Heartbleed

Legacy Systems

Heartbleed
CBS

Heartbleed

Implementation of SSL/TLS

- Example: Visit a website, observe HTTPS in the address bar - secure connection.
  - Encrypting traffic to ensure confidentiality of the data passing over the connection.
  - Secured socket or connection between the user and the server.

Heartbleed

Implementation of SSL/TLS

- Maintaining the socket, costs some resources or energy - which is problematic or worth considering if you have a lot of servers and lots of connections.
  - A server can only support a number of sockets or number of connections at any one time.
- Server can only manage so many sockets or connections.
  - Aggressively always looking to close connections or sockets when it does not need them.
  - Servers may have a timer and after a period of inactivity on the connection, the server will just close the connection

Heartbleed

Implementation of SSL/TLS

- The client will combat implementation choices with a 1 kb message or 'heartbeat' to say im still active.
- Heartbeat can be anything up to 64 kb of data and the server will respond to the client with the same packet size.
  - Send 1kb, get 1kb back, client sends 64kb then 64kb is returned.
- Heartbleed issue becomes a problem when you send 1kb message to the server, but tell the server it is effectively 64 kb.

Heartbleed

Implementation of SSL/TLS

- Server will return the 1kb sent but then pad that packet with data.
  - Pad with data from Random Access Memory (RAM).
- The padding could just be garbage, but it could also be valuable such as usernames, passwords or other useful data.
- The server could potentially *bleed* a lot as you can keep sending heartbeats and keep getting back data.

Heartbleed

Implementation of Secure Socket Layer (SSL)

- The issue is not with the Secure Socket Layer (SSL) protocol but the implementation of the protocol, specifically OPENSSL.
- This would not be a particular concern if OPENSSL was not in use, but there is a concern as multiple systems use OPENSSL.

Heartbleed

Legacy Systems