Further techniques for Anonymity

Tor: anonymity online

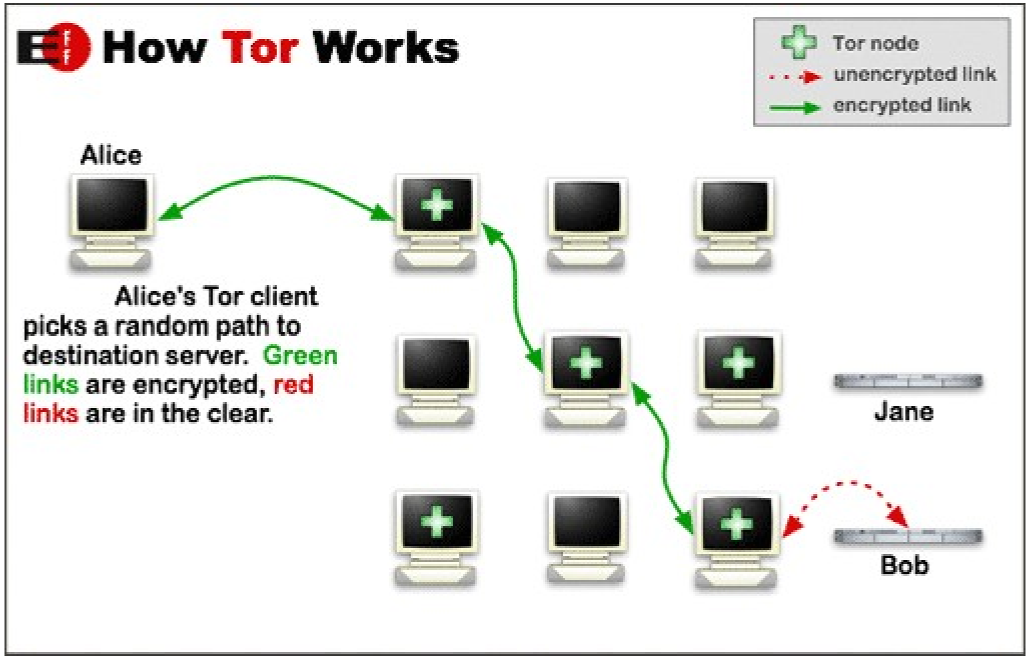
- Tor (from The onion routing project, originated in the US Naval Research Lab) – practical solution for anonymity protection : www.torproject.org
- It is free and open source and available for major platforms: Windows, Mac, Linux/Unix, Android
- Can be used for web browsing and instant messaging, prevents people from learning your location or browsing habits

Tor-Networks

- Traffic routed via encrypted nodes. Each node decrypts one layer of encryption

- Entry point is only known by the first link, and exit point is only known by the last link

- Nodes are only aware of the next link.



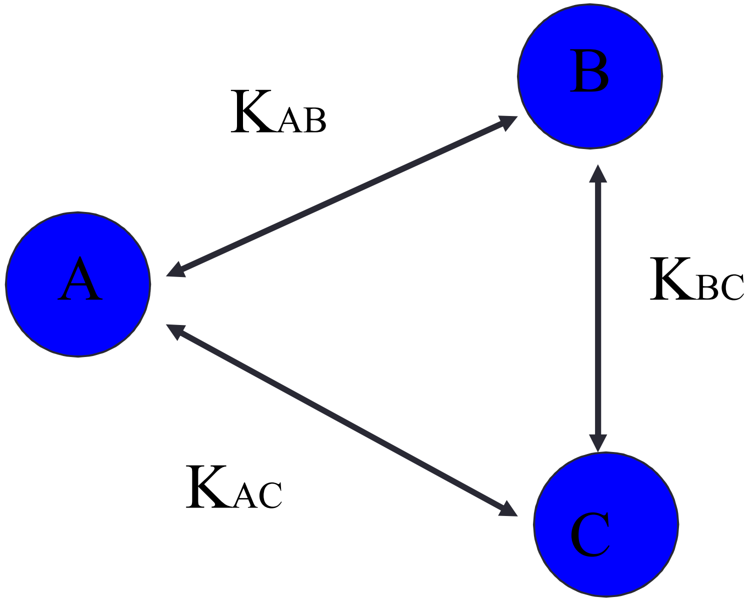
Alice's Tor client picks a random path to destination server. Green links are encrypted, red links are in the clear.

Tor: main principles

- A combination of (variants of) mix-network and crowd mechanisms
  - Using a set of relay nodes (~crowd); currently >6000
  - Routing using random choice (similar to crowds)
  - Encrypted connections between neighbouring nodes (similar to mix-networks)
  - It uses public key cryptography to share the secret keys between neighbouring nodes => uses the shared secret to perform symmetric encryption in further communications between neighbours
  - temporarily available virtual channels

DC-networks

- D.Chaum, 1988: **D**ining **C**ryptographer networks

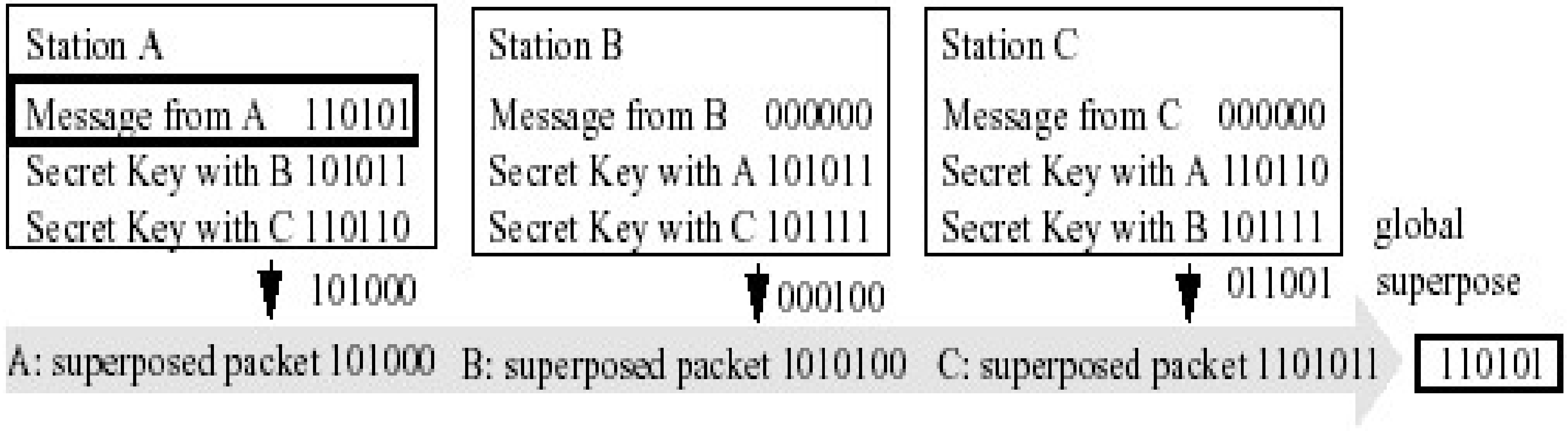


• At the preliminary stage between some pairs of nodes (at the picture between all) secret keys (sequences of bits) are exchanged

DC-networks

- To send a message M (sequence of bits), a node, say A, *broadcasts* the value **$(M +_2 K_{AB} +_2 K_{AC})$**, i.e. superposition of the message and all keys of A, here **$+_2$** stands for bitwise addition modulo 2 (or XOR operation)
- All other nodes broadcast superpositions of all their keys. So, B broadcasts **$(K_{AB}+K_{BC})$** and C broadcasts **$(K_{AC}+K_{BC})$**
- All nodes then superpose all received messages and get **$(M+_2K_{AB}+_2K_{AC}+_2K_{AB}+_2K_{BC}+_2K_{AC}+_2K_{BC}) = M$**
- (the initial message !!!)

DC-network



Station A

Message from A 110101

Secret Key with B 101011

Secret Key with C 110110

101000

Station B

Message from B 000000

Secret Key with A 101011

Secret Key with C 101111

000100

Station C

Message from C 000000

Secret Key with A 110110

Secret Key with B 101111

011001

global superpose

A: superposed packet 101000 B: superposed packet 1010100 C: superposed packet 1101011

110101

A message sent by A in the DC-network.

Picture by A.Pfitzmann

Anonymity by DC-networks

- DC-networks provide for *sender* anonymity because an adversary is unable to decide whether the packets he may observe contain a message or not;
- DC-networks can be used in combination with other mechanisms, such as mix-networks to enhance anonymity
- A major drawback is that DC-Networks require the preliminary stage exchanging the secret keys between participants
- Every round of communication requires a new set of keys
- Every node needs to participate every time a message is broadcasted => high load on the nodes => impractical in large networks

Recent developments in DC-networks

- Dissent system (~2012):
  - Scalable to thousands nodes
  - Client-server architecture with several servers and small groups served by a server
  - Retro-active blame mechanism to deal with *jamming*
  - XOR together with more complicated *group multiplication* operations are used
  - See further details at *dedis.cs.yale.edu/dissent*