

## **Paper Review Report**

**Title:** End-to-End Arguments in System Design

**Authors:** J. H. SALTZER, D. P. REED, and D. D. CLARK, Massachusetts Institute of Technology Laboratory for Computer Science

The authors proposed an end-to-end placement argument as a design principle that guides the placement of functions among the modules of a computer system. End-to-end functionality that puts the functionality at the application level focuses on making underlying systems more flexible for the applications that run on them. The problem that the authors are trying to solve is where to implement and put functions in designing systems. i.e. Creating a design principle, that would generate system designs by placing functionality at the most appropriate level, to improve system performance. These functions can be put in any level between the highest and the lowest levels though there is no clearly defined place on where to put them.

Due to the fact that functionality done at the lower level faces two issues, the authors argue that end-to-end functionality is most often the best approach in system design:

- Duplication of lower level functionality at the application level (Lower level is not aware of the needs of the application level). i.e. Lower level functionalities are insufficient.
- Applications in the same lower level pay the cost for the lower-level functionality even if they are not using it

Secure data transmission (i.e. low-level encryption) is one of the examples that the authors use to support their idea. Secure management of encryption/decryption keys, vulnerability and authenticity are some of the issues during data encryption and decryption process. They point out on when, how and where to use encryption and decryption activities.

Delivery guarantees (acknowledgements), Reliable file transfer, Duplicate message suppression, Guaranteeing FIFO are also other examples provided by the authors.

Even though the paper successfully argues why much functionality should be outside of the lowest levels, it does not deeply talk about the current technologies: layering paradigm. Should applications contain the functionality? Libraries that those applications use? The operating system at the API level? The kernel level? Those questions remain relevant in system design and still the idea can be applied to a variety of different technologies and need to be taken into account in the future.

**Title: The design Philosophy of the DARPA Internet Protocols**

**Authors:** David D. Clark\* Massachusetts Institute of Technology Laboratory for Computer Science Cambridge, MA. 02139

For decades, Internet has revolutionized the computer and communications world like nothing before. The invention of telegraph, telephone, radio and computer set the stage for this unprecedented integration of capabilities. The Internet protocols, such as TCP/IP, are special set of rules that communicating devices use when they communicate. For the last fifteen years many improvements have been done on the way these protocols work nothing has been done on why protocols are designed in this way is ignored. In this paper the the author answers this question by listing the desired goals of original DARPA Internet in order of importance. They also addresses the motivations for the Internet protocols in early time and the reasoning behind the design decisions.

The paper discusses about the key designing features and what to consider. For example, the reason that packet switching was accepted instead of circuit switching. in the Internet architecture.

The paper also discusses about the key implementation goals and their rank in the order of importance; It clearly discusses about how the different protocols were created and the reason why they were created in this way not the others.

Some limitations of the Internet architecture such as: cost effective, resource management, accountability, the extremely challenging relationship between architecture and performance are discussed in the paper.

Impact of network layering and its relations with protocols is not addressed in this paper. Detail design features of TCP/IP is not described.