacts IP address and requests URL
http://video.netcinema.com/6Y7B23V
CDN Cluster Selection Strategy

• The CDN’s DNS decides which edge server to use
  • Pick CDN node geographically closest to client
  • Pick CDN node with shortest delay (min hops) to client (CDN nodes periodically ping access ISPs, report results to CDN DNS)

• Or let the Client decide – give client a list of several CDN servers
  • Calculates best edge server to use
  • Selects which edge server based on the required bit rate and latency
  • Continually probes the best way of receiving content
  • Known as “Client Intelligence”
Case Study: Netflix’s first Approach

• Owned very little infrastructure, used 3rd party services
  • Own registration, payment servers
  • Amazon (3rd party) cloud services
    • Netflix uploads studio master to Amazon cloud
    • Create multiple version of movie (different encodings) in cloud
    • Upload versions from cloud to CDNs
  • Three 3rd party CDNs host/stream Netflix content: Akamai, Limelight, Level-3
Case Study: Netflix

1. Bob manages Netflix account

2. Bob browses Netflix video

3. Manifest file returned for requested video

4. DASH streaming

upload copies of multiple versions of video to CDNs

Amazon cloud

Akamai CDN

Limelight CDN

Level-3 CDN
DASH - Dynamic Adaptive Streaming over HTTP

• Server
  • Divides video files into multiple chunks
  • Each chunk stored encoded at different bit rates
  • Manifest file: provides URLs for different chunks

• Client
  • Periodically measures server-to-client bandwidth
  • Consulting manifest, requests one chunk at a time
  • Chooses maximum bit rate chunk given current bandwidth
  • Can choose different coding rates at different points in time (depending on available bandwidth at time)

• The Intelligence happens at the client level so it can make sure that there is no buffer starvation or overflow
MPEG-DASH Adoption

• MPEG DASH is independent, open and international standard, which has broad support from the industry
• Heavy plugins like Silverlight and Flash perform poorly and cause security issues
• HTML5 Media Source Extensions includes MPEG-DASH support

• Chrome dropped the Silverlight support,
  • It was a problem for the majority of premium video providers,
  • They delivered their streams via Smoothstreaming and Playready DRM, which enforced Silverlight
  • These providers switch to using HTML5 with MPEG-DASH and MPEG-CENC based DRM
Netflix OpenConnect CDN

• Netflix wanted the absolute best streaming they could get, while lowering cost
• They developed OpenConnect
• High optimised for delivery large files, still use Akamai for small assets.
• Data centers around the world
  • There may be a data center with a couple of racks that contain the entire Netflix library
  • Others might only have 80% of the most popular content.
• Unpopular material will have to travel further
• Uses Client Intelligence approach
Learning Outcomes

• How reverse proxies route requests
• The infrastructure of Content Delivery Networks
• CDNs in practice