Freenet - Hur kommunikationen, protokollet och P2P-arkitekturen fungerar

Joel Hogeman
http://www.dat15jho@student.lu.se
Datateknik, LTH

***Sammanfattning*—Peer-to-peer är en nätverksarkitektur som vuxit i popularitet de senaste åren, mycket pga att den används av stora fildelningsapplikationer som Bittorrent. Arkitekturen består av ett decentraliserat system med likvärdiga noder som både agerar som servrar och klienter. Freenet - ett nätverk för censurfri och säker delning och publicering av information - använder sådan peer-to-peer-arkitektur. Men Freenet lägger inte i första hand fokus på decentraliseringen i sig, utan på säkerhet och anonymitet. Nätverkets noder samarbetar genom att tillsammans lagra och routa distribuerad och krypterad data genom nätverket. Ingen nod känner till vad det är den lagrar eller var platsen för destinationen av transmitterad data ligger. Kommunikationen mellan noderna styrs av Freenets egna protokoll och sker i form av request- och reply-meddelanden som sänds i segment på transportlagret. Denna artikel beskriver Freenets kommunikationssystem genom att bl.a. titta på hur protokollet ser ut, vilka meddelanden som skickas, vad de innehåller, etc. Uppfångad trafik från och till en dator med Freenet körande, redogörs för, och förklaras baserat på det som beskrivits tidigare i artikeln.**

I. Introduktion

> *I worry about my child and the Internet all the time, even though she's too young to have logged on yet. Here's what I worry about. I worry that 10 or 15 years from now, she will come to me and say 'Daddy, where were you when they took freedom of the press away from the Internet?'*
> - Mike Godwin, Electronic Frontier Foundation

Freenet är ett censurfritt nätverk för anonym och säker kommunikation och delning av information. Detta inkluderar bl.a. fildelning, browsing och publicering av webbsidor (i Freenet kallade "freesites") och forum för chatt. Citatet ovan, taget från freenetproject.org under rubriken 'What is Freenet?', ger en bild av vad Freenets grundare ville bekämpa[1].

Många anonymitets- och säkerhetsbetonade plattformar som Tor fungerar som proxytjänster, vilket möjliggör anonym kommunikation med internet[2]. Freenet skiljer sig från dessa genom att istället använda ett eget nätverk separat från internet, vilket dess användare är uppkopplade till. För att vara en del av Freenet krävs först att huvudapplikationen, som går att ladda ned kostnadsfritt från freenetproject, installeras. Det applikationen gör efter att den laddats ned och startats, är att den kopplar datorn till Freenets nätverk. All nedladdning, freesite-browsing etc. sker sedan via en vanlig webb-browser med HTTP-gränssnitt.

Ursprungligen var Freenet en idé beskriven i ett studentprojekt av Ian Clarke, som då studerade på University of Edinburgh[3]. Projektet stod klart 1999 och Freenet realiserades sedan åren därpå av ett flertal personer, och har sedan fortsatt utvecklas och uppdateras. De fem mål som Clarke i sitt projekt beskrev som centrala för designen av Freenet, och som utgör grunden för projektet sammanfattas i följande[4]:

- Anonymitet för producenter och konsumenter av information
- Nekad tillgång till informationslagrare
- Motstånd till försök från en tredje part att neka tillgång till information
- Effektiv dynamisk lagring och dirigering av information
- Decentralisering av alla nätverksfunktioner

För att uppfylla ovannämnda principer har Freenet implementerats med s.k. peer-to-peer-arkitektur[4].

Denna artikel har som mål att beskriva hur Freenet-nätverket utnyttjar P2P för kommunikation, samt att redogöra för trafik som spårats under körning av applikationen.

II. Teori

A. Peer-to-peer

Peer-to-peer (P2P) är en nätverksarkitektur som till skillnad från "Client/Server-modellen" inte är beroende av centrala servrar för kommunikation. Istället använder sig nätverket av ett decentraliserat system där alla nätverkets noder, eller peers, agerar som både servrar och klienter genom att lagra och skicka data mellan varandra[5]. Ingen av noderna i nätverket har några speciella privilegier gentemot de övriga. Det behövs inte mer för en dator än att kopplas till en nod i ett P2P-nätverk för att bli en del av det. Därför kan ett sådant nätverk enkelt expandera utan att tappa funktionalitet. P2P har under de senaste åren blivit en allt populärare arktitektur för applikationer och har implementerats på många olika sätt. Främst har den använts av fildelningsapplikationer som Napster, Kazaa, Gnutella och Bittorrent[6].

B. *Freenet*

1) P2P-arkitekturen: En grundidé bakom Freenet är att yttrandefrihet på nätet endast går att uppnå om användare ges total anonymitet[1]. Detta gör Freenet unikt, eftersom filosofin går stick i stäv mot det traditionella P2P-konceptet om öppen kommunikation mellan peers. I övrigt är Freenet i stort sett implementerat som ett traditionellt P2P-nätverk av noder som samarbetar genom att lagra och routa datafiler. En viktig skillnad mellan Freenet och övriga P2P-nätverks-applikationer är dock att Freenets användare inte kan kontrollera vad som lagras på deras maskiner. All data i Freenets nätverk är krypterad och utspridd bland samtliga noder som var och en har reserverat ett datalager åt Freenet från sitt minne. De datafiler som lagras i nätverket har namn som fungerar som nycklar[6].

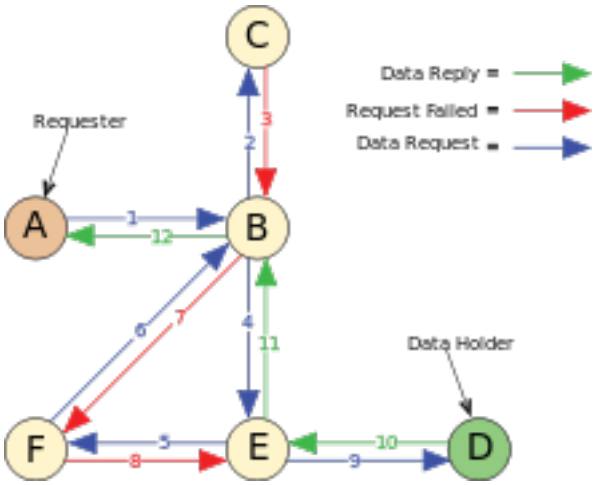
2) Protokoll och kommunikation: Kommunikation mellan noderna styrs av ett särskilt paketorienterat Freenet-protokoll, och tar sig i uttryck i form av meddelanden. Vilka olika meddelanden som Freenet-protokollet ger upphov till, visas tillsammans med beskrivningar i figur 1.

Alla meddelanden innehåller ett slumpmässigt genererat 64-bitars transaktions-ID så att noderna kan hålla koll på antalet förfrågningar. Detta gör nätverket mer flexibelt i valet av transportprotokoll, det är t.ex. kompatibelt med både TCP och UDP. Varje nod i nätverket är kopplad till ett antal grann-noder. Dessa är sparade i en dynamisk routingtabell tillsammans med de nycklar som de tros hålla. För att en nod ska få tillgång till en fil i nätverket skickar den först ett Reply.Data-meddelande, innehållande en sökt nyckel, till sig själv. Om den hittar datafilen vars namn matchar den sökta nyckeln kan filen hämtas från det egna lokala Freenet-lagret. Annars inleds en Freenet-transaktion med den nod i routingtabellen som baserat på sin nyckel tros vara närmst den sökta nyckeln.

Transaktionen inleds med ett Request.Handshake-meddelande som specificerar vilken adress som ska användas för returmeddelande. Om noden är aktiv och tar emot förfrågningar, kommer den att svara med en Reply.Handshake som bekräftar att fortsatt transaktion tillåts. Handslag cachas i ett par timmar så att det under den tiden kan ske transaktioner mellan samma noder utan upprepning av detta steg. Efter att bekräftelsen nått sändarnoden skickar den ett Reply.Data-meddelande med den sökta nyckeln. Denna process fortsätter tills en nod som lagrar nyckeln hittas, då returneras den sökta filen i ett Send.Data-meddelande tillsammans med nodens adress som källa, till noden som skickade förfrågan.

Meddelande	*Beskrivning*
Request.HandShake	Förfrågan om “Freenet peer connection“
Reply.HandShake	Svar från en peer som kan “connecta“
Reply.Data	Förfrågan om data (query)
Send.Data	Protokollmässig mekanism för nedladdning
Request.Insert	Förfrågan om insättning av data till nätverket
Reply.NotFound	Returneras då sökning eller insättning misslyckats

Figur 1. Freenetprotokollets olika typer av meddelanden[6]



Figur 2. En modell av filöverföringsprocessen i Freenet[7]

Protokoll-lager	*Beskrivning*
Applikationslagret	Bestämmer hur de faktiska meddelandena ska tolkas
Meddelandelagret	Bestämmer formatet på Freenet-protokoll-meddelandena
Kryptografilagret	Hanterar krypteringen
Transportlagret	Använder traditionella transportprotokoll som TCP eller UDP

Figur 3. Freenets olika protokoll-lager[6]

Om en nod får samma förfrågan som den tidigare fått, ignorerar den meddelandet och skickar tillbaka det till noden före som tvingas välja en annan nod från sin routingtabell att skicka det till[6]. För att undvika att en förfrågan om en nyckel som inte existerar i nätverket skickas runt i all oändlighet, innehåller det ursprungliga meddelandet ett s.k. hops-to-live-värde som minskar med ett för varje hopp i nätverket. Om gränsen når noll skickas ett Reply.Not.Found-meddelande till ursprungsnoden. En s.k. depth counter används också i dessa meddelanden. Den har i ursprungsmeddelandet ett värde satt till 0 som istället inkrementeras för varje hopp. Det värdet som den innehåller då Reply.Data-meddelandet når noden med den sökta nyckeln, kopieras till hops-to-live-värdet

för Send.Data-meddelandet så att ursprungsnoden nås[4]. Processen illustreras i figur 2.

Freenetprotokollet består utöver transportlagret av 3 ovanliggande lager (se figur 3).

III. Uppfångad trafik under körning av Freenet

A. Material

Uppfångandet av trafik genomförs med hjälp av följande material:

- En dator med den lokala IP-adressen 192.168.1.158 samt globala IP-adressen 81.236.215.142.
- En router med access till internet.
- En ethernetsladd som kopplar datorn till routern.
- Paketanalysatorn Wireshark
- Freenets huvudapplikation
- Mozilla Firefox

B. Utförande

Datorn startas och Freenet körs igång. Sedan fångas trafik från och till datorn med hjälp av capture-funktionen i Wireshark. Resultatet noteras och analyseras.

C. Resultat med noteringar

Det första som händer när Freenet startar är att Freenets startsida med alla alternativ öppnas i en browser. Eftersom Mozilla Firefox används som browser i detta utförande, är den första synliga trafiken i Wireshark därför i form av DNS(Domain Name System)-querys och DNS-responses till och från servrar med namn som tiles.services.mozilla.com, location.services.mozilla.com, etc.

Sedan ökar trafiken ordentligt och Wiresharkfönstret fylls med UDP-paket. Detta är inte alls märkligt eftersom Freenet-protokollet förordar att all data som transmitteras inom nätverket ska vara inkapslad av segment från antingen UDP eller TCP på transportlagret. Vad som dock är intressant att notera är att den senaste versionen av Freenet verkar föredra UDP framför TCP. En uppenbar fördel med UDP är att transmissionen av meddelanden mellan noderna i nätverket är snabbare än med TCP, vilket lär vara anledningen till preferensen.

Segmenten skickas från och till väldigt många olika IP-adresser vilka inte verkar ha någon koppling till varandra eftersom samtliga har olika nätid. Genom googling på några av de IP-adresser som syns i käll- och destinationsadressfälten ges ett intressant resultat (se figur 4).

IP-adress	*Plats*
173.175.212.214	USA, Dallas
79.196.46.243	Tyskland, Saarbrucken
84.209.15.63	Norge, Oslo
107.170.137.131	USA, New York
106.186.114.41	Japan, Tokyo
185.52.1.21	Nederländerna, Enschede

Figur 4. Spårade Freenet-kopplingar

Dessa måste vara adresser till andra Freenetanvändare som datorn är uppkopplad till och har sparade i sin dynamiska routingtabell. Kommunikation med dessa adresser och ett antal andra bevaras under ett flertal transaktioner innan de successivt byts ut mot andra adresser. Detta tyder på att nätverket som väntat är ostrukturerat och att routingtabellen verkligen är dynamisk.

Trafiken verkar inte påverkas i någon större utsträckning oavsett läge (passivt, browsing, nedladdning). Det är också omöjligt att tolka innehållet eftersom det inte finns någon information utöver bitföljder.

IV. Slutsats

Freenets noder kommunicerar med varandra genom att skicka och vidarebefodra requests med hjälp av en dynamisk routingtabell och ett Freenet-protokoll som beskriver hur olika typer av meddelanden ska tolkas. Trafikuppfångandet visar att det som beskrivs i teorin även stämmer i praktiken. En viktig iakttagelse är att UDP används som transportprotokoll.

Referenser

[1] "About" på *Freenetproject.* Tillgänglig: https://freenetproject.org/about.html Hämtad: Dec. 07, 2016.

[2] "About Tor" på *Torproject* Tillgänglig: https://www.torproject.org/about/overview.html.en Hämtad: Dec. 07, 2016.

[3] "Help" på *Freenetproject.* Tillgänglig: https://freenetproject.org/help.html. Hämtad: Dec. 07, 2016.

[4] I. Clarke, O. Sandberg, B. Wiley och T.W Hong, "Freenet: A Distributed Anonymous Information Storage and Retrieval System," (2000). Tillgänglig: http://people.cs.uchicago.edu/ nugent/papers/Clarke00.pdf. Hämtad: Dec. 07, 2016.

[5] "Peer-to-peer (P2P)" på *Searchnetworking.* Tillgänglig: http://searchnetworking.techtarget.com/definition/peer-to-peer. Hämtad Dec. 07, 2016.

[6] C.Muoh, "A Tutorial on Gnutella, Bittorrent and Freenet Protocols" (2006) Tillgänglig: http://medianet.kent.edu/surveys/IAD06S-P2PArchitectures-chibuike/P2P%20App.%20Survey%20Paper.htm Hämtad Dec. 07, 2016.

[7] "Freenet" på *Wikipedia.* Tillgänglig: https://en.wikipedia.org/wiki/Freenet Hämtad: Dec. 07, 2016.