Just how secure is the TCP/IP protocol?

Andy Leigh
Information Security Strategist
BBC UK

Technology Direction | Technology Leadership Across The BBC

What I will cover:

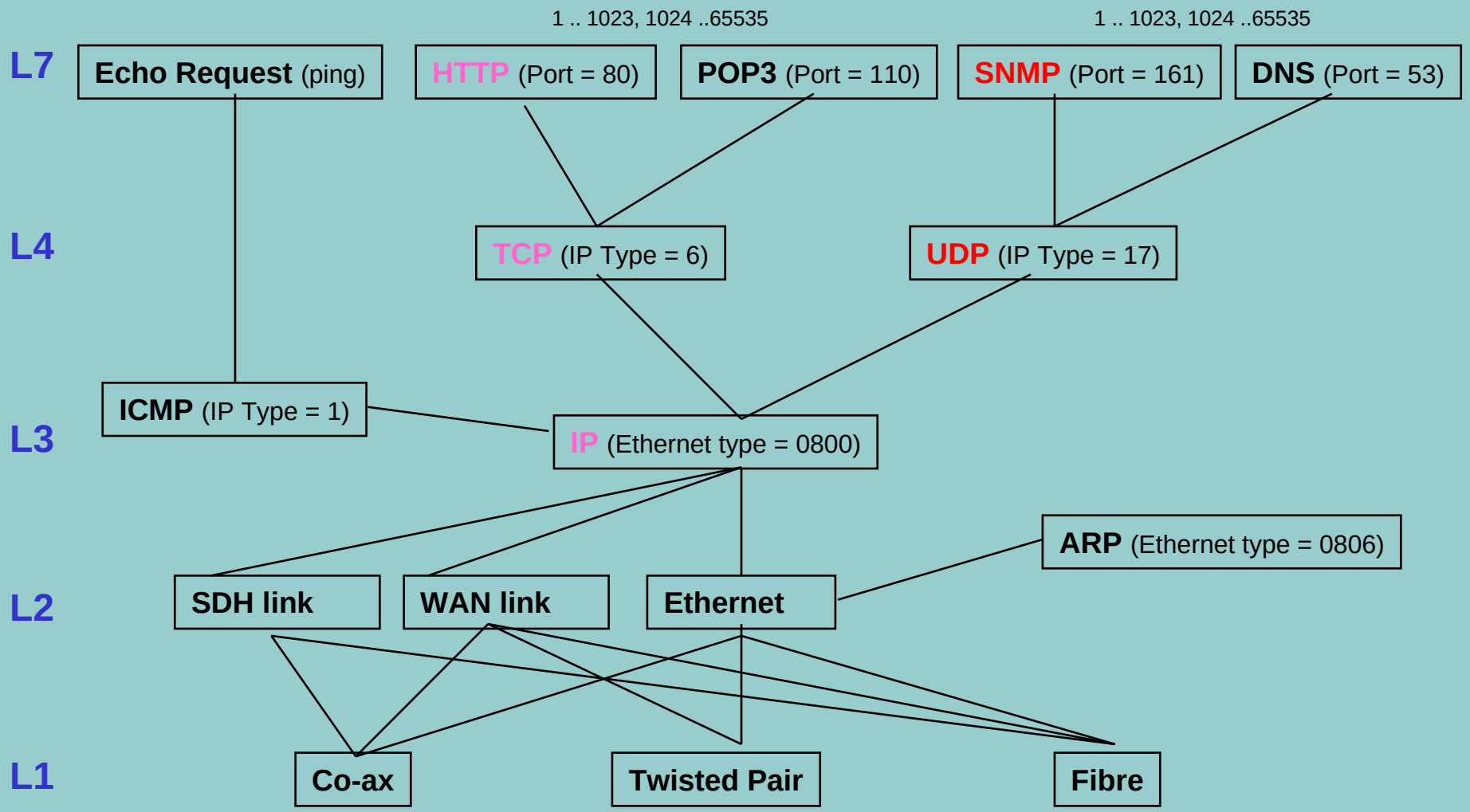
Packet Networks
- So, just what is TCP/IP?
- Packets and circuits – a security comparison
- How does an IP packet get from there to here?

Security Primer
- The whole security picture
- The principles of good security
- How hacks happen

TCP/IP Security
- General IP security issues
- TCP specific security issues
- Some Solutions

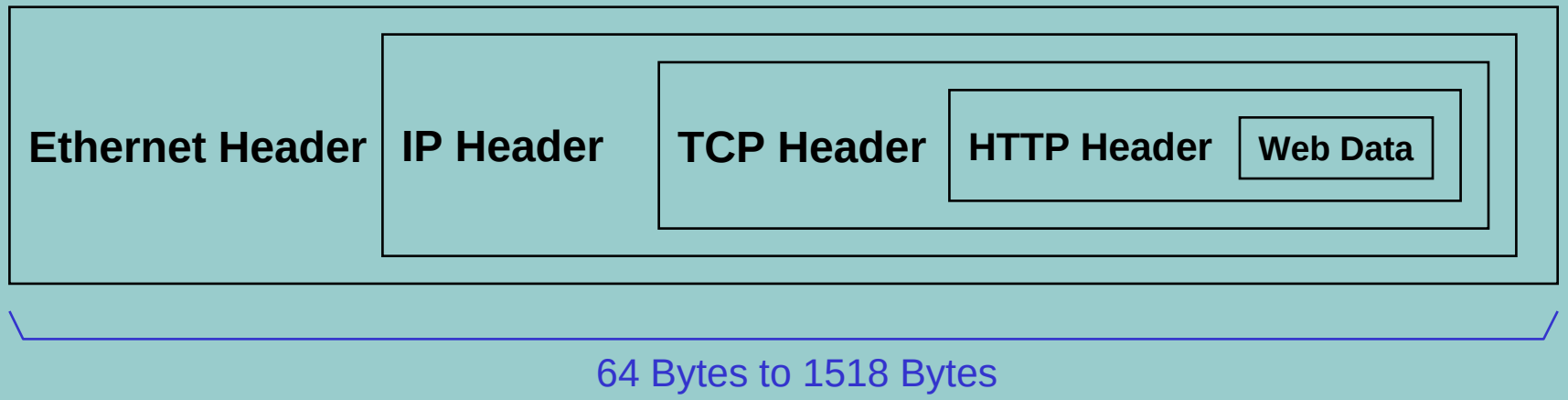
Conclusions

So, just what is TCP/IP?



BBC

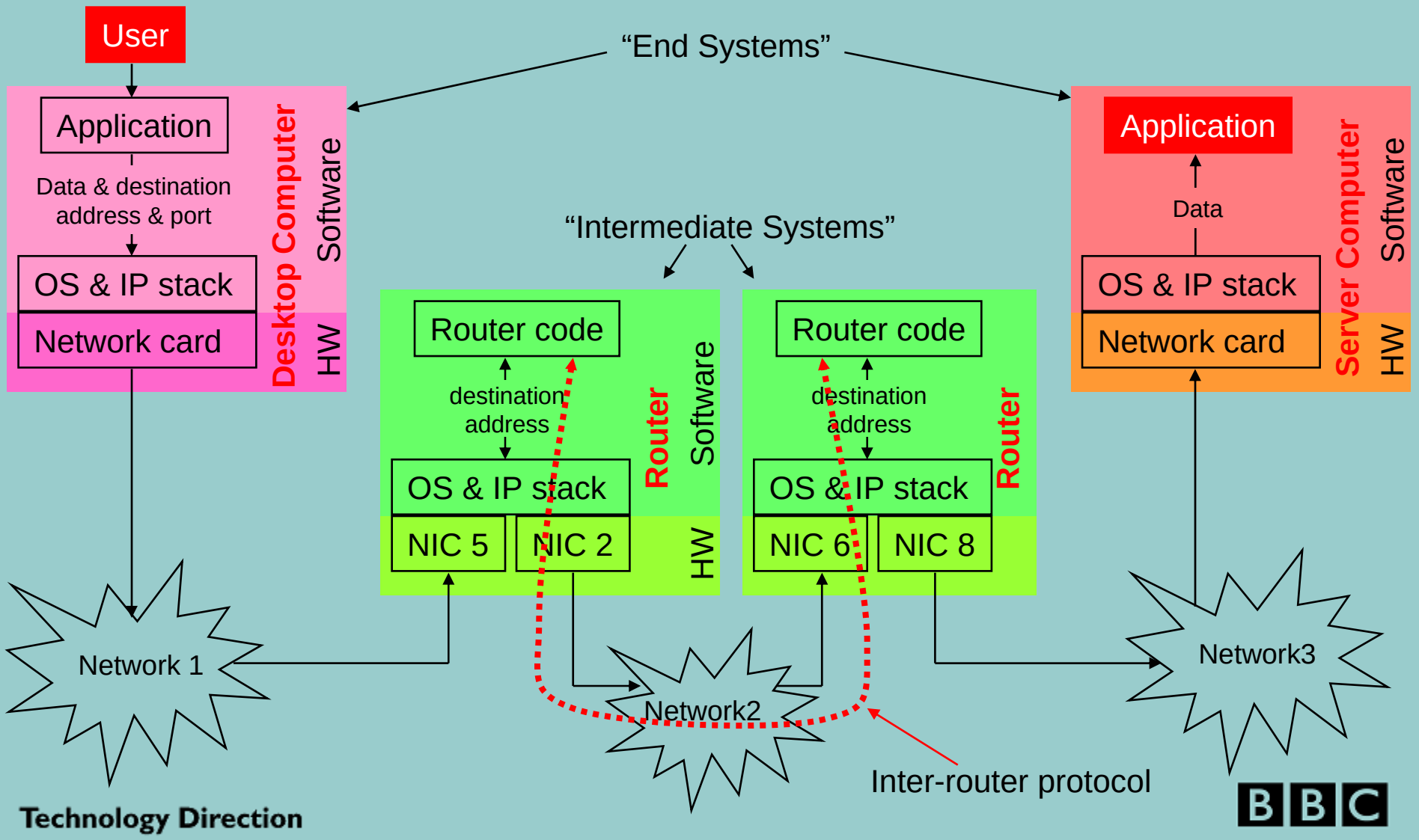
Wrapping up packets



Packets and circuits – comparing security features

	Permanent	Dialled	Packet (TCP/IP)
End-to-end connection setup	Yes	Yes	IP=N/A, UDP=N/A, TCP=Yes
End-to-end connection tear-down	Yes	Yes	IP=N/A, UDP=N/A, TCP=Yes
Same route during the connection	Yes	Yes	No
Same route for weeks	Yes	No	No
Service Provider does setup/tear-down	Yes	No	No
End "user" does setup/tear-down	No	Yes	IP=N/A, UDP=N/A, TCP=Yes
Connectionless "letter" mode	No	No	IP=Yes, UDP=Yes, TCP=No
Public global addresses used	No	Yes	Yes
Route decided by destination address	No	Yes	Yes
Global address self-assignable	No	No	Yes

How does an IP packet get from here to there?



Technology Direction

BBC

The bigger security picture

TCP/IP is not to blame for many computer security issues

We also need to consider:

- Badly specified, designed, installed or operated technologies such as:
  - Misconfigured Intermediate Systems (routers, firewalls)
  - Weak network-glue (DNS, authentication systems)
  - Flawed or unpatched Operating Systems
  - Bug-ridden or unpatched applications, databases etc.
- Badly trained and poorly informed end-users and support personnel
- The attackers themselves

The (simple) principles of good security

1. Never trust a network
2. Authenticate everything and everyone
3. Build systems to survive attacks

Saltzer & Schroeder's 1975 principles

DO NOT SHARE RESOURCES UNLESS YOU HAVE TO

1. Economy of mechanisms
   - keep design simple
2. Fail-safe defaults
   - "deny all" unless approved
3. Complete mediation
   - every access on every object checked for authority
4. Open design
   - don't depend on obscurity
5. Separation of privilege
   - 2 separate keys (or devices) better than one
6. Least privilege
   - users and software operate with least privilege necessary to get the job done
7. Least common mechanism
   - minimise mount of kit shared by, or critical to, more than user
8. Psychological acceptability
   - make the process easy to use so people just accept it

How hacks happen – a refresher

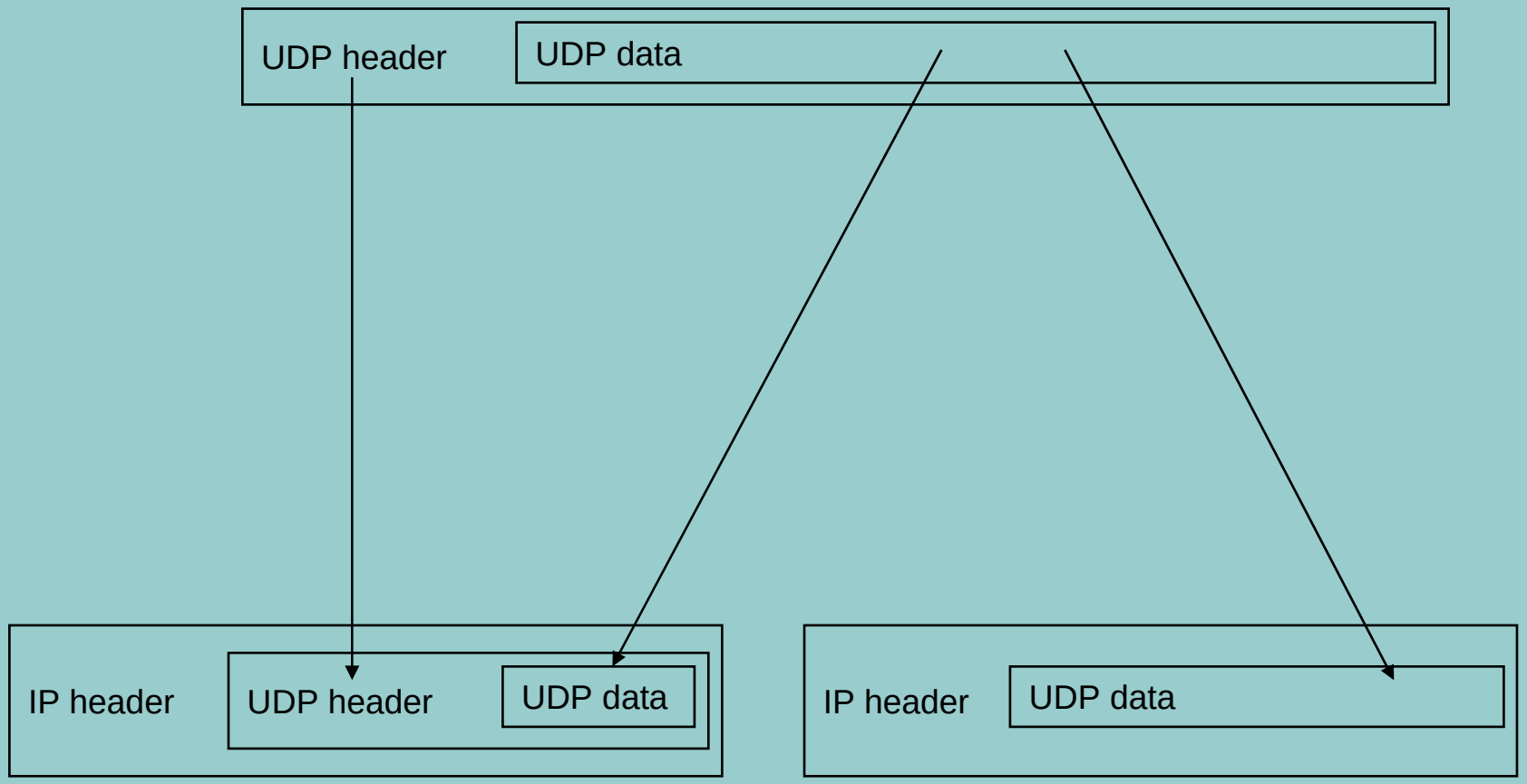
- Discovery and fingerprinting
- Scanning for vulnerabilities
- Analysing and looking for weaknesses
- Gaining low-level access
- Escalating to high-level access
- Inserting a back-door
- Cleaning up and exiting

Some attacks exploit weak passwords (many don't)
It's all over in minutes

General IP/UDP security issues

- UDP is “stateless” and is easy to forge
  - But it’s needed for one-way streams and multicast
- Checksums are not a security technology
  - Anyone can change the packet details or fake a packet and easily get the CRC correct
  - Each router has to change the IP checksum anyway to deal with hop-counts (TTL)
- IP source addresses are not guaranteed to be correct
  - Anyone can send a packet with fake source address
  - IP addresses are therefore NOT a form of authentication
- IP breaks packets into fragments if they are too big for the link
  - Higher-level headers carried only in the first fragment (second fragment may be something completely different)
  - Only end-stations are allowed to re-assemble

IP fragmentation

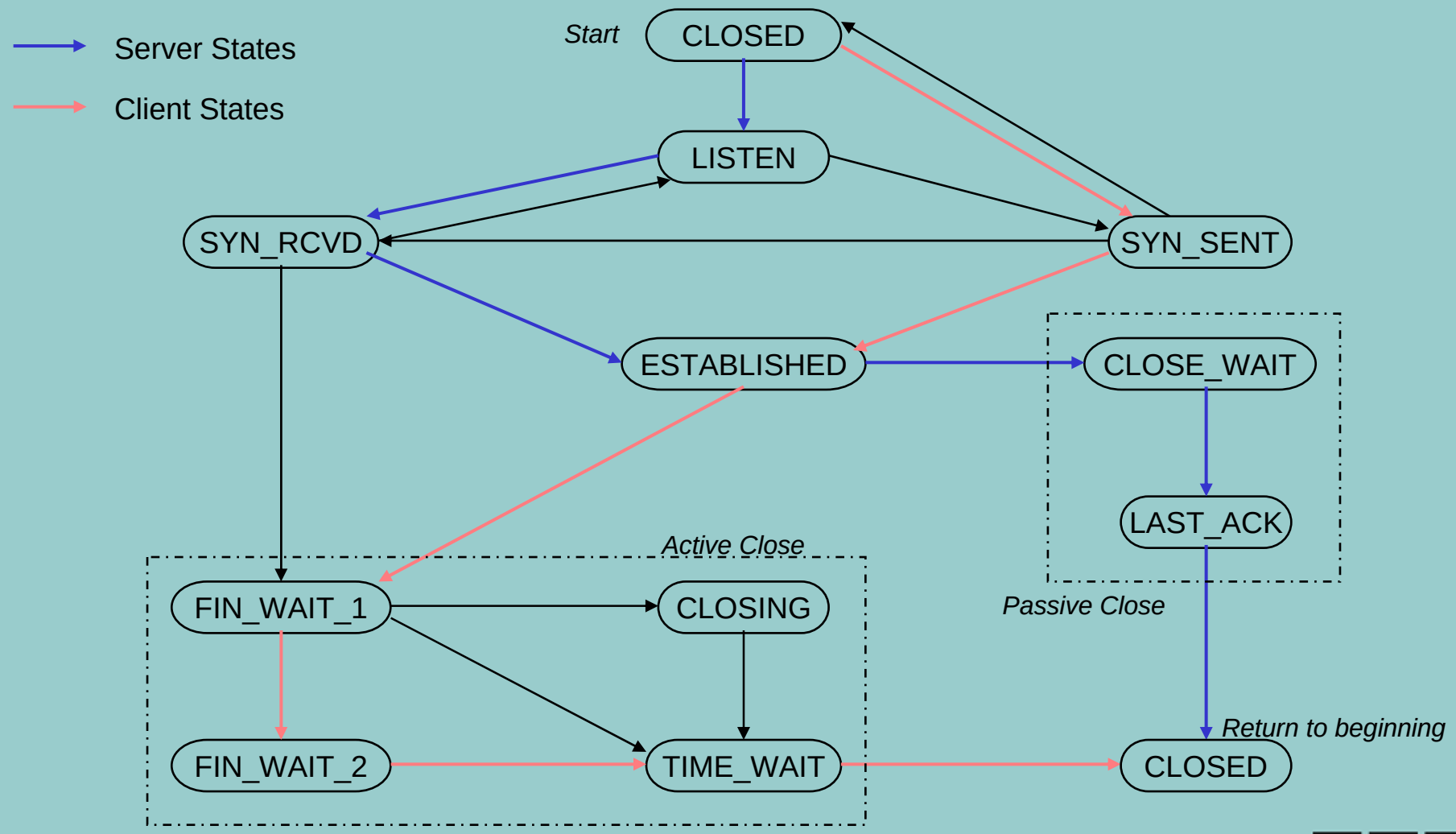


TCP specific security issues – TCP states

- TCP states – Denial Of Service weaknesses
  - An established TCP connection is open “forever” unless keep-alive timers (optional) are invoked
  - SYN attacks – each incoming “SYN” packet hold a listen-queue connection open for 75 seconds
  - Double-ended synchronisation is possible

NB these are all “flaws” in the original specification – most real-world Operating Systems have fixed the problems

The TCP finite-state-machine



TCP specific security issues

- Poor randomisation in the Initial Sequence Number
  - TCP's handshake invokes a "random" Sequence Number which should be hard to guess
  - In practice many implementations have been very predictable
  - TCP-connection spoofing is therefore possible
    - Connection hijacking
    - RST Denial Of Service
    - FIN Denial Of Service

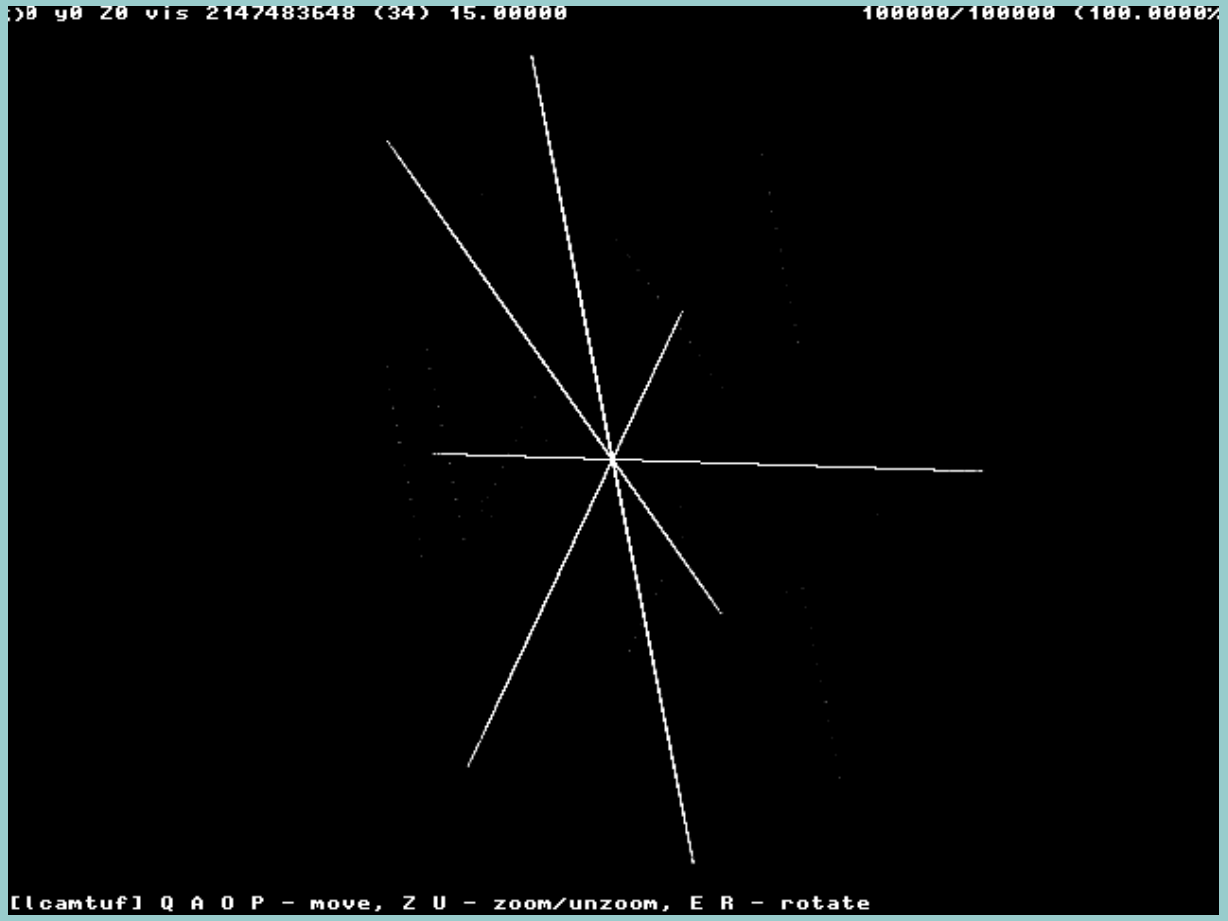
Some real world ISN “random numbers”

From “Strange Attractors and TCP/IP Sequence Number Analysis - One Year Later” by Michal Zalewski

Cisco’s IOS 12.2.10a

Some real world ISN “random numbers”

From “Strange Attractors and TCP/IP Sequence Number Analysis - One Year Later” by Michal Zalewski

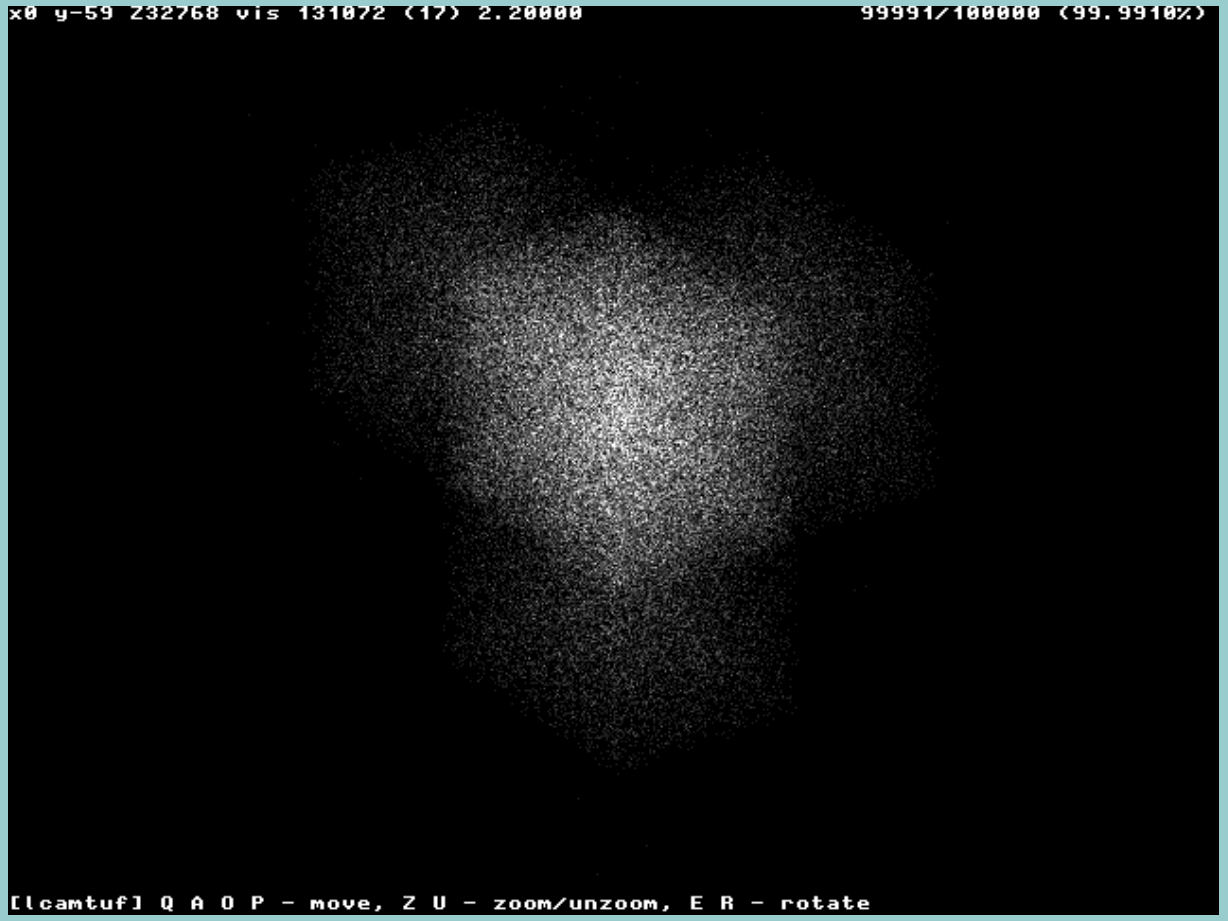


Some real world ISN “random numbers”

From “Strange Attractors and TCP/IP Sequence Number Analysis - One Year Later” by Michal Zalewski

Some real world ISN “random numbers”

From “Strange Attractors and TCP/IP Sequence Number Analysis - One Year Later” by Michal Zalewski



Win XP

Other “TCP/IP” security issues

- Name-spoofing
- Attacks against routing algorithms
- ICMP “Redirect” DOS attack
- ICMP “Destination Unreachable” DOS attack
- SNMP
- ARP spoofing

Solutions to TCP/IP's problems

- Re–write TCP/IP
- Move to IPv6
  - IPv6 Authentication Header
  - IPv6 ESP and Encryption Header
- Use IPSec to secure point-to-point connections
  - Looks remarkably similar to IPv6 security solutions (AH, ESP)
- "Harden" any common equipment & use application-level protection (e.g. SSL)
  - And follow good security operational practices
- Build layered security using choke-points (e.g. using firewalls) …

Solutions to TCP/IP's problems – Firewalls

Positives

- Stateful filtering firewalls
  - Watch the "state" of any connections to ensure they are valid
  - Can prevent most network attacks
- Proxy firewalls
  - Don't pass through packets (if well-built, are impervious to network attacks)
  - They communicate application-to-application

Negatives

- Firewalls cannot stop application-level attacks
- They affect performance
- Secure rules sometimes impossible in real-world
- They don't prevent snooping

Conclusions

- Packet networks are inherently less secure than switched-circuit networks
- There's a great deal more to security than secure networking
- The basic principles of good security:
  - Never trust a network
  - Always authenticate every transaction
  - Build systems to survive
- IP and associated protocols have a number of security flaws
- TCP has a number of inherent security flaws (*which have been mitigated by most real-world implementations*)
- Firewalls, hardening, IPv6 and encryption can help

Good security is essential, but can affect performance

Broadcasters must agree amongst themselves good security standards

Thanks for Listening

Any Questions?