Bonusuppgift - TDTS06 Datornät
Geographic Mapping of Services

Niclas Olofsson (nicol271)

5 oktober 2012

1 Inledning

1.1 Sammanfattning

Målet med uppgiften var att gå igenom de 100 populäraste sidorna på internet, och analysera de requests som skickas om man besöker deras förstasidor. Utifrån detta kunde bestämmas att totalt 538 olika servrar kontaktades, och 344 av dessa var servrar som minst en gång kontaktades av en sida som inte var ägare till servern. 65 av de 100 sidorna anropade minst en gång en Google-ägd server. De allra flesta av de kontaktade servrarna låg i USA eller Kina, en del servrar fanns även i Europa.

1.2 Tolkning av uppgiften

Uppgiften var att från alexa.com hämta en lista över de 100 populäraste sidorna på internet. Sedan skulle förstasidan för varje sådan sida besökas, och ett antal frågor besvaras utifrån den totala mängden data som skickades i och med dessa requests.

1.2.1 Hur många servrar kontaktades totalt?

För det första vore det orimligt att tro något annat än att uppgiften avser besök med en webbläsare och inte bara en enkel HTTP-request utan att sedan hämta några bilder eller liknande. Man skulle dessutom kunna se varje server som antingen ett unikt hostname, eller som en unik ip-adress. Vi valde att testa båda varianterna, skillnaden visade sig inte vara jättestor. Enbart unika hostnames räknades.

1.2.2 Hur många servrar ägs av ett annat företag än företaget som äger den server man kontaktade från början?

Vår definition här var att om en viss sida (t.ex. facebook.com) gör en request till någon server som har en annan ägare än den aktuella sidan, så uppfyller servern detta villkor. Enbart unika servrar räknades. Ägarskapet till en viss server definierades genom det företag som stod registrerat på hostnamnet, eller själva

hostnamnet i de få fall detta inte var tillgängligt.

1.2.3 Hur många sidor kontaktar minst en server som ägs av Google?

Som frågan anger räknade vi här vilka sidor som medför minst en request till en Google-ägd server. Ägarskapet till en server definierades som i föregående fråga.

1.2.4 Hur många servrar använde gzip-kompression?

Vi definierade detta som antalet servrar som någon gång skickade ett svar med gzip-encoding.

1.2.5 Hur mycket utrymme sparade gzip i dessa fall?

Vi har här räknat den datamängd (bytes) som sparades genom gzip-encoding jämfört med om dessa responses inte använt gzip.

1.2.6 Vart och hur långt bort i världen finns servrarna?

Med tanke på titeln på uppgiften vore det orimligt att tro något annat än att den geografiska positionen aves. Med "hur långt bort"valde vi att utgå från Linköping som referens.

2 Utförande

2.1 Skicka requests till de pouläraste sidorna

Som uppgiften föreslog använde vi oss av topplistan över sidor på alexa.com. Denna kan exporteras som en csv-fil, vilken enkelt kunde läsas in som en vanlig textfil. För att utföra själva hämtningen av en viss sida användes Python-biblioteket Spynner. Detta bibliotek använder sig av Webkit för att simulera en webbläsare, inklusive t.ex. hämtning av statiskt innehåll såsom bilder och CSS, samt mer avancerat innehåll som sådant som hämtas via t.ex. AJAX-requests. Det var ett enkelt sätt att programmatiskt kunna göra requests mot sidorna och fortfarande få samma resultat som om de hämtats manuellt med en webbläsare. Dessutom slapp vi problem med en eventuell cache som eventuellt skulle vara tvungen att rensas mellan varje sida (eller i alla fall körning) för att få korrekta resultat.

Valet av verktyg för insamlingen av requests föll på Wireshark. Ett problem var dock att vissa av frågorna kräver att det går att skilja på de olika webbsidorna, vilket kan bli svårt att ta reda på i efterhand. Särskilt i de fall där namnet på sidan inte stämmer överens med namnet på den första servern som kontaktas. Lösningen blev förvisso inte jättefin, men enkel. Mellan varje sida skickades en request till en viss förutbestämd sida som inte fanns med på listan och som svarade med ett väldigt okomplicerat svar. Valde en icke-existerande sida på min egen server, som skulle svara med ett 404-fel: niclasolofsson.se/mupp.

Ett script (se Appendix A) användes för att besöka sidorna, under tiden som insamlingen skedde. En av sidorna (avg.com) gick av okänd anledning inte att besöka med hjälp av scriptet, denna besöktes istället manuellt i en webbläsare. En annan av de sidor som vid uppgiftens påbörjande fanns med på Alexa topp 100 (360.cnp) har inte varit tillgänglig över huvud taget, och vi har därför helt bortsett från denna.

Såg till att inte ha några andra störande program igång samtidigt. Resultatet i Wireshark filtrerades på "http", vilket räckte för att bara få den trafik som vi önskade. Detta exporterades från Wireshark till en textfil med hjälp av Print... -> selected packets", packet summary line"vilket låg till grund för analysen till de frågor som inte var gzip-relaterade.

2.2 Analys av data

Vi valde att först fokusera på de frågor som kunde lösas med bara kännedom om de hostnames som kontaktades. Ett annat Python-script (Appendix B) användes för att läsa in den exporterade textfilen och listan på populära sidor. Varje av de 100 sidorna innehöll en lista på de hostnames som sidan kontaktat.

För att kunna bestämma geografisk position använde vi oss av API:et ipaddresslabs.com, som tillhandahåller IP och hostname-lookups inklusive geografisk position (och framför allt går att använda med en begränsad gratislicens). Skrev ett tredje script som gjorde requests mot detta API för alla insamlade hostnames, och dumpade ut resultatet som XML i en fil. Utökade sedan analys-scriptet för att parsa denna data så att varje hostname hade geografisk information. Som en liten bonus fanns även information om vilket företag som ägde varje hostname, samt till vilken IP-adress som denna pekade.

Sedan var det bara att traska igenom alla sidor och sammanställa svaren på frågorna, utifrån tolkningen av dessa ovan. För att kunna plotta resultatet på en karta användes verktyget www.darrinward.com/lat-long. Hittade senare ett bättre verktyg, www.batchgeo.com som användes för att generera en bättre karta.

Avståndet till alla servrar beräknades med hjälp av en implementation av Haversineformeln. Denna finns i med scriptet i Appendix B.

2.3 Lösning av gzip-uppgifterna

3 Resultat

3.1 Totalt antal kontaktade servrar

Om man lägger ihop det totala antalet servrar som varje sida kontaktar blir antalet 917 st. Många av dessa servrar verkar dock upprepas; det totala antalet unika servrar som kontaktas är 544 stycken om man räknar varje hostname som en server, eller 538 stycken om man även kräver att servrarna ska ha olika IP-adresser.

3.2 Tredjeparts-servrar

Totalt är 344 av 538 servrar sådana som ägs av ett annat företag än det företag som är ägare till sidan man besöker. Av de 100 sidorna anropar 65 st någon gång en server som ägs av Google.

3.3 Användande av gzip-komprimering

Av de 538 besökta servrarna så använde 242 av dessa gzip-komprimering minst en gång. Resultatet kan i mitt tycke verka ganska lågt. Gjorde en rimlighets-analys genom att räkna antalet totala requests (cirka 5800 st) samt antalet responses med gzip-kompression (cirka 1300 st). Med andra ord verkar resultatet inte vara helt fel ute med tanke på den ringa mängden svar med gzip i förhållande till antalet requests.

Utan gzip hade servrarna som nu använde gzip gjort av med 44 Mb data. Nu används istället 12 Mb, och alltså sparades 32 Mb (72 %) data!

3.4 Servrarnas geografiska position

En plot av alla de servrar som kontaktades se i Figur 3. Man kan dock misstänka att åtminstone någon av de koordinater som fåtts från Geo-API:et är felaktig, då till exempel annars skulle ha en server i vattnet utanför Maroccos kust.

En något tydligare översikt över de mest server-täta områdena på kartan finns i Figur 4. En onlineversion av denna karta över reultatet finns (oktober 2012) på http://bit.ly/SBejYE, där man dessutom t.ex. kan kika på hostnamnet för varje punkt om man så vill.

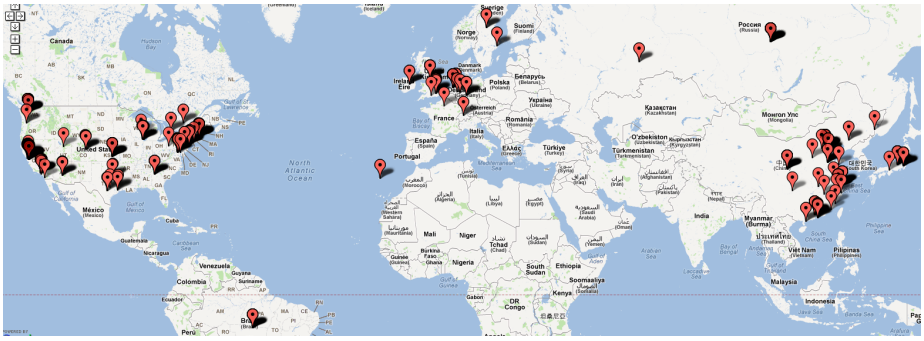
Man kan även göra en tabell över hur många servrar i ett land som kontaktas för att få en bättre överblick av tätheten, vilken kan ses i Figur 1. Vi kan notera att för en server fanns ingen exakt information om landet, bara att den låg i Europa.

Land	*Antal servrar*
Brazil	10
Canada	4
China	135
Europe	1
France	1
Germany	8
Hong Kong	4
Ireland	4
Japan	23
Netherlands	28
Portugal	2
Russian Federation	29
Sweden	5
Switzerland	1
United Kingdom	13
United States	270
Summa	538

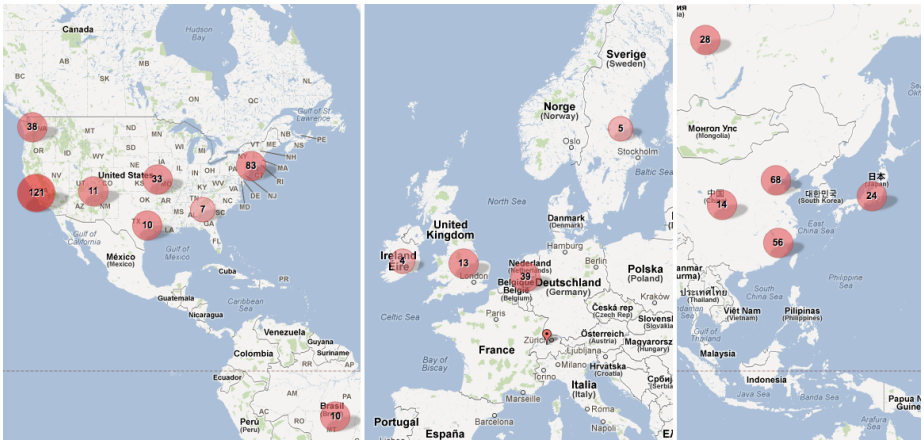
Figur 1: Tabell över antal kontaktade servrar per land

Avstånd x (km)	*Antal hosts*
$0 < x < 500$	5
$500 < x < 1000$	36
$1000 < x < 5000$	51
$5000 < x < 6000$	41
$6000 < x < 7000$	118
$7000 < x < 8000$	120
$8000 < x < 9000$	163
$9000 < x < \infty$	10
Summa	538

Figur 2: Tabell över antal servrar i förhållande till avståndet från Campus Valla



Figur 3: Karta över positionen för de kontaktade servrarna



Figur 4: Detaljbild över de geografiska områden som hade flest kontaktade servrar. En full version finns (oktober 2012) på http://bit.ly/SBejYE