peer-to-peer and agent-based computing
Freenet

UNIVERSITY OF ABERDEEN

Plan of lecture

- Freenet Architecture
  - Goals and Properties
- Searching a network
  - Searching/Routing algorithm
  - Adaptive behaviour
  - Differences from other algorithms
- Keys
  - KSK keys
  - SSK keys
  - CHK keys
- Network Evolution and Clustering
  - Clustering keys

Freenet

- A decentralised system for storing/retrieving files within a massively distributed network
  - Each participant provides some network storage space
  - Peers are servents, providing/requesting storage
  - Different philosophy from Gnutella (no write permission in Gnutella)
- A storage and retrieval facility
  - Clients add a file to the network but do not know the actual storage location
  - Information is kept private by employing various levels of encryption as the data traverses through the network
- Adapts itself according to usage patterns

Freenet: Origins

Architect/inventor: Ian Clark

- Founded companies to commercialise Freenet technology
- Raised US$4M venture capital
- Top 100 innovators under the age of 35 by the MIT Technology Review magazine (Oct 2003)
- Holds a degree in Artificial Intelligence and Computer Science from Edinburgh University, Scotland

http://en.wikipedia.org/wiki/Ian_Clarke_(computer_scientist)

Why Freenet?

- Designed to provide extensive protection from hostile attack
  - Both from inside and outside by addressing information privacy and survivability issues
- Based around the P2P environment, which is inherently unreliable and untrustworthy
  - Assumes that all participants in the network could potentially be malicious or their peer could fail without warning
- Implements a self-organising routing mechanism over a decentralised structure
  - This algorithm dynamically creates a centralised/decentralised network

Why Freenet? (Cont'd)

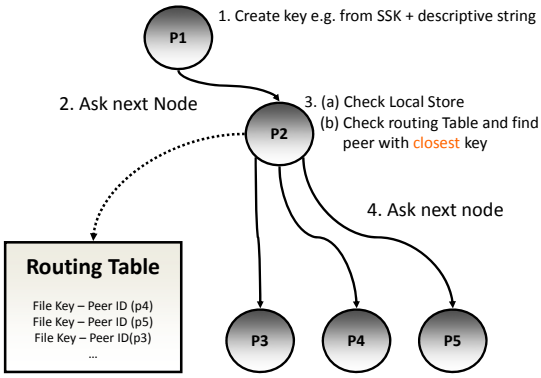
- The network learns
  - It route queries in a better fashion from local (not global!) knowledge
  - Achieves this by using file keys and sub-dividing the key space to partition the location of the stored files across the network

  Good example of how various technologies can be used in an innovative system.
- It supports:
  - P2P
  - Security (and privacy)
  - Scalability
  - Decentralised networks

Populating the Freenet Network

- File Keys
  - Used to route storage or retrieval requests onto the Freenet network
  - Constructed from the user or the file itself (more on this later)
- Routing Tables
  - Each peer has a routing table
  - Stores file keys and location of key (i.e. on connected peers) – see next slide

Routing Tables

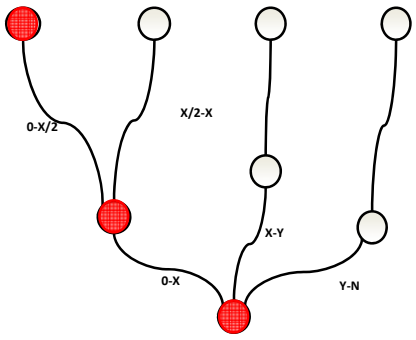


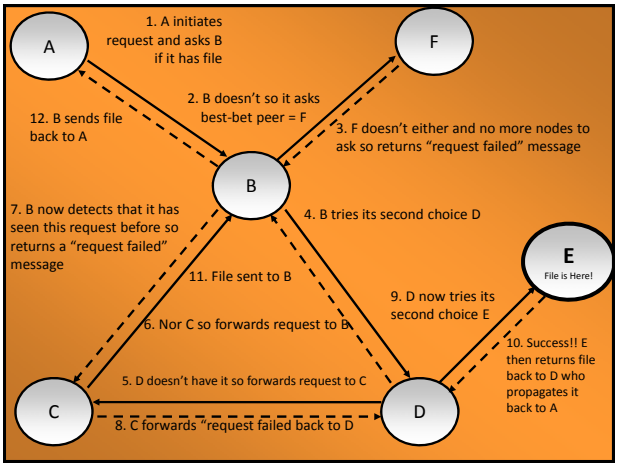
Searching & requesting

Searching:

- Peers try to intelligently route requests
- Peers ask neighbours (like Gnutella) BUT…
  - Peers do not forward request to all peers
- Peers find the closest key to the one supplied in their local routing table and pass the request only to this peer
  - Intelligent routing (subdividing keyspace)
- At each hop keys are compared and request is passed to the closest matching peer
- And so on…

Example key mapping





Updating routing tables

- If a peer forwards the request to a peer that can retrieve the data
  - Then address of upstream peer (which contains or is closer to the data) is included in reply
- This peer uses this information to update its local routing table to include the peer that has a more direct route to the data
- Then, when a similar request is issued again the peer can more effectively send the request to a node that is closer to the data

Adaptive behaviour

- Dynamic algorithm to update information is analogous to the way humans reinforce decisions based on past experiences
- The Milgram experiment:
  - Milgram noted that 25% of all requests went through the same person (the local shopkeeper)
  - People in this experiment used their experience of the local inhabitants to attempt to forward the letter to the best person who could help it reach its destination.

Adaptive behaviour (Cont'd)

- Local shopkeeper was a good choice:
  - He knew a number of out-of-town people
  - Therefore could help the letter get closer to its destination
- If experiment repeated using the same people
  - Surely the word would spread quickly in Omaha: shopkeeper is a good place to forward letters to
  - Success rate and efficiency would improve
  - People in Omaha would learn to route better!
- This is what Freenet does:
  - Adapts routing tables based on prior experiences

Similarities with other techniques

- Gnutella:
  - User searches the network by broadcasting her request to every node within a given TTL
  - Not efficient
- Napster:
  - Uses central database that contains locations of all files in network
  - Not scalable
  - Subject to attack due to centralisation of its file indexing
- However, both matured into using multiple caching servers in order to be able to scale the network
  - Resulting in a centralised/decentralised topology

But the Freenet approach…

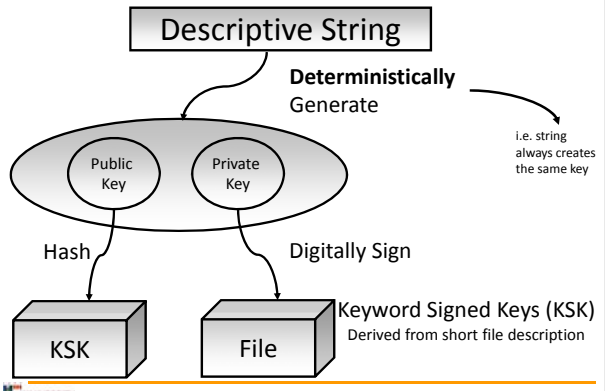
- Caching services (super peers or Napster indexes) are building blocks of Freenet
  - Each peer contains a routing table
- Key difference
  - Freenet peers do not store locations of files
  - Instead, they contain file keys that indicate the direction in key space where file is likely to be
  - File keys used to route the query to the stored file
  - But there are many different types of keys…

Keys

Three types of keys:

1. Keyword-Signed Keys (KSK)
   - The simplest of Freenet keys
   - Derived directly from a descriptive string that the user chooses for the file
2. Signed-Subspace Keys (SSK)
   - Used to create a subspace
   - To define ownership OR
   - To make pointers to a file or a collection of files
3. Content-Hash Keys (CHK)
   - Used for low-level data storage
   - Obtained by hashing the contents of the data to be stored

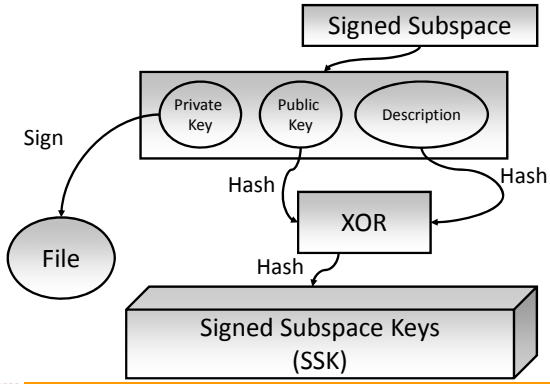
Keyword-Signed Keys (KSKs)



KSKs (Cont'd)

- Key Generation:
  - Derived from a descriptive string in a deterministic manner
  - Therefore same key pair gets created for the same key
  - If we change the string then a new key gets generated and therefore a new file gets created
  - If we create the same key, then the old file gets overwritten
- Ownership:
  - None – file owned only by descriptive string

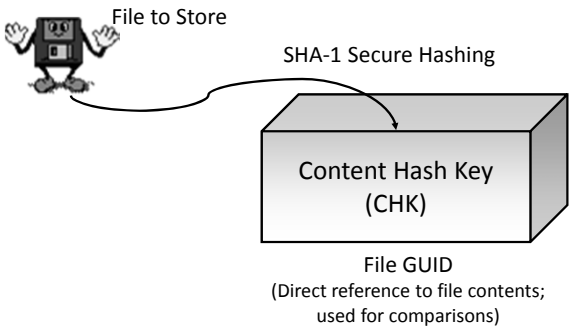
Signed Subspace Keys (SSKs)



SSKs (Cont'd)

- Key Generation:
  - Derived from subspace key pair + description
  - Unique within this sub-domain (i.e. the key subspace)
- Ownership:
  - Creates a read-only file system for all users
  - Only owners of the subspace can overwrite the files within the subspace
  - i.e. they need private subspace key to generate the correct signature

Content Hash Keys (CHKs)



CHKs (Cont'd)

- Key Generation:
  - derived directly from the contents of the file
- Ownership:
  - None – normally associated with a subspace to define ownership

Analogies for keys

Three types of keys:

- Keyword-Signed Keys (KSK):
  - Like filenames on a file system
  - Analogous to having all files in one directory
- Signed-Subspace Keys (SSK):
  - May contain collections of filenames
  - Analogous to using (multiple level) directories
- Content-Hash Keys (CHK):
  - Like inodes on a file system, i.e. a pointer to the file on disk

The use of keys

- Keyword-Signed Keys (KSK) and Signed-Subspace Keys (SSK):
  - Used to create a user view of the file
  - E.g. a description or a subspace
- Content-Hash Keys (CHK):
  - Used to verify file – for file version control, integrity, etc

Distribution of keys within key space

- All keys use hash functions to create final key value
- Hash functions have a good avalanche effect
  - Therefore input has no correlation with output
- So, two very similar files will create two completely different hash keys (CHKs)
  - Therefore, similar files will be put in completely different parts of the network

Properties of key distribution

Does this random behaviour matter?

- Helps file distribution across network
  - Imagine experiment in which all data is quite similar (e.g. people faces, star features, etc.)
  - Freenet keys will create quasi-random keys from these files
  - This ensures even (random) distribution across all peers within the network

Summary

- Example technology
- Demonstrates how some of the technologies can be used in a system
  - E.g., security and privacy policies/techniques
- Shows how centralised-decentralised models can be dynamically created
  - In a self-organising fashion

Further reading

- Freenet web site (including downloads):
  http://freenetproject.org
  - In particular, the "philosophy" page
    https://freenetproject.org/about.html#philosophy
- Freenet Wikipedia page:
  http://en.wikipedia.org/wiki/Freenet
- MiniHowTo
  http://www.minihowto.org/freenet_minihowto/freenet%20a%20very%20short%20minihowto.html
- Textbook chapter on Freenet available on MyAberdeen