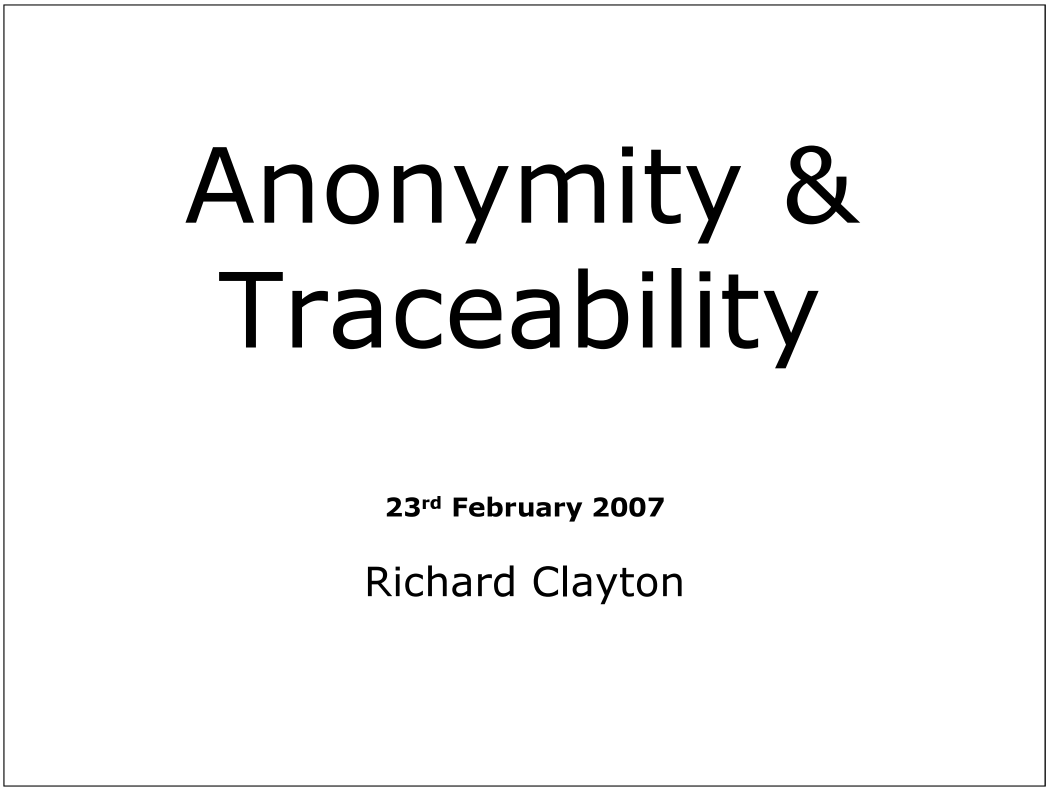


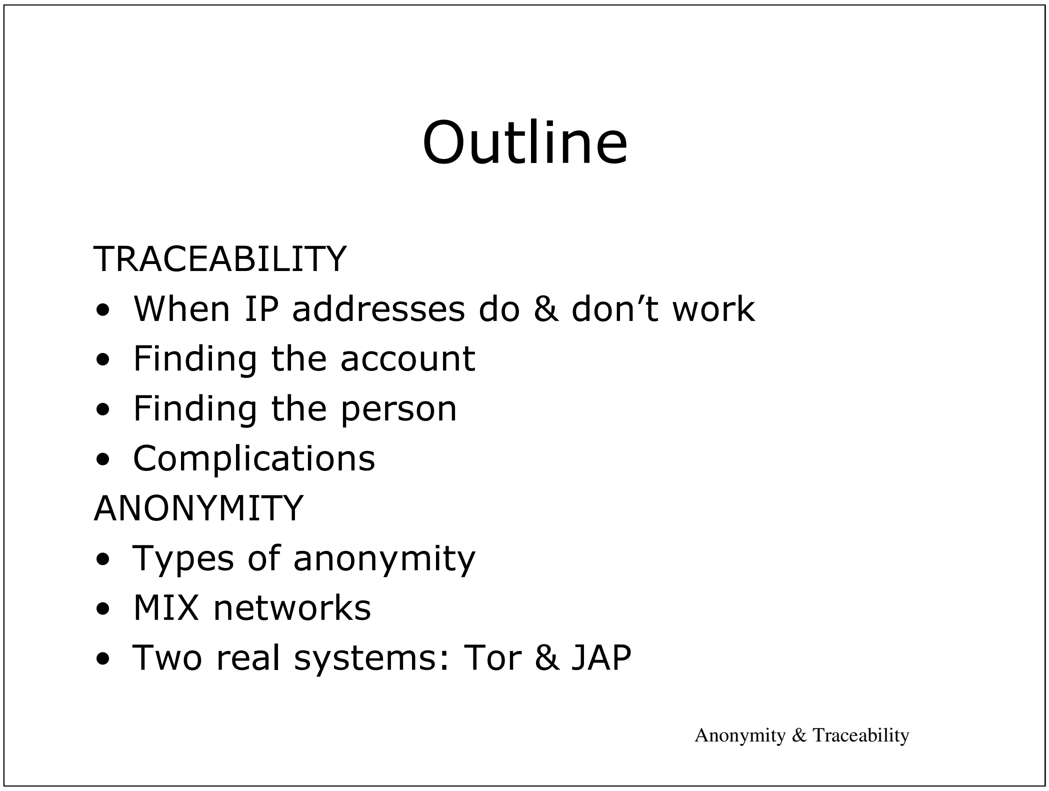
Anonymity & Traceability

23rd February 2007

Richard Clayton

These lecture notes were specially prepared for a lecture in the University of Birmingham's "Communication Skills and Professional Issues" course.

richard.clayton@cl.cam.ac.uk



Outline

TRACEABILITY

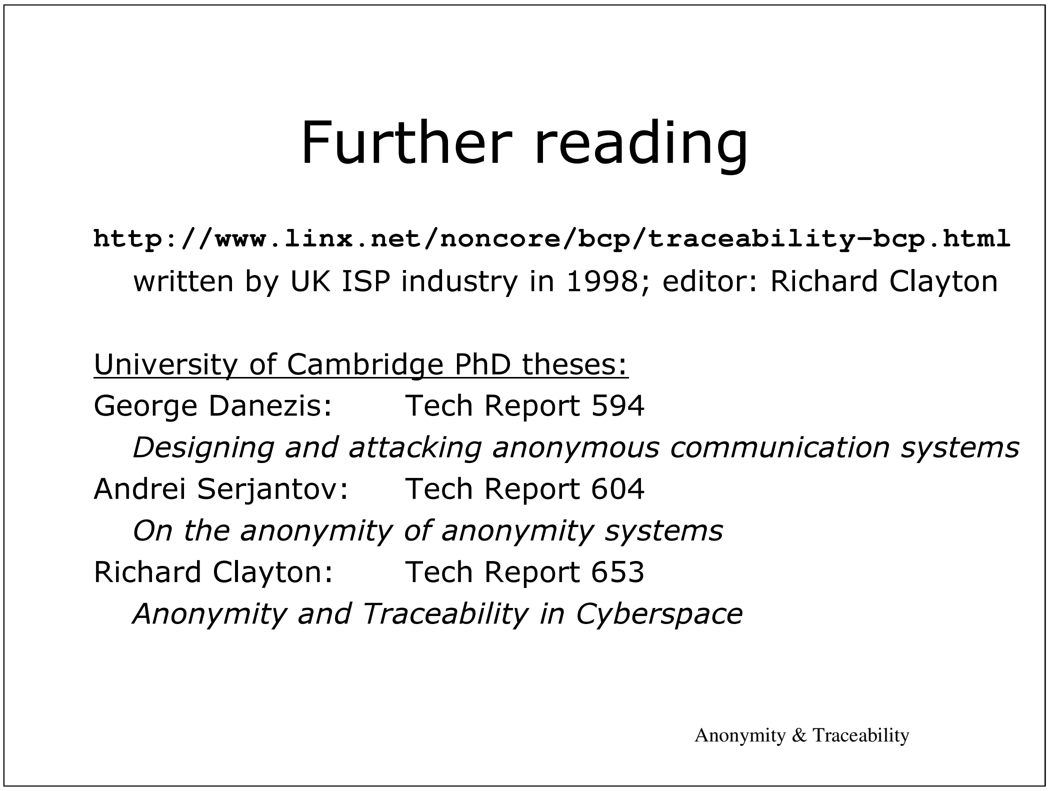
- When IP addresses do & don’t work
- Finding the account
- Finding the person
- Complications

ANONYMITY

- Types of anonymity
- MIX networks
- Two real systems: Tor & JAP

Anonymity & Traceability

The slides give the broad outline of the lecture and the notes ensure that the details are properly recorded, lest they be skipped over on the day. However, it is at least arguable that it will be far more interesting to take notice of what I say off-the-cuff rather than relying on this document as an accurate rendition of what the lecture was really about!



Further reading

`http://www.linx.net/noncore/bcp/traceability-bcp.html`
written by UK ISP industry in 1998; editor: Richard Clayton

University of Cambridge PhD theses:

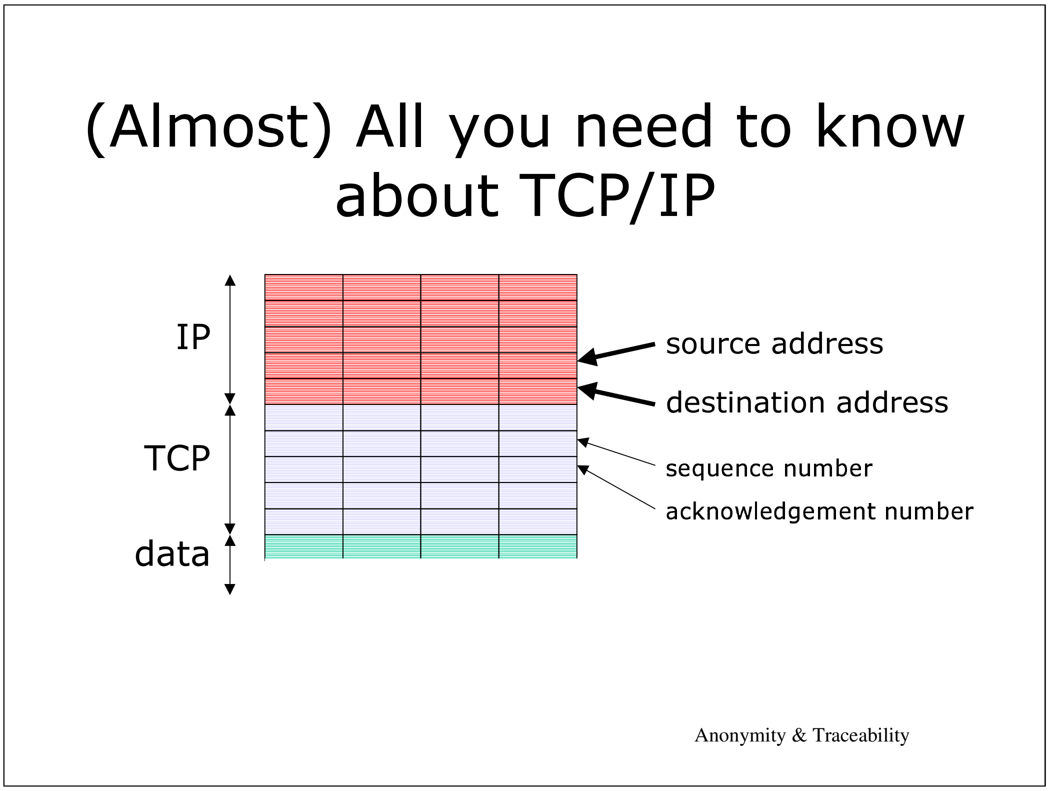
George Danezis: Tech Report 594
Designing and attacking anonymous communication systems

Andrei Serjantov: Tech Report 604
On the anonymity of anonymity systems

Richard Clayton: Tech Report 653
Anonymity and Traceability in Cyberspace

There's not been a great deal of material collected together on the topics covered in this lecture. However, these notes give some further references, as appropriate, on particular issues; and the theses have extensive bibliographies.

Don't be daunted by these being PhD theses – they're all pretty readable, at least to start with!



(Almost) All you need to know about TCP/IP

IP

TCP

data

source address

destination address

sequence number

acknowledgement number

Anonymity & Traceability

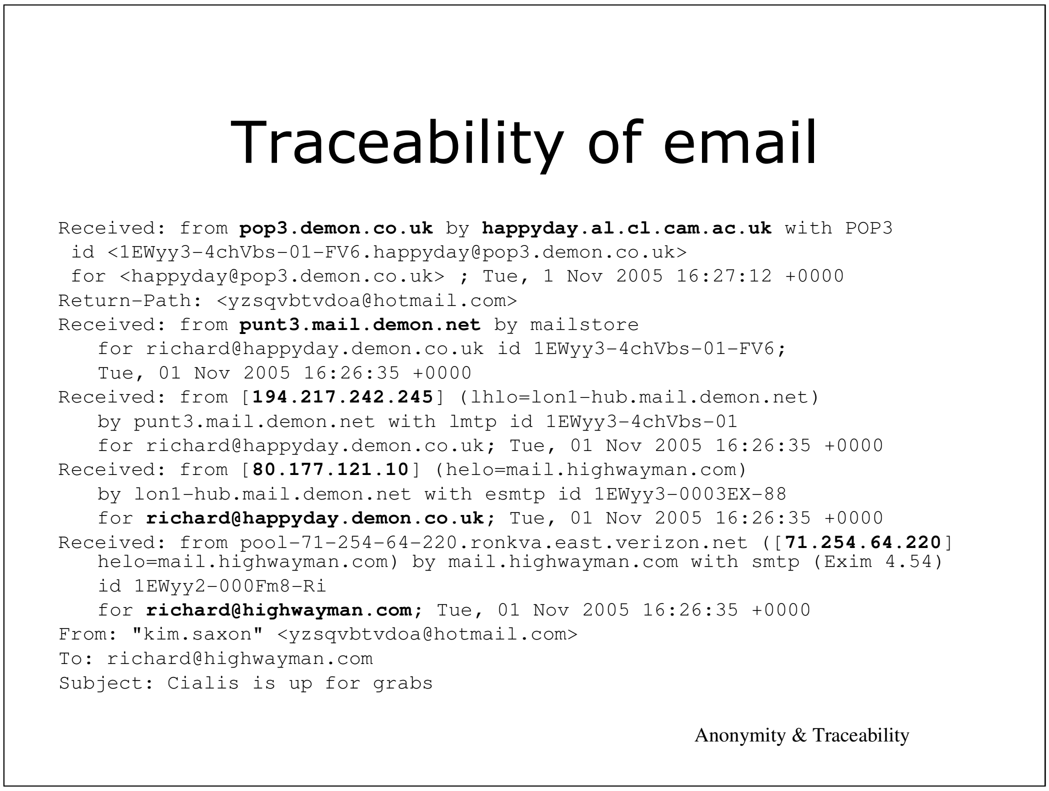
✶ TCP/IP is described in many textbooks. There are only a few important aspects of the protocol from the point of view of Traceability.

✶ The *destination IP address* says where the packet is to be sent. It is always, by definition, valid.

✶ The *source IP address* indicates where the packet came from. It can be forged (but may not then be allowed out of its originating network if the "firewalls" there (usually in fact just simple routers) are configured in accordance with RFC2267).

✶ When the packet reaches its destination, the source and destination addresses will be swapped over for the return journey.

✶ The *sequence number* indicates where the contents of the current packet fit in the notional buffer for the whole conversation. The *acknowledgement number* indicates how much of that buffer has been received so far. Both of these values start from a randomly chosen point in a 2^{32} byte buffer.



Traceability of email

```
Received: from pop3.demon.co.uk by happyday.al.cl.cam.ac.uk with POP3
 id <1EWyy3-4chVbs-01-FV6.happyday@pop3.demon.co.uk>
 for <happyday@pop3.demon.co.uk> ; Tue, 1 Nov 2005 16:27:12 +0000
Return-Path: <yzsqvbtvdoa@hotmail.com>
Received: from punt3.mail.demon.net by mailstore
   for richard@happyday.demon.co.uk id 1EWyy3-4chVbs-01-FV6;
   Tue, 01 Nov 2005 16:26:35 +0000
Received: from [194.217.242.245] (lhlo=lon1-hub.mail.demon.net)
   by punt3.mail.demon.net with lmtp id 1EWyy3-4chVbs-01
   for richard@happyday.demon.co.uk; Tue, 01 Nov 2005 16:26:35 +0000
Received: from [80.177.121.10] (helo=mail.highwayman.com)
   by lon1-hub.mail.demon.net with esmtp id 1EWyy3-0003EX-88
   for richard@happyday.demon.co.uk; Tue, 01 Nov 2005 16:26:35 +0000
Received: from pool-71-254-64-220.ronkva.east.verizon.net ([71.254.64.220]
   helo=mail.highwayman.com) by mail.highwayman.com with smtp (Exim 4.54)
   id 1EWyy2-000Fm8-Ri
   for richard@highwayman.com; Tue, 01 Nov 2005 16:26:35 +0000
From: "kim.saxon" <yzsqvbtvdoa@hotmail.com>
To: richard@highwayman.com
Subject: Cialis is up for grabs
```

Anonymity & Traceability

✶ As email passes through a mail system a “Received:” header line will be added to the top of the existing message. Inspecting the header lines will therefore provide a trace of where the email has come from.

The formal format of the Received: header lines is documented in RFC2821 & 2822, though in practice a fair amount of variability will be encountered. In principle you will be told the name of the machine generating the Received: header line, when it was added, where the email came from and who it was addressed to at that stage.

✶ See a FAQ, eg: http://www.stopspam.org/email/headers.html for more about reading email headers… from which you will deduce that this particular email definitely came from a Demon Internet ADSL connection and if, like me, you trust the system there, you can assert that it came from a verizon.net connection, quite possibly in Roanoke, Virginia, USA

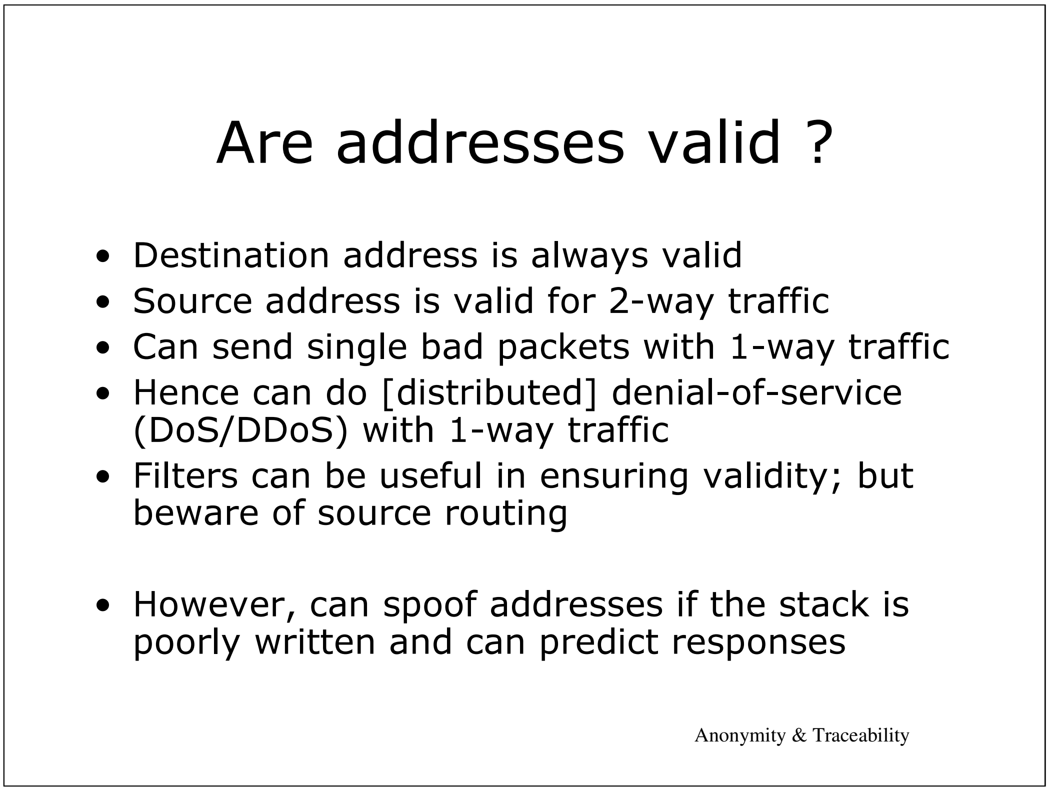
✶ It is not uncommon to see three different identities presented for the machine from which the email came:

the name claimed (in the SMTP “HELO” line)

the remote IP address

the reverse DNS lookup for the remote IP address

The two names may differ for legitimate reasons, but here the spammer is relaying via an insecure machine and saying HELO with the remote machine’s MX identity (which is just plain wrong).



Are addresses valid ?

- Destination address is always valid
- Source address is valid for 2-way traffic
- Can send single bad packets with 1-way traffic
- Hence can do [distributed] denial-of-service (DoS/DDoS) with 1-way traffic
- Filters can be useful in ensuring validity; but beware of source routing

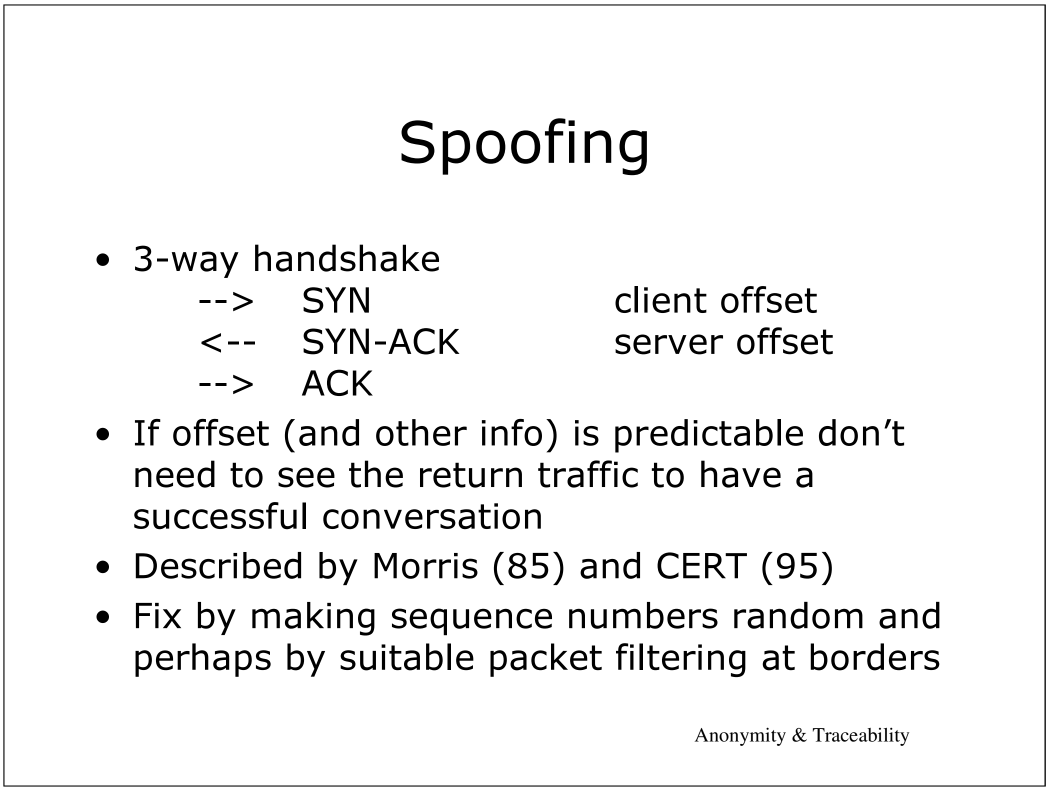
- However, can spoof addresses if the stack is poorly written and can predict responses

Anonymity & Traceability

✶ If you are not interested in getting packets back from a remote machine then the validity of the source address is irrelevant. If you wish to avoid being traced then you might set an invalid address. There are a number of attacks that are possible with 1-way traffic such as denial of service attacks and the sending of malformed packets that crash the remote system.

eg Teardrop (invalid fragments)
Land (connection to self)
“Ping of Death” (extra long packets)
WinNuke (buffer overflow on 139)
SYN (only one handshake packet, so consumes resources)
Reflector (repeated SYN-ACK responding to forged SYN)
Looping UDP (connects echo(7) to chargen(19))
etc etc

✶ It is possible to filter packets to ensure they are valid (spotting insider addresses coming from outside and vice versa). However, IP does have a concept of “source routing” which causes packets to go via particular intermediate addresses first. In practice, however, source routed packets may well get dropped because they’re prima facie evidence of wickedness !



Spoofing

- 3-way handshake

-->	SYN	client offset
<--	SYN-ACK	server offset
-->	ACK	

- If offset (and other info) is predictable don’t need to see the return traffic to have a successful conversation
- Described by Morris (85) and CERT (95)
- Fix by making sequence numbers random and perhaps by suitable packet filtering at borders

Anonymity & Traceability

✶ Spoofing connections was first described in:

“A Weakness in the 4.2BSD UNIX TCP/IP Software”, Robert T. Morris, Computing Science Technical Report No. 117, AT&T Bell Laboratories, Murray Hill, New Jersey. 1985

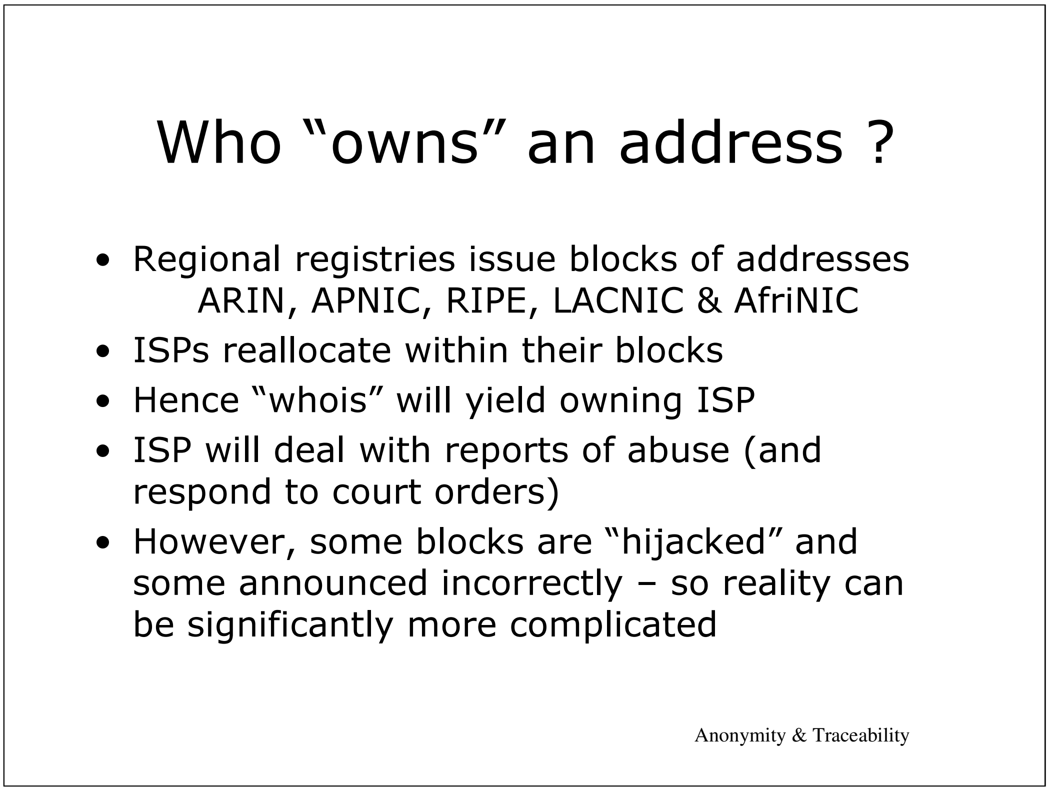
online at: http://www.pdos.lcs.mit.edu/~rtm/papers/117.pdf

The paper is particularly concerned with systems that trust other local machines (through hosts.allow mechanisms permitting rlogin &c). If you can successfully pretend to be local then you will have unauthorised access.

✶ To run the attack successfully you have to predict the sequence numbers (either by knowing them a priori, or by knowing an offset from another (non-spoofed) connection made first). Since the spoofed host will notice the unexpected SYN-ACK traffic it may also be necessary to run a “denial of service” on it to keep it from issuing a RST for the “connection”.

✶ Morris suggests filtering out packets coming in from the outside that have internal source addresses (this is related to the RFC2267 filtering) and also ensuring that the sequence numbers are truly random.

✶ In 1995 there were enough systems being compromised for CERT to issue an advisory (CA-1995-01), and as late as October 2000 FreeBSD was being fixed to use something better than a simple PRNG to create “random” sequence numbers!



Who “owns” an address ?

- Regional registries issue blocks of addresses
  ARIN, APNIC, RIPE, LACNIC & AfriNIC
- ISPs reallocate within their blocks
- Hence “whois” will yield owning ISP
- ISP will deal with reports of abuse (and respond to court orders)
- However, some blocks are “hijacked” and some announced incorrectly – so reality can be significantly more complicated

Anonymity & Traceability

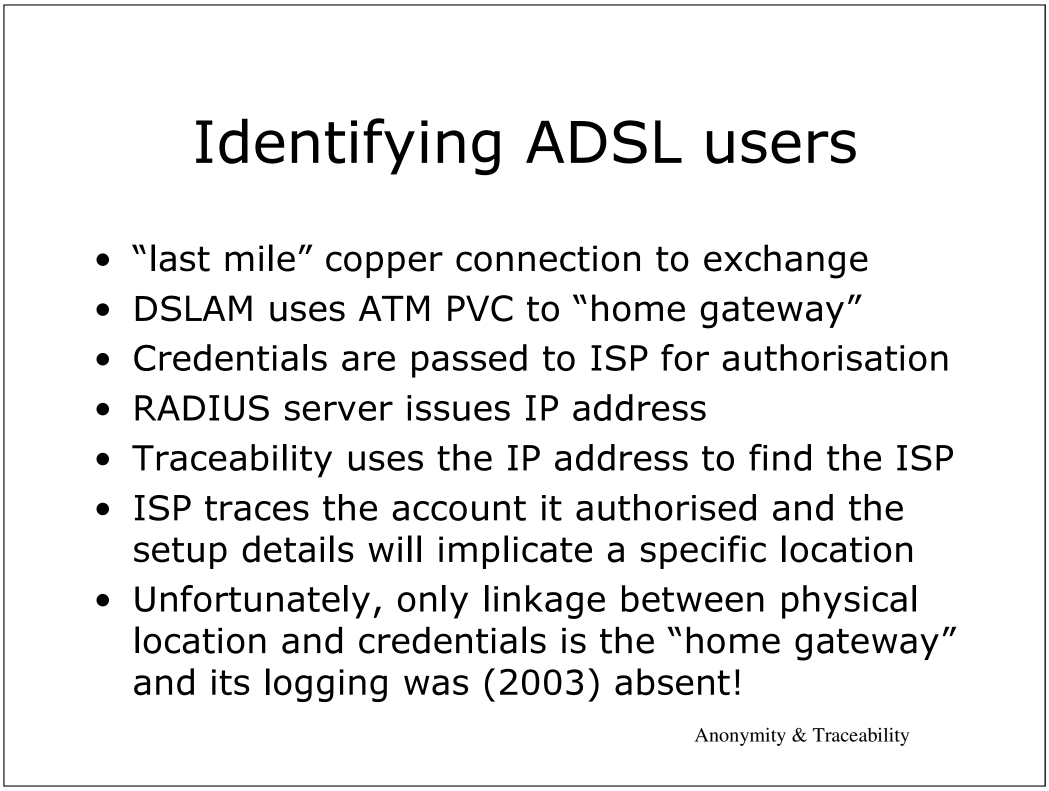
✶ IANA “owns” the IP address space, but it is managed by five “regional” registries:

ARIN	North America http://www.arin.net/
APNIC	Asia-Pacific (ie Far East & the Antipodes) http://www.apnic.net/
RIPE	Europe, Middle East http://www.ripe.net/
LACNIC	Latin America & The Carribbean http://www.lacnic.net
AfriNIC	Africa http://www.afrinic.net

✶ The registries provide IP address registration services. They maintain databases of IP address “ownership” and AS (Autonomous System) numbers (collections of routable blocks of IP space).

✶ Other systems and registries provide the “forward” mapping from domain names to IP addresses, but the regional registries maintain the framework for the “reverse” mapping from IP address to “machine name”. The actual “reverse DNS” entries are of course held in a distributed database in the normal manner.

✶ For examples of hijacking of address space see:
http://www.securityfocus.com/news/5654



Identifying ADSL users

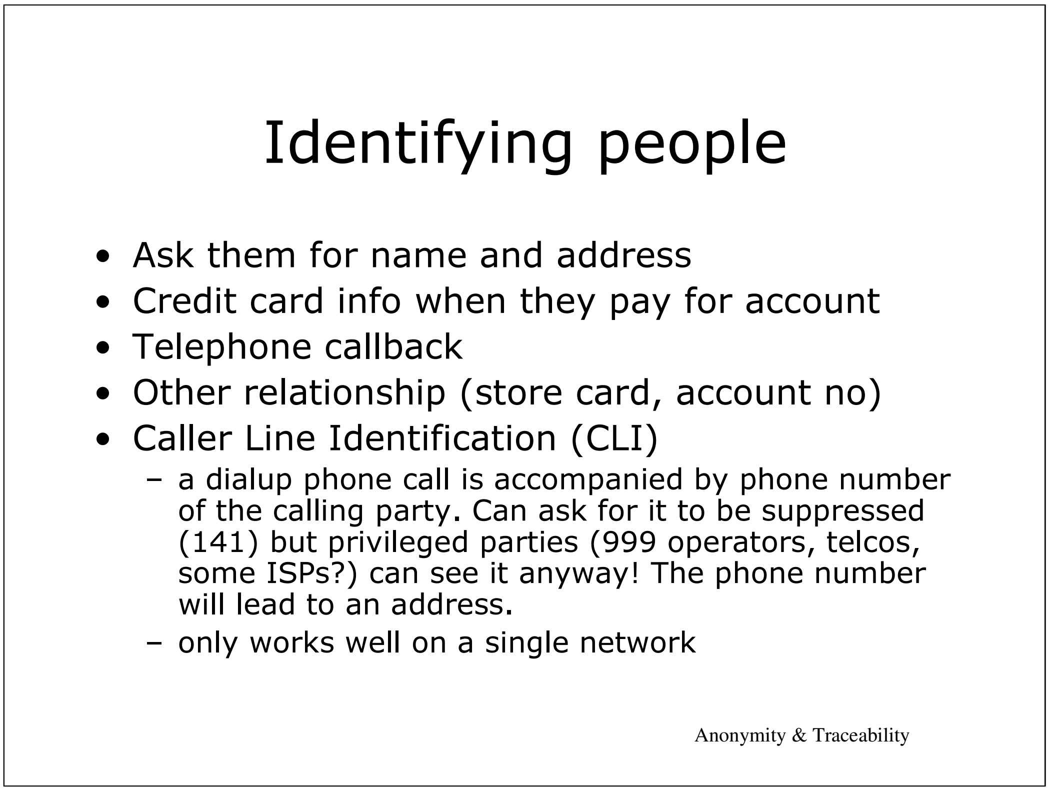
- “last mile” copper connection to exchange
- DSLAM uses ATM PVC to “home gateway”
- Credentials are passed to ISP for authorisation
- RADIUS server issues IP address
- Traceability uses the IP address to find the ISP
- ISP traces the account it authorised and the setup details will implicate a specific location
- Unfortunately, only linkage between physical location and credentials is the “home gateway” and its logging was (2003) absent!

Anonymity & Traceability

✶ ADSL connections (in the UK, in the vast majority case of BT provision, for IPStream systems) work as follows:
The signals pass along the copper wires to a DSLAM (Digital Subscriber Line Access Multiplexer) in the local exchange. This equipment constructs a PVC (Permanent Virtual Circuit) across the ATM cloud to the “home gateway”. This machine will take the user provided access credentials (login name and password) and pass these to the appropriate ISP for authorisation. On receipt of authorisation the home gateway will arrange for further traffic to be transferred along a 155Mb or 622Mb “fat pipe” to the ISP which will send it out onto the global Internet.

✶ Tracing works backwards from IP address to ISP, via RADIUS (Remote Authentication Dial In User Service – RFC2865) logs to the account (assuming the time is accurately known!) and from the account to the setup details for the ADSL link which implies a specific physical location – where the copper terminates!

✶ If you examine the descriptions of the forward and reverse activity carefully you can see that if credentials are used on a different ADSL line then the traceability fails. Hence you should examine the logs on the “home gateway” to establish the PVC information and from that the identity of the DSLAM and the precise bit of copper used. When the police first did this (because traceability had led them to the wrong place) they found that these logs did not exist. BT refuse to comment, on principle, whether (or when) this lacuna was corrected.



Identifying people

- Ask them for name and address
- Credit card info when they pay for account
- Telephone callback
- Other relationship (store card, account no)
- Caller Line Identification (CLI)
  - a dialup phone call is accompanied by phone number of the calling party. Can ask for it to be suppressed (141) but privileged parties (999 operators, telcos, some ISPs?) can see it anyway! The phone number will lead to an address.
  - only works well on a single network

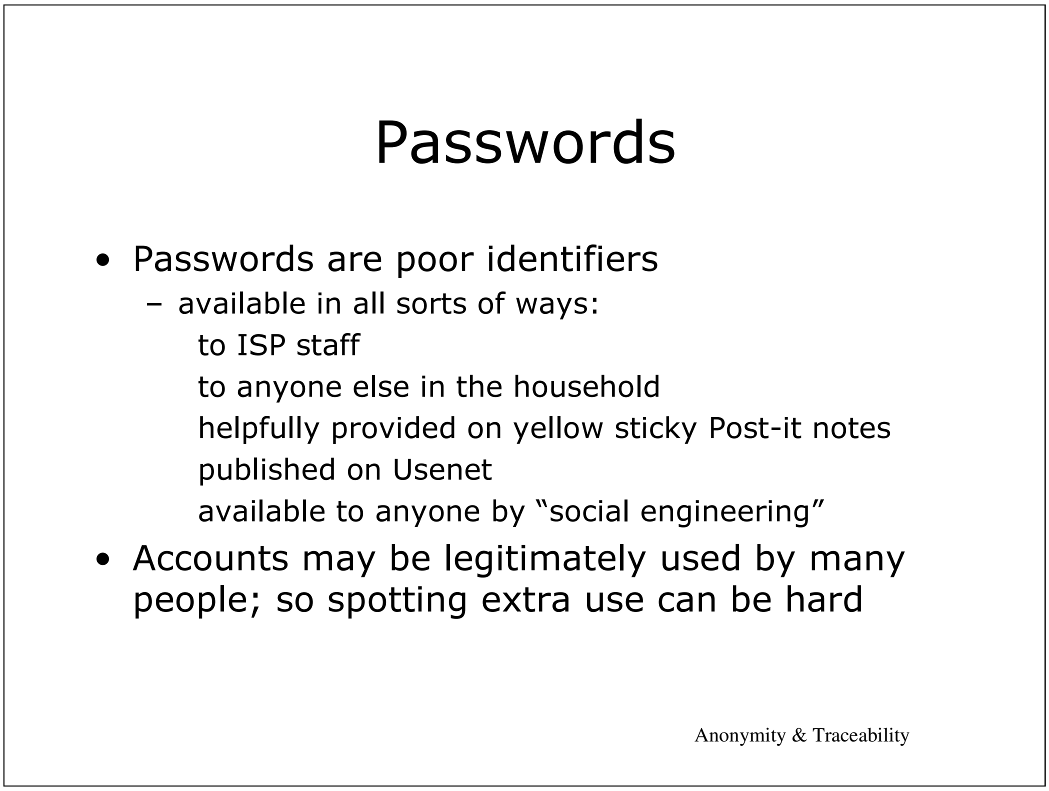
Anonymity & Traceability

✶ Having established which account used the IP address that “did something” then it is usually desirable to determine who was operating the account. This is not always the case – sometimes just knowing the account is sufficient; if it is an abuse incident (unsolicited bulk email perhaps) then the account will be suspended. The identity of the user is not relevant in such a case. However, a police officer seeking the poster of paedophile material will be interested in establishing who the user was.

✶ Most ISPs will wish to know your name and address before letting you open an account. They will probably check its internal consistency (does this postcode apply to this town?) to try and screen out grossly inaccurate responses. Online postcode databases make this check easy to evade.

✶ If you are paying for the account then it is likely that you’ll be using a credit or debit card. This provides, through the banking system, traceability to a particular person.

✶ Free ISPs also like to identify their customers, both for marketing purposes or to prevent abuse. They may collect information like your Tesco card number in order to identify you. It would be unusual for a free ISP to allow dial-up connection without Caller Line Identification (CLI).



Passwords

- Passwords are poor identifiers
  - available in all sorts of ways:

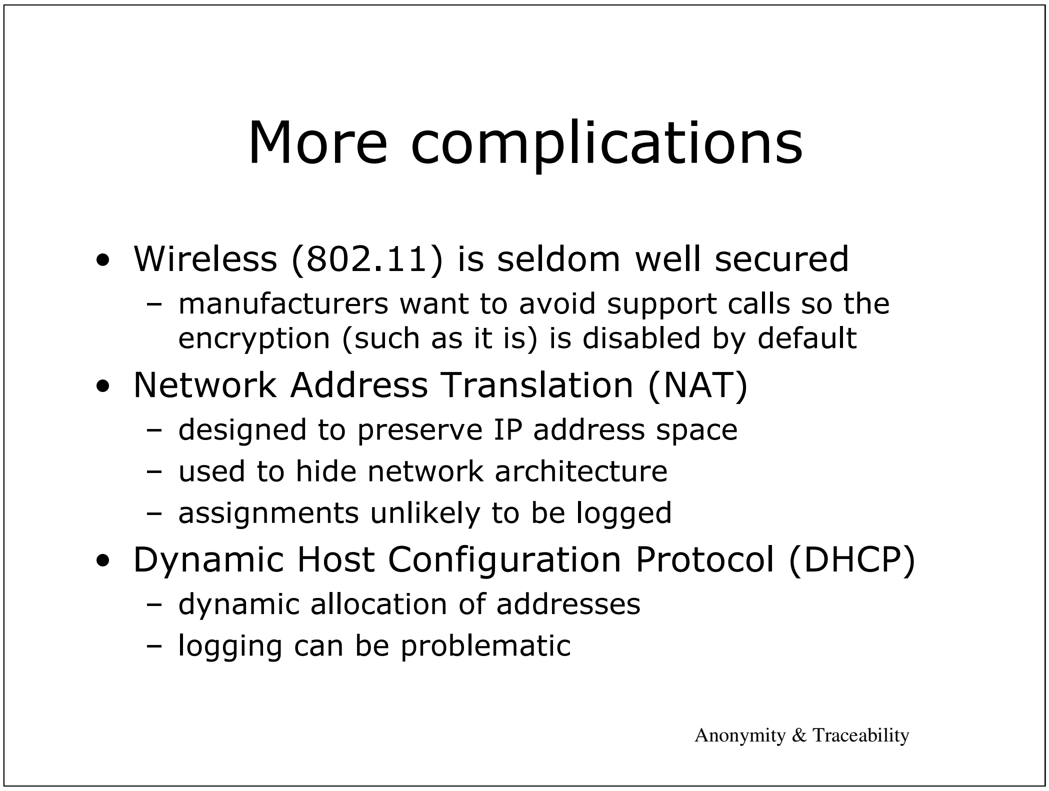
    to ISP staff
    to anyone else in the household
    helpfully provided on yellow sticky Post-it notes
    published on Usenet
    available to anyone by “social engineering”
- Accounts may be legitimately used by many people; so spotting extra use can be hard

Anonymity & Traceability

✶ Tracing an event via its IP address to an ISP account is not the same as locating the person who “did it”. The account may have been in use by someone other than its owner. Account ownership is usually demonstrated by providing a password – and that password can be compromised in many ways.

✶ The ISP staff may be aware of customer password settings. Others in the same household or office may know the password. The password may have been inadvertently posted to Usenet (along with some other debugging information relating to a dial-up problem) or indeed the password may have been disclosed to someone plausible who just asked for it (a process usually known as “social engineering”).

✶ Alternatively, the account may have multiple legitimate users and there may be insufficient records to demonstrate which of the users was responsible for a particular event. This may not be a problem to the ISP, who will close an account no matter which individual perpetrated some abuse, but it will be a problem to a police officer who needs to arrest the correct person.



More complications

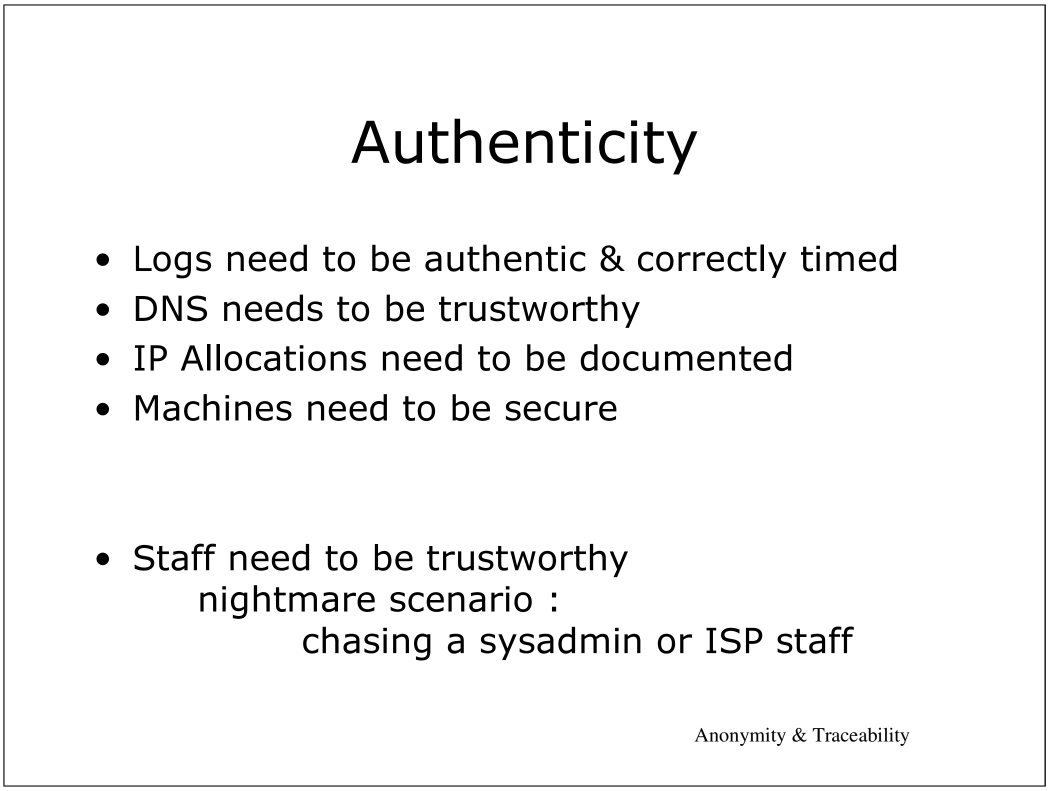
- Wireless (802.11) is seldom well secured
  - manufacturers want to avoid support calls so the encryption (such as it is) is disabled by default
- Network Address Translation (NAT)
  - designed to preserve IP address space
  - used to hide network architecture
  - assignments unlikely to be logged
- Dynamic Host Configuration Protocol (DHCP)
  - dynamic allocation of addresses
  - logging can be problematic

Anonymity & Traceability

✶ Wireless systems are seldom secure (even when their users have made an attempt to ensure that they are). The traceability will lead to the operator of the access point and seldom any further at all.

✶ Network Address Translation (NAT) is widely used to conserve address space, to allow the operation of several machines on a single dial-up connection and for security reasons by ensuring that machines are not visible to the open Internet. The IP address recorded at a remote site is likely to be the address of the kit doing the NAT. Mapping this to a particular machine “behind the NAT” is unlikely to be possible since it is rare to record NAT assignments in logs.

✶ Even where machines are on the open Internet, their IP addresses may not be fixed, but may be dynamically allocated using a protocol such as DHCP (Dynamic Host Configuration Protocol). This means that an individual machine may change IP address from day to day. Keeping logs would be unusual. Keeping them for long periods would be more unusual still.



Authenticity

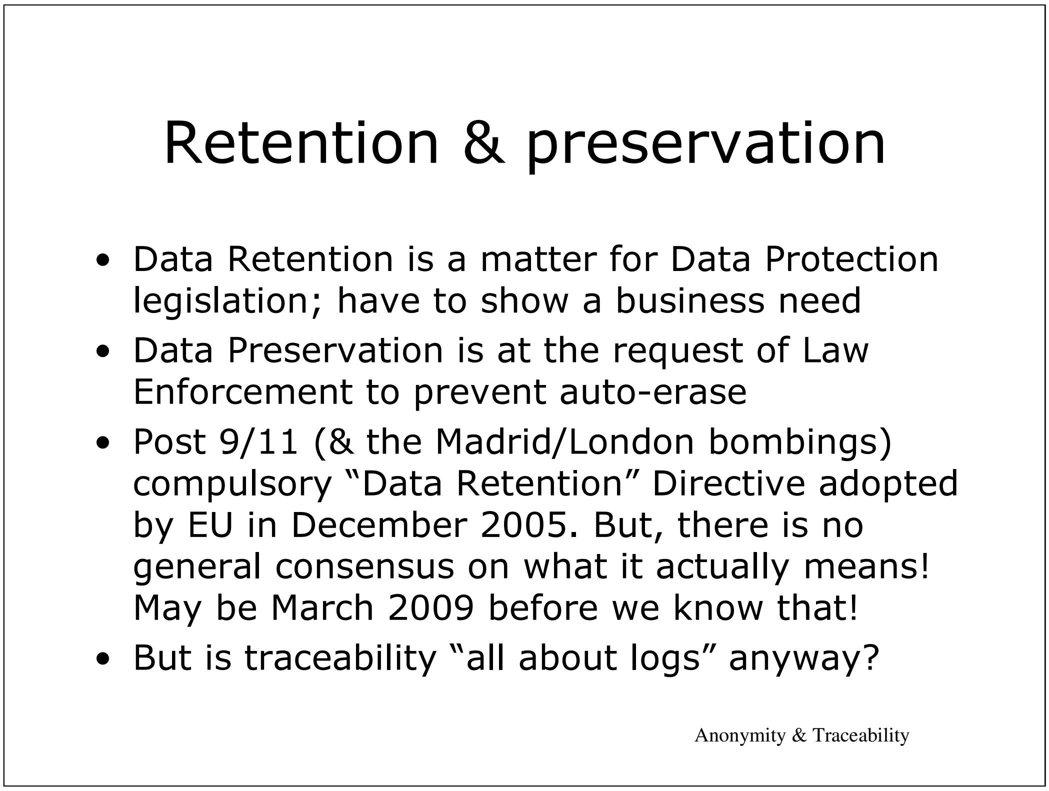
- Logs need to be authentic & correctly timed
- DNS needs to be trustworthy
- IP Allocations need to be documented
- Machines need to be secure

- Staff need to be trustworthy
  nightmare scenario :
  chasing a sysadmin or ISP staff

Anonymity & Traceability

✶ Traceability is the process of following a chain of data, from IP address to ISP, to customer account, to end user. If any part of this chain contains dud data, whether through accident or design, then it will not lead to the correct account, let alone the correct person. Authenticity is therefore essential.

✶ The risks of relying on DNS remaining the same between when a log is created and when it consulted have already been mentioned. Further problems arise in assessing the authenticity of logs if the local provision of DNS can be subverted, perhaps by “cache poisoning” attacks. It is usually considered best practice to record raw IP addresses alongside any DNS results.



Retention & preservation

- Data Retention is a matter for Data Protection legislation; have to show a business need
- Data Preservation is at the request of Law Enforcement to prevent auto-erase
- Post 9/11 (& the Madrid/London bombings) compulsory "Data Retention" Directive adopted by EU in December 2005. But, there is no general consensus on what it actually means! May be March 2009 before we know that!
- But is traceability "all about logs" anyway?

Anonymity & Traceability

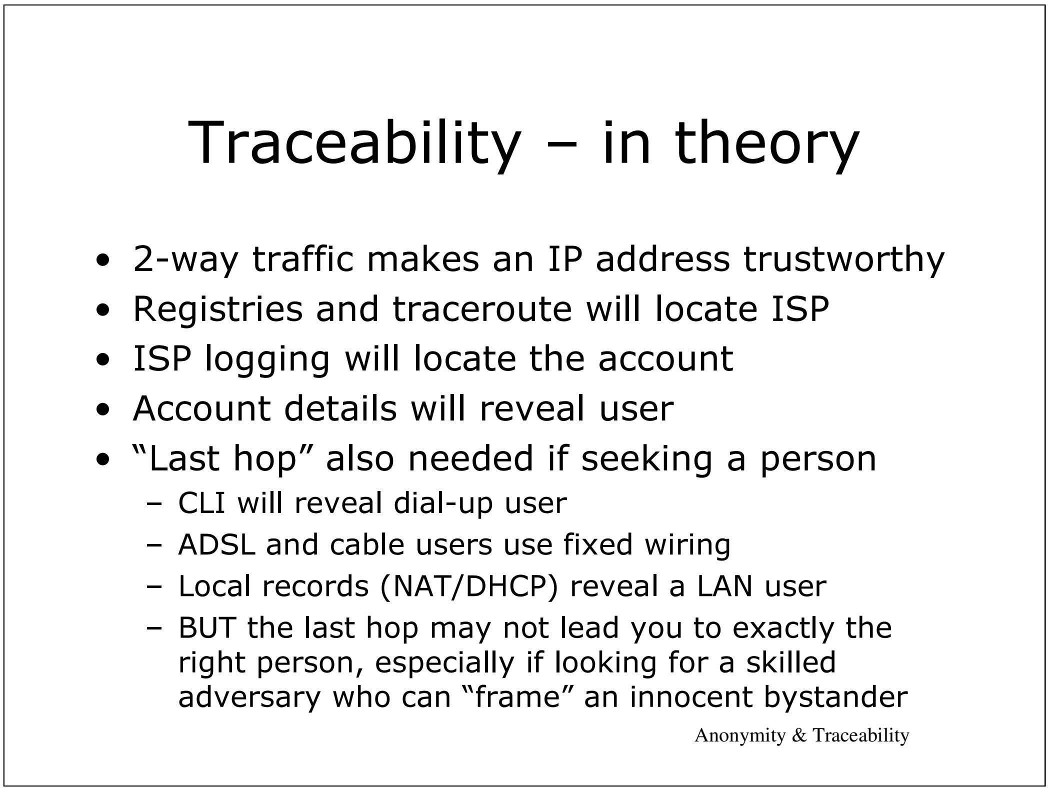
✶ In the UK, retention of logging data is currently governed by the *Data Protection Act 1998* and *The Privacy and Electronic Communications (EC Directive) Regulations 2003*. In general terms, under the DPA you may not keep data unless you have a business need to do so. The regulations set specific requirements for information relating to a "call". It is generally accepted that even where logs are not required for billing purposes, they can still be kept for a month (or six) in order to prevent "abuse" by customers. Thereafter they must be destroyed or anonymised.

✶ Keeping logs just in case the police need them is not a business need. However, the *Anti-Terrorism, Crime and Security Act 2001* creates a voluntary Code of Practice on keeping logs to prevent terrorism. If a voluntary code fails then the Secretary of State has powers to make it compulsory. The voluntary code is said to be working, but no details are released as to who has signed up to it.

✶ The *Cybercrime Convention* (first signed in 2001, yet to be ratified by the UK) contains provisions for data preservation (ie the storage of logs so that they are not destroyed) for up to 90 days and for "expeditious disclosure" of information that indicates the source of traffic.

http://conventions.coe.int/Treaty/en/Treaties/Html/185.htm

✶ The current situation on data retention in Europe remains fast moving. A good place to track events is on the EDRI website: http://www.edri.org/issues/privacy/dataretention



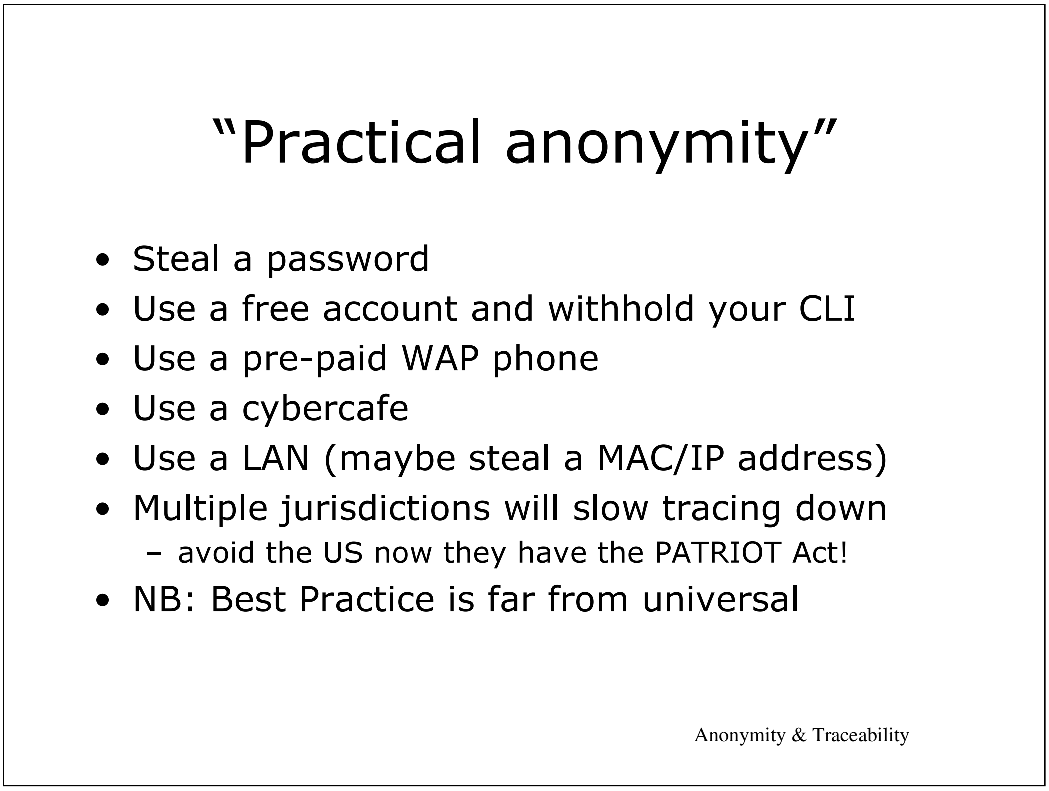
Traceability – in theory

- 2-way traffic makes an IP address trustworthy
- Registries and traceroute will locate ISP
- ISP logging will locate the account
- Account details will reveal user
- "Last hop" also needed if seeking a person
  - CLI will reveal dial-up user
  - ADSL and cable users use fixed wiring
  - Local records (NAT/DHCP) reveal a LAN user
  - BUT the last hop may not lead you to exactly the right person, especially if looking for a skilled adversary who can "frame" an innocent bystander

Anonymity & Traceability

✶ It should probably not be surprising that traceability over the "last hop" from an account to a user is poorly supported. Most of the traceability mechanisms are provided by ISPs and they discharge their obligations to the network by being able to locate a miscreant account and disable it. They have limited interest in locating a specific individual.

✶ For a discussion of how "last hop" traceability breaks down in a number of different Internet access technologies see my PhD thesis!



“Practical anonymity”

- Steal a password
- Use a free account and withhold your CLI
- Use a pre-paid WAP phone
- Use a cybercafe
- Use a LAN (maybe steal a MAC/IP address)
- Multiple jurisdictions will slow tracing down
  - avoid the US now they have the PATRIOT Act!
- NB: Best Practice is far from universal

Anonymity & Traceability

✶ One might reasonably take the view that where traceability fails, as in the slide, then there is some practical anonymity.

✶ If you steal a password then someone else will be blamed for your actions – except you’ll be traced through CLI.

✶ If you hide your CLI with 141 then you’ll be untraceable – except that the telco SS7 logs will locate you.

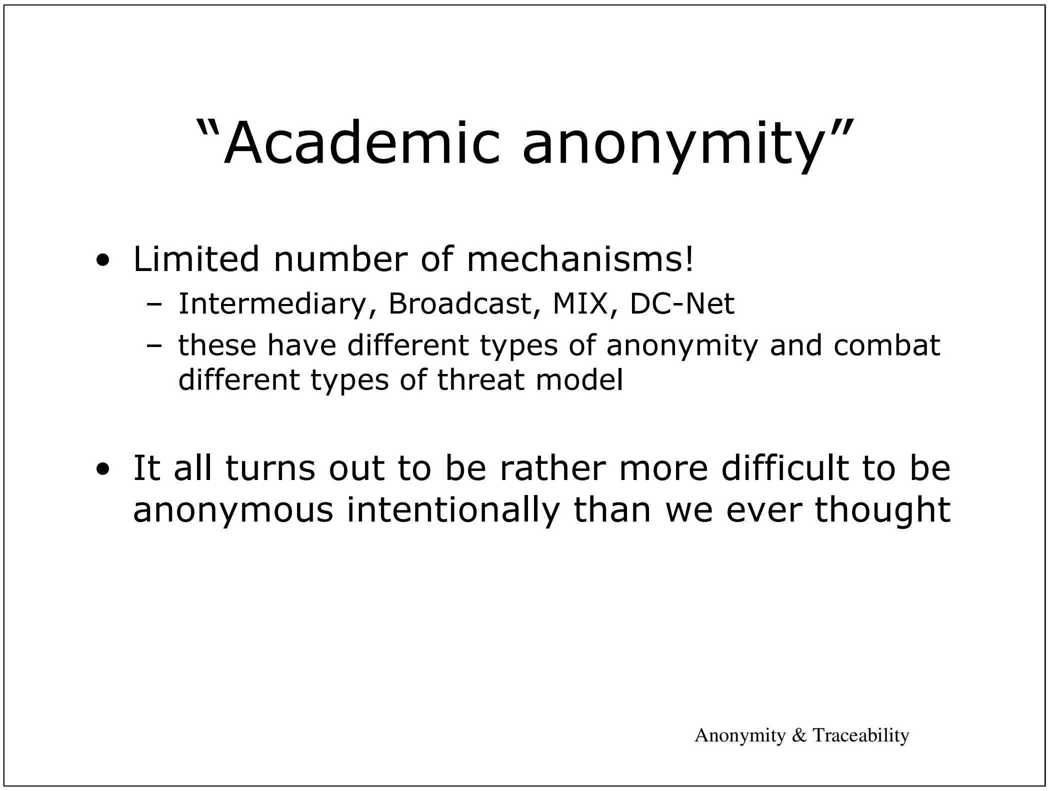
✶ If you use a prepaid mobile then all the telco will learn is your number – except they’ll work out your identity because you regularly ring up your mum and the curry house.

✶ If you use a cybercafe (and pay cash) then the trail will run cold if you are not still on the premises – except that they’ll have your face on CCTV and the congestion charge cameras will have snapped your car.

✶ If you impersonate someone else on your LAN then you may be hard to locate. But if you’re still online when they seek you then your machine can be fingerprinted and your network segment identified.

✶ If you use multiple jurisdictions then it may be too much trouble to chase you down. But beware of threatening human life and using any US resources because the PATRIOT act gives American law enforcement a great many powers to access traceability information.

✶ But practical anonymity may just come from a lack of “Best Practice” at your ISP or the remote machine that didn’t actually create any logs or know what time it was anyway.



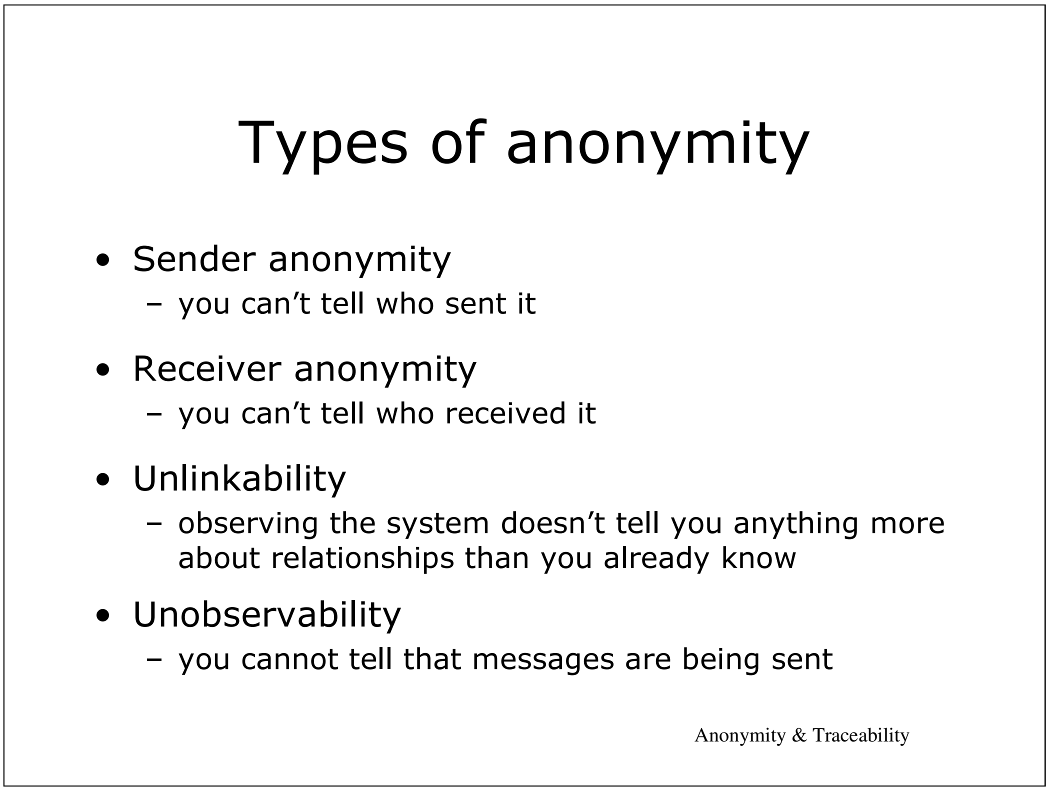
“Academic anonymity”

- Limited number of mechanisms!
  - Intermediary, Broadcast, MIX, DC-Net
  - these have different types of anonymity and combat different types of threat model
- It all turns out to be rather more difficult to be anonymous intentionally than we ever thought

Anonymity & Traceability

✶ The Anonymity systems discussed in the rest of the lecture prevent the secret police from knowing which of *n* people sent some traffic. When *n* is small they may lock them all up anyway. However, the academic study of anonymity mechanisms is more high minded than this and involves an assessment of whether a particular mechanism can cope with a particular threat model (a global adversary who can examine any traffic at will [think NSA and Echelon], an active attacker who can join your network and then misbehave, or perhaps an adversary with more limited powers who can only examine packet flow counters on some intermediate routers).

✶ Anonymity is becoming ever more complex (see the PhD theses of Danezis and Serjantov for much more on this) and it turns out to be considerably harder to provide a secure system than might have been thought a few years ago. In particular, a number of very powerful attacks against “real time” systems means that academics would not really consider them secure, although in practice your local secret police might have significant problems dealing with them – and would end up just locking you up for possessing subversive programs.



Types of anonymity

- Sender anonymity
  - you can’t tell who sent it
- Receiver anonymity
  - you can’t tell who received it
- Unlinkability
  - observing the system doesn’t tell you anything more about relationships than you already know
- Unobservability
  - you cannot tell that messages are being sent

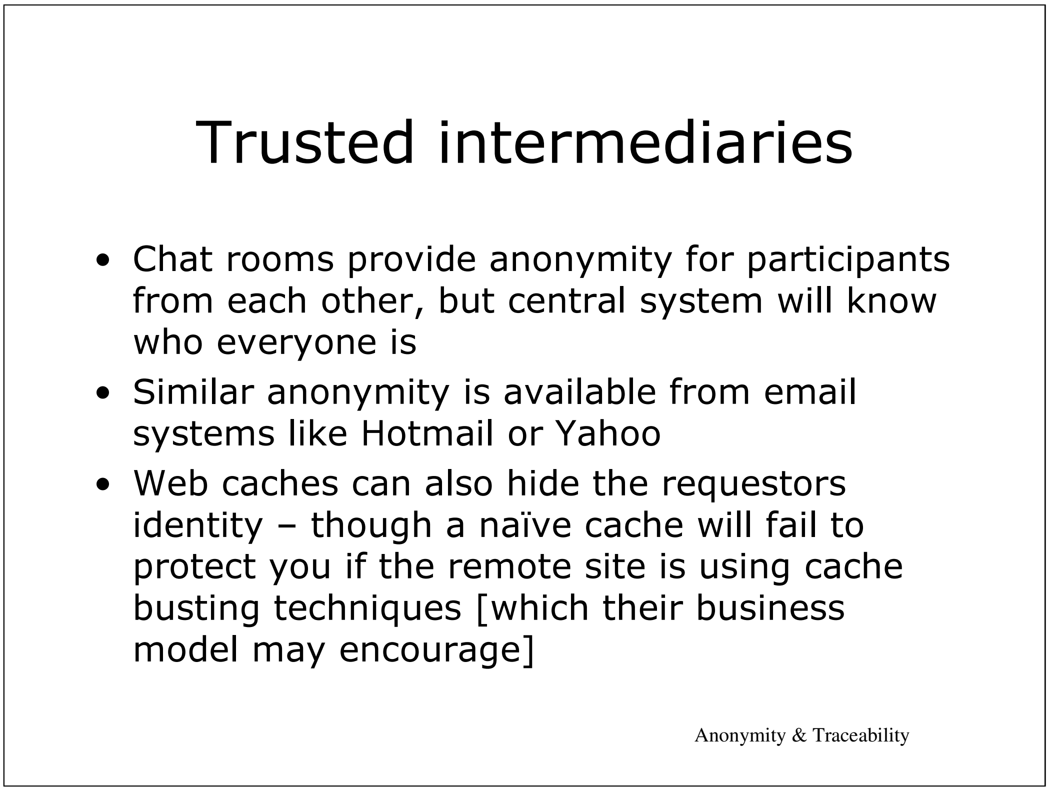
Anonymity & Traceability

✶ For some more terms and a deeper analysis of the issues see:

“Anonymity, Unobservability and Pseudonymity – a Proposal for Terminology”, Andreas Pfitzmann and Marit Köhntopp in: Hannes Federrath (Hg.): Designing Privacy Enhancing Technologies; Proceedings Workshop on Design Issues in Anonymity and Unobservability; LNCS 2009; 2001; 1-9.

This is an evolving document. For the latest version see:

http://dud.inf.tu-dresden.de/Anon_Terminology.shtml



Trusted intermediaries

- Chat rooms provide anonymity for participants from each other, but central system will know who everyone is
- Similar anonymity is available from email systems like Hotmail or Yahoo
- Web caches can also hide the requestors identity – though a naïve cache will fail to protect you if the remote site is using cache busting techniques [which their business model may encourage]

Anonymity & Traceability

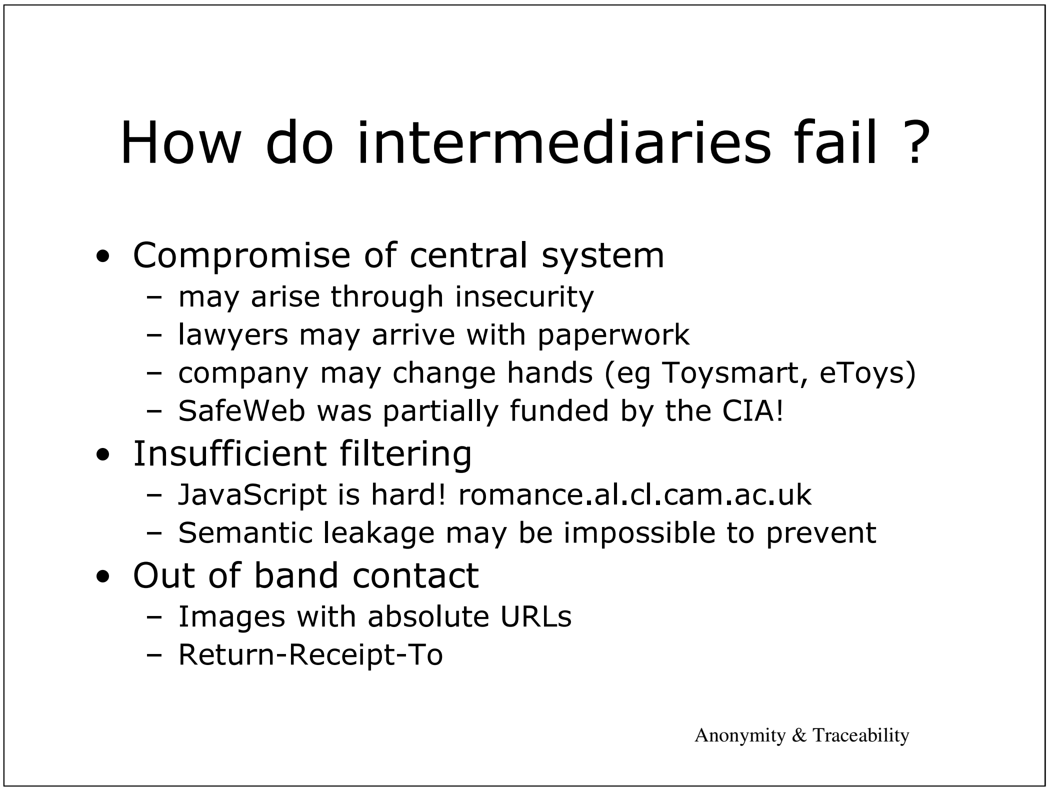
✶ Chat systems are extremely common on the Internet. Besides systems such as IRC or web-based chat on AOL, there are chat systems provided with most online games, and anyone who has played a MUD (Multi-User Dungeon) regularly will know that people come as much for the conversation as for the collecting the treasure, solving the puzzles or killing the dragons. These systems will usually provide “screen names” for the participants and their real IP address or email address will not be disclosed in public (though it will be apparent to the system operator).

✶ If you sign up for a Hotmail account (www.hotmail.com), then the sysadmins there will know where you connected from and the personal information you provided on your sign-up form. However, no-one else will be able to know who is actually behind a *aardvark512@hotmail.com* address (this may not be quite true for outgoing email because the IP address of the sender may be recorded in headers, but it’s certainly true for recipients).

✶ Caches hold local copies of remote content which can save external bandwidth and provide the information quickly. However, if the remote site has “banner ads” or is counting “hits” to demonstrate popularity then caches will reduce their income. Adding headers to say “do not cache” may not work against caches that themselves break the rules. Various tricks can be played with Java or JavaScript to create “dynamic pages” that caches cannot hold successfully, or you just give banner ad images unique names:

e.g.: http://www.clickz.com/tech/ad_tech/article.php/843731

✶ See also: http://www.cl.cam.ac.uk/~rnc1/Patterns_of_Failure.pdf



How do intermediaries fail ?

- Compromise of central system
  - may arise through insecurity
  - lawyers may arrive with paperwork
  - company may change hands (eg Toysmart, eToys)
  - SafeWeb was partially funded by the CIA!
- Insufficient filtering
  - JavaScript is hard! romance.al.cl.cam.ac.uk
  - Semantic leakage may be impossible to prevent
- Out of band contact
  - Images with absolute URLs
  - Return-Receipt-To

Anonymity & Traceability

✶ For a longer discussion of this topic see section 4 of:

"Real World Patterns of Failure in Anonymity Systems", Richard Clayton, George Danezis and Markus Kuhn. Information Hiding Workshop, Pittsburgh 2001, in Ira S. Moskowitz (ed.): Information Hiding 2001, LNCS 2137.

online at: http://www.cl.cam.ac.uk/~rnc1/Patterns_of_Failure.pdf

✶ There are numerous examples of legal "attacks" on intermediaries. See for example:

http://www.december.com/cmc/mag/1997/sep/helmers.html

about anon.penet.fi

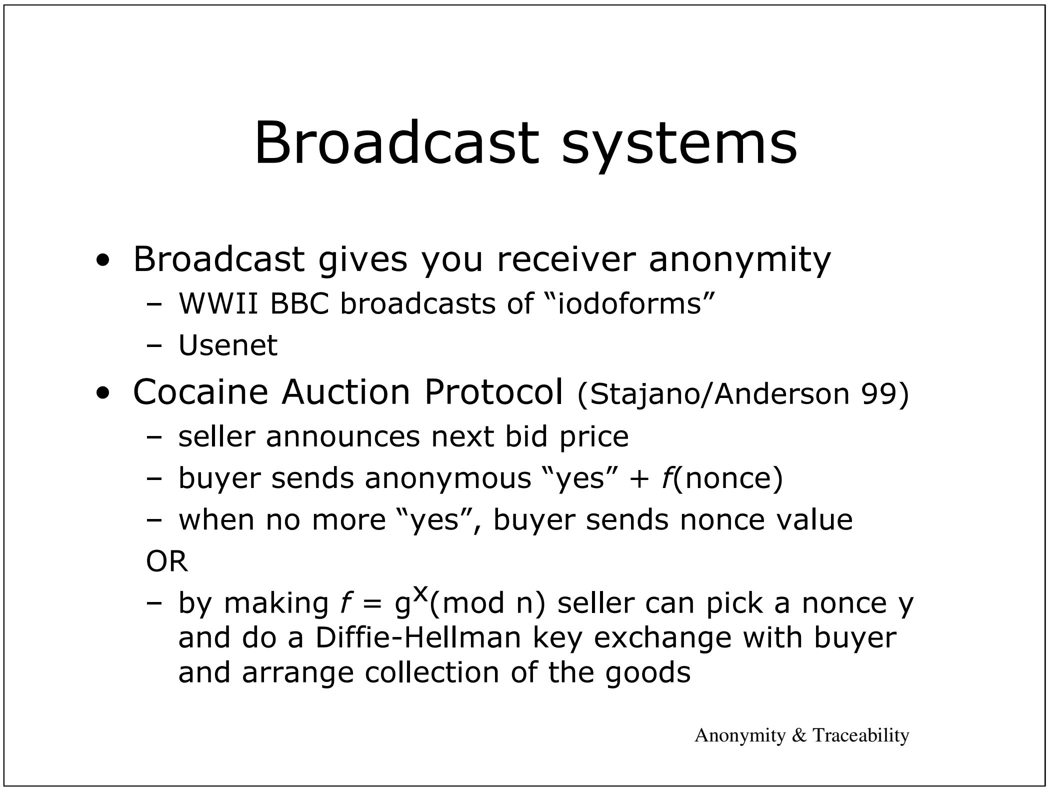
http://news.bbc.co.uk/1/hi/sci/tech/1231419.stm

Motley Fool & Interactive Investor International

http://www.eff.org/Privacy/Anonymity/cyberslapp.php

summary of US cases

✶ For a discussion of the various attacks that worked against the Cambridge "Romance" server (romance.al.cl.cam.ac.uk) in Michaelmas 2000 then see the paper cited above (most of the simple issues are now fixed).



Broadcast systems

- Broadcast gives you receiver anonymity
  - WWII BBC broadcasts of "iodoforms"
  - Usenet
- Cocaine Auction Protocol (Stajano/Anderson 99)
  - seller announces next bid price
  - buyer sends anonymous "yes" + *f*(nonce)
  - when no more "yes", buyer sends nonce value

  OR
  - by making f = g^x(mod n) seller can pick a nonce y and do a Diffie-Hellman key exchange with buyer and arrange collection of the goods

Anonymity & Traceability

✶ The WWII broadcasts that the BBC made on behalf of S.O.E. were called **iodoforms** "by someone with a classical education" (though apparently not a chemist since iodoform is CHI_3). Besides issuing coded commands they provided a way that agents could prove their bona fides (it would be arranged that a message of a doubter's choosing was broadcast later in the week). This proved to be of immense value in obtaining assistance from the locals.

The topic is covered in "Between Silk and Cyanide – a Codemaker's War 1941-1945", Leo Marks, HarperCollins 2000.

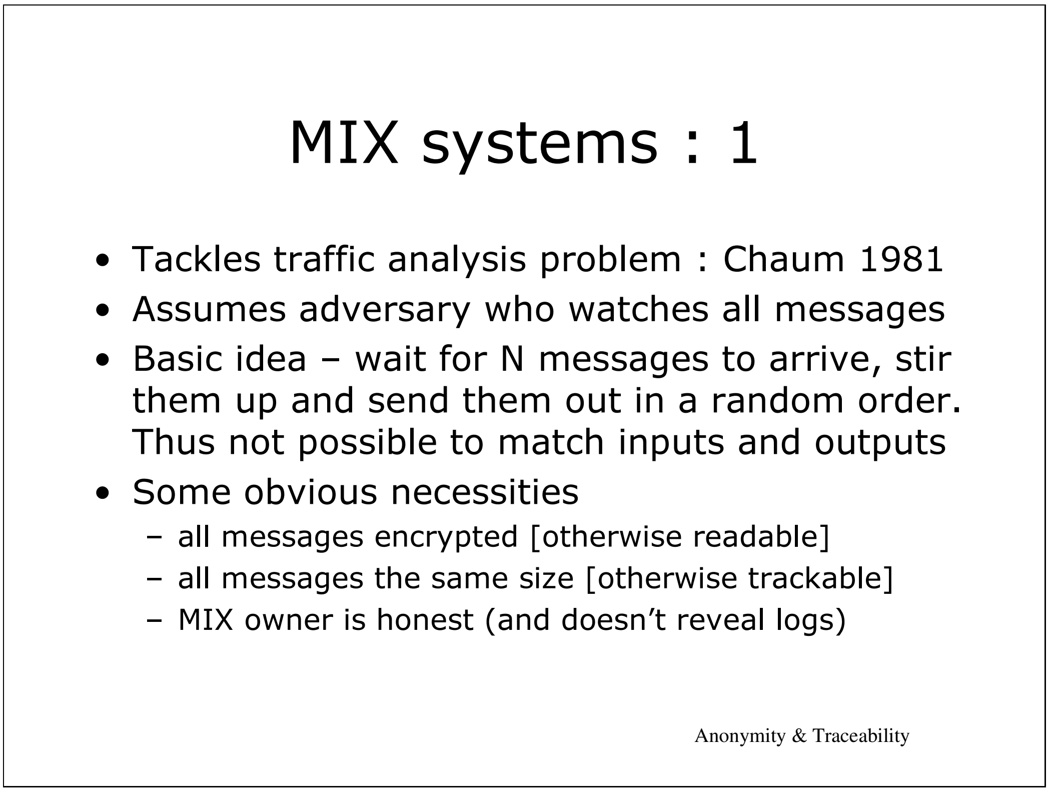
✶ The flood-fill algorithm used by Usenet is described in RFC977.

✶ "The Cocaine Auction Protocol: On The Power Of Anonymous Broadcast", Frank Stajano and Ross Anderson. Information Hiding Workshop, Dresden 1999 in A. Pfitzmann (ed), Information Hiding, 1999, LNCS 1768 pp434-447

http://www.cl.cam.ac.uk/~rja14/cocaine.pdf

The paper considers the protocol at rather more depth than is possible in the lecture. In particular it examines attacks such as what happens when the seller does not sell to the highest bidder, when the seller bids at his own auction and how to deal with "deadbeat bidders" who never show up with the money.

The paper also considers the issues surrounding broadcast as an anonymity primitive (raw broadcast not suitable for the Internet, but can be efficient on LANs or with short range wireless techniques – provided that attackers cannot use sophisticated electronics, such as direction finding kit, to monitor who is actually sending).



MIX systems : 1

- Tackles traffic analysis problem : Chaum 1981
- Assumes adversary who watches all messages
- Basic idea – wait for N messages to arrive, stir them up and send them out in a random order. Thus not possible to match inputs and outputs
- Some obvious necessities
  - all messages encrypted [otherwise readable]
  - all messages the same size [otherwise trackable]
  - MIX owner is honest (and doesn't reveal logs)

Anonymity & Traceability

✶ Original paper is straightforward to read: "Untraceable electronic mail, return addresses, and digital pseudonyms", David Chaum, Communications of the ACM, 24(2), 1981, pp 84-88.

Online at: http://world.std.com/~franl/crypto/chaum-acm-1981.html

Also recommended is: "Mixing E-mail with BABEL", C.Gülcü & G.Tsudik, ISOC Symposium on Network and Distributed System Security, Feb 1996.

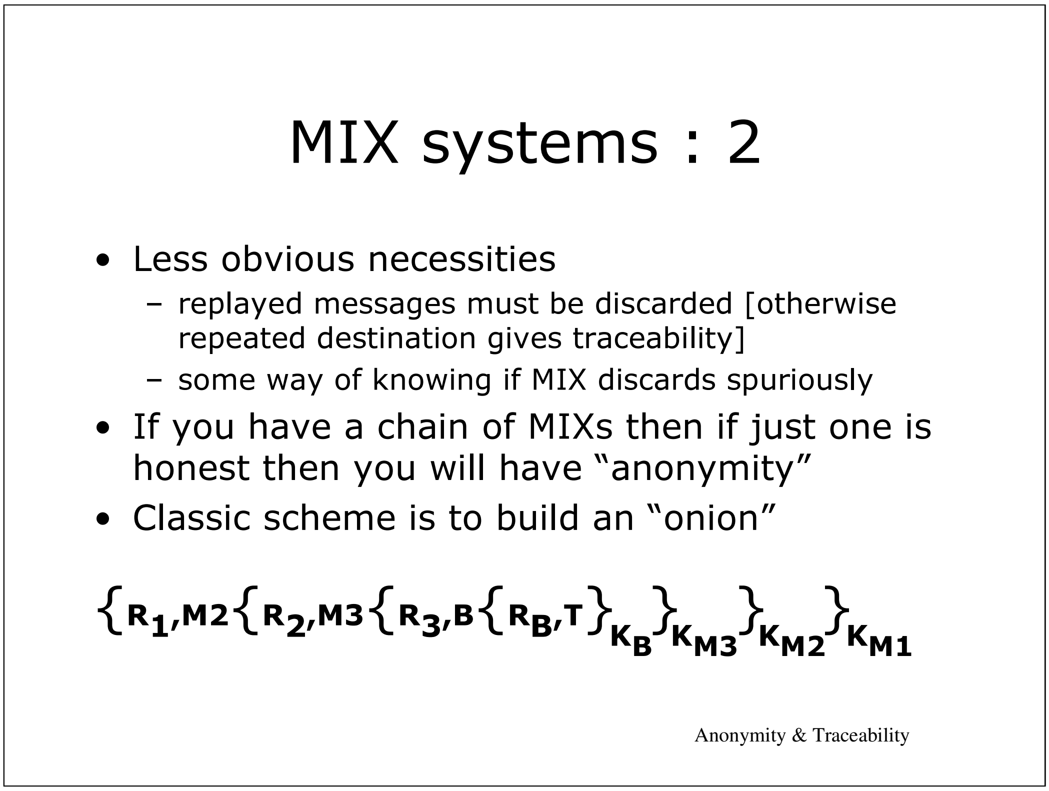
Online at: http://www.ics.uci.edu/~gts/paps/guts96.ps.gz

✶ The messages are encrypted not just to make them unreadable per se, but because otherwise it would be simple to detect which input corresponded to which output (the unlinkability property), and thereby trace the sender.

✶ If it takes a long time for N messages to arrive, then the MIX will do nothing and so messages can be delayed for a long time. Various schemes have been proposed for timed MIXs and "pool MIXs" where some messages are retained and new arrivals mixed in with them, so that the "anonymity set" is not just the most recent messages, but every message the MIX has ever received. For a modern review paper on MIX types see:

"From a Trickle to a Flood: Active Attacks on Several Mix Types", Andrei Serjantov, Roger Dingledine & Paul Syverson, in Fabien Petitcolas (Ed), Proc. 5th Workshop on Information Hiding, October 2002, LNCS

http://www.cl.cam.ac.uk/~aas23/taxonomy.pdf



MIX systems : 2

- Less obvious necessities
  - replayed messages must be discarded [otherwise repeated destination gives traceability]
  - some way of knowing if MIX discards spuriously
- If you have a chain of MIXs then if just one is honest then you will have “anonymity”
- Classic scheme is to build an “onion”

$$\{R_1,M2\{R_2,M3\{R_3,B\{R_B,T\}_{K_B}\}_{K_{M3}}\}_{K_{M2}}\}_{K_{M1}}$$

Anonymity & Traceability

✶ Chaum did not call the multi-mix messages “onions”. This term only became popular with the arrival of “onion routing” in the late 90s.

✶ Discarding duplicates can be very important. If a MIX has a batch size of 100 and, over time, distributes messages evenly to 10,000 different recipients then the probability that the next batch will contain another message to the same recipient as in this batch is about 1 in 100.

✶ Note that even if all MIXs in a chain are controlled by the NSA except for one, then the anonymity is still being achieved. One uses a chain to ensure that you aren’t trusting one particular system operator to be honest.

✶ The onion on the slide is sent to MIX M1, which will peel off a layer to reveal the next destination as M2. Then M2 will peel off the next layer and send it to M3. M3 will then discover the ultimate destination to be B. B will be able to remove the final layer of encryption and read the message text T.

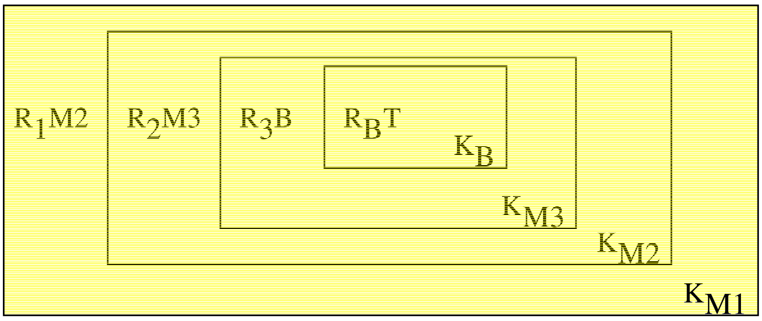
The K_n values are encryption keys. It’s usual to use Public Key Encryption (such as RSA) so that messages can be encrypted to an entity’s public key and only they will hold the private key and be able to decrypt it.

✶ The various “nonce” values R_1, R_2, R_3 and R_B are chosen randomly by the originator of the message.

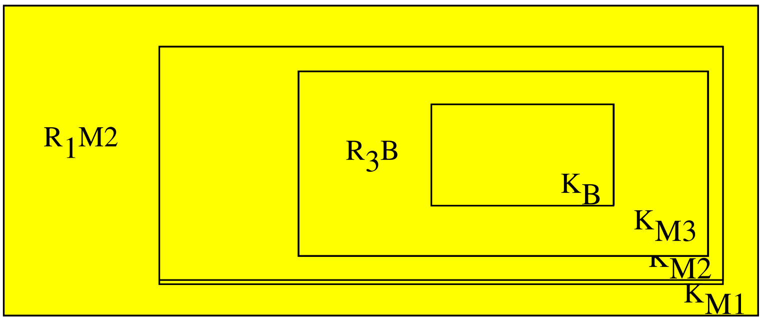
R_B is present to ensure that the message delivered to B cannot be determined. Otherwise (in encryption schemes such as RSA) one could encrypt “flee at once, all is discovered” (or any other message) using K_B and spot its arrival.

The other values R_n are present to prevent an attacker from matching up any input and output packets.

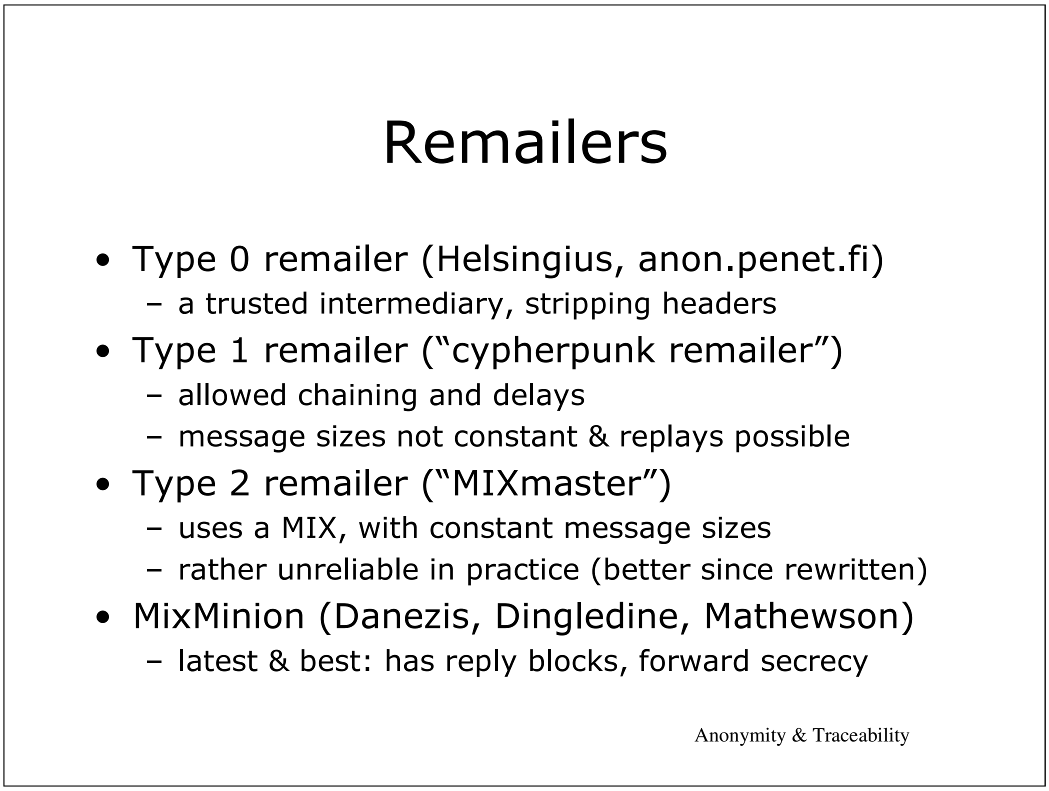
$$\left\{R_1,M2\left\{R_2,M3\left\{R_3,B\left\{R_B,T\right\}_{K_B}\right\}_{K_{M3}}\right\}_{K_{M2}}\right\}_{K_{M1}}$$



$$\{R_1,M2\{R_2,M3\{R_3,B\{R_B,T\}_{K_B}\}_{K_{M3}}\}_{K_{M2}}\}_{K_{M1}}$$



[illegible]



Remailers

- Type 0 remailer (Helsingius, anon.penet.fi)
  - a trusted intermediary, stripping headers
- Type 1 remailer ("cypherpunk remailer")
  - allowed chaining and delays
  - message sizes not constant & replays possible
- Type 2 remailer ("MIXmaster")
  - uses a MIX, with constant message sizes
  - rather unreliable in practice (better since rewritten)
- MixMinion (Danezis, Dingledine, Mathewson)
  - latest & best: has reply blocks, forward secrecy

Anonymity & Traceability

✶ anon.penet.fi stripped all incoming headers, but recorded the "from" address. It automatically generated a pseudonym which could then be used by recipients for return email. The server would then deliver the email to the person who wrote the original message.

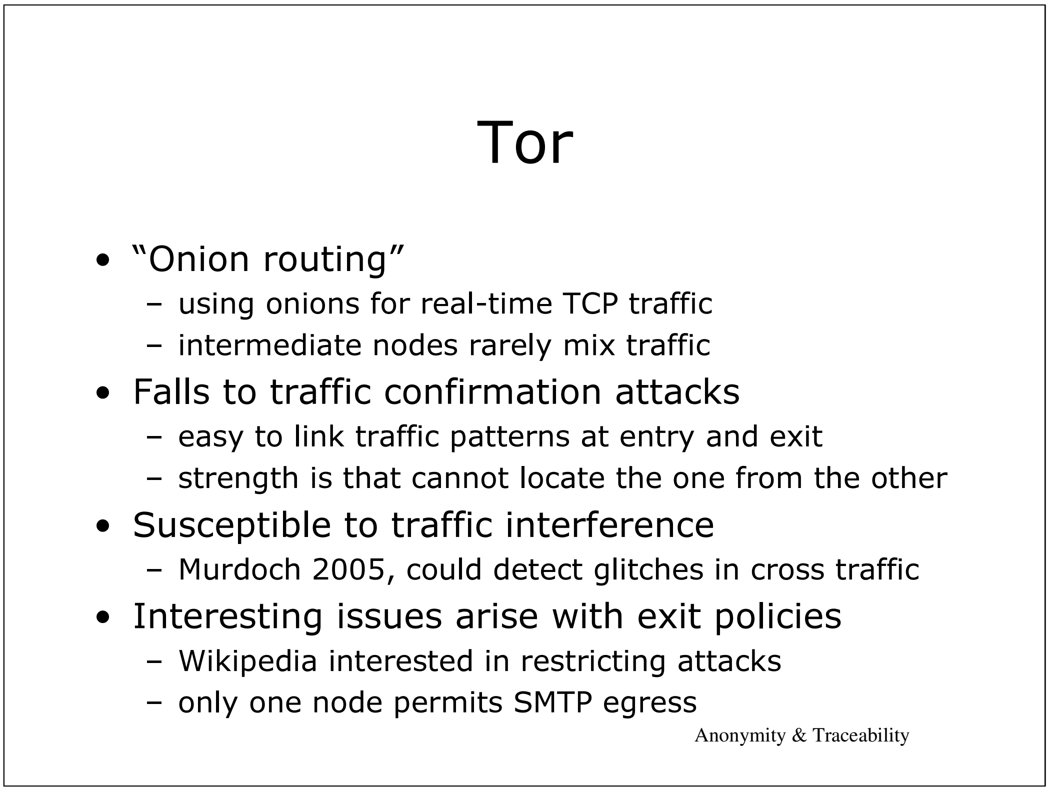
anon.penet.fi had strict limitations on message size, making it unsuitable for anything but short text messages. Attachments of pictures would be too big to be transmitted.

anon.penet.fi was shut down in 1996 when the Scientologists succeeded in a legal action to force the operator to divulge the real email address hidden behind a pseudonym.

✶ Cypherpunk remailers still exist, but are considered to be insecure against an attacker who can monitor their activities.

✶ There are about 25 MIXmaster remailers running, of which less than half regularly achieve "4 nines" reliability. There are slightly more MIXminion systems, with a higher percentage being highly reliable.

✶ For more about MixMinion including detailed explanations of its mechanisms and the rationale for its design see http://mixminion.net



Tor

- “Onion routing”
  - using onions for real-time TCP traffic
  - intermediate nodes rarely mix traffic
- Falls to traffic confirmation attacks
  - easy to link traffic patterns at entry and exit
  - strength is that cannot locate the one from the other
- Susceptible to traffic interference
  - Murdoch 2005, could detect glitches in cross traffic
- Interesting issues arise with exit policies
  - Wikipedia interested in restricting attacks
  - only one node permits SMTP egress

Anonymity & Traceability

✶ Full details of Tor are linked from http://tor.eff.org/

✶ There are an estimated 100,000 Tor users, routing their traffic through about 900 volunteer Tor servers on six continents (one of these servers is in the Computer Lab).

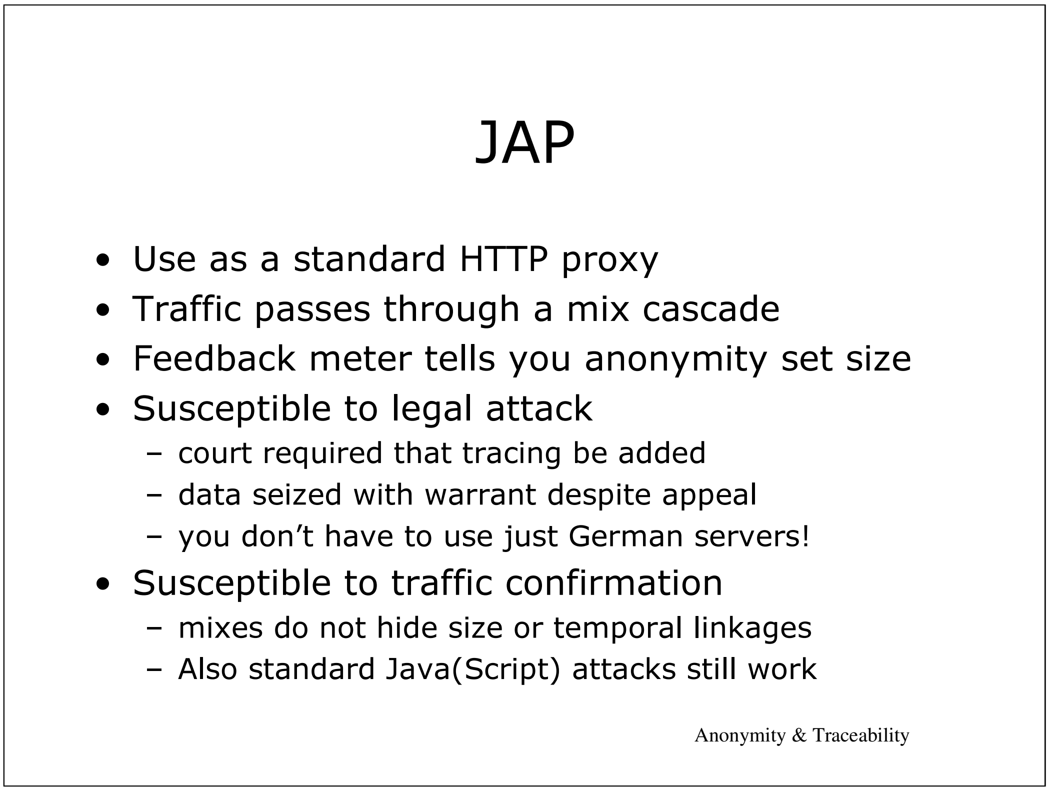
✶ Tor works by encrypting TCP streams within “onions” – as earlier, but using symmetric keys negotiated with each node on the path; which permits the reverse traffic to be returned to the sender.

Because the system is “real time”, the anonymity comes from the assumption that adversaries cannot observe all nodes – and hence packets cannot be tracked through the network.

An interesting attack was found by Steven Murdoch, who showed that he could detect which intermediate nodes were being used by a data stream by sending other traffic through those nodes.

Steven J. Murdoch, George Danezis: Low-Cost Traffic Analysis of Tor, 2005 IEEE Symposium on Security and Privacy, Oakland, California, USA.
http://www.cl.cam.ac.uk/users/sjm217/papers/oakland05torta.pdf

✶ Some of Tor’s users are using their anonymity to misbehave. Although most nodes block email traffic (so it cannot be used for harassment or simple spamming) problems arise with Usenet articles (created via the web interface at Google Groups) and abuse of Wikipedia (creating wikispam to link to inappropriate sites or just editing wars when there is disagreement about the content of pages). There are some interesting research questions here!



JAP

- Use as a standard HTTP proxy
- Traffic passes through a mix cascade
- Feedback meter tells you anonymity set size
- Susceptible to legal attack
  - court required that tracing be added
  - data seized with warrant despite appeal
  - you don’t have to use just German servers!
- Susceptible to traffic confirmation
  - mixes do not hide size or temporal linkages
  - Also standard Java(Script) attacks still work

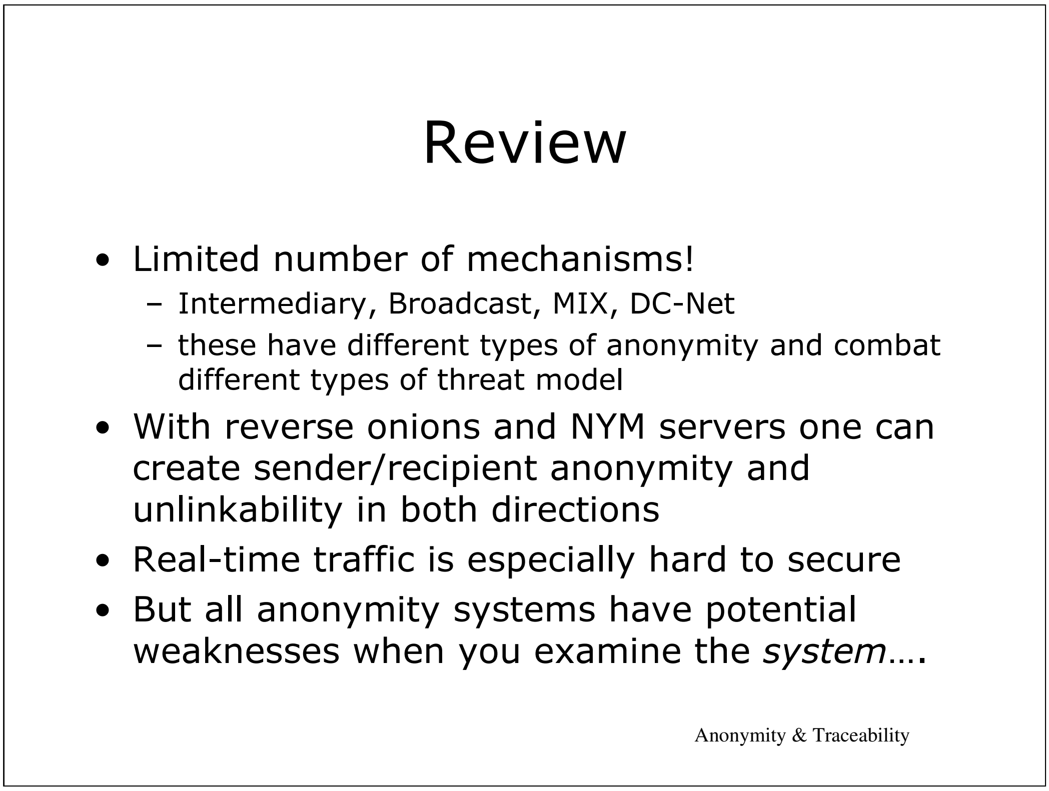
Anonymity & Traceability

✶ JAP (Java Anonymizing Proxy) is a project run by the Universities of Regensburg and Dresden. http://anon.inf.tu-dresden.de/index_en.html

✶ The system is a connected series of mix machines, with pre-determined linkages (a “mix cascade”), but several such cascades are available.

✶ The system is used for real-time web-browsing but the cascades permit the system to offer you good feedback on how many other users your traffic is being intermingled with. Once again the system is susceptible to traffic confirmation attacks and it has, in practice, shown itself to be subject to court sanctioned interference.

✶ It is one of the simpler anonymity systems to use (Tor for example is in the process of running a competition to provide a more friendly user interface).



Review

- Limited number of mechanisms!
  - Intermediary, Broadcast, MIX, DC-Net
  - these have different types of anonymity and combat different types of threat model
- With reverse onions and NYM servers one can create sender/recipient anonymity and unlinkability in both directions
- Real-time traffic is especially hard to secure
- But all anonymity systems have potential weaknesses when you examine the *system*….

Anonymity & Traceability

✶ You can create real anonymity today – which can be of real use

anonymous helplines for victims/sufferers

whistleblowers, police informants

feedback to lecturers

refereeing of conference/journal papers

privacy – hiding from marketeers

privacy – hiding from your boss (or future boss)

privacy – hiding from your mum (spouse, or the Chief Whip)

social and political movements

criminals!

BUT

✶ The literature is full of real-world attacks on anonymity systems…

… for example, if a system is creating an onion to send through a MIX system then it might be interesting to inspect its DNS traffic and see which addresses it is looking up. This may yield B, M1, M2 and M3 directly!

✶ When you're assessing a system for its anonymity properties you have to look at the whole system – not just the specialist mechanism(s) provided by the academics.