CSC6730 – ADVANCED NETWORKING

REVIEW OF CRITICAL REVIEWS AND DESIGN OF DARPA INTERNET

Susmit Shannigrahi
Computer Science
sshannigrahi@tntech.edu

Tennessee TECH

Discussions from previous class - logistics

- Can not download papers
  - Google the paper title – I have checked all of them are publicly available.
  - If you still can’t find me, email me.

- Skipped Slides
  - No exam, so don’t worry about those. If you need clarification, ask.

Discussions from your feedback – technical questions

- Are there some instances where at a big picture level there are net gains from having a functionality at a certain layer that might share resources with another function, but at an individual level there is a loss?
  - Yes! In-network caching vs application layer caching
  - Application layer caching
    - faster for YOUR subsequent requests
    - Does not help others
  - In-network caching
    - Slightly slower for your requests
    - Does help the others

Discussions from your feedback – technical questions

- The 16 bit checksum possibly needing to be larger. Is it not possible to calculate the probability of the checksums overlapping from an error and the cost of having a larger checksum to determine the net worth? Do we need proof to move forward?

- We already know the error probability – 1 in 65536 (2^16)
- Checksums are tailored to particular errors
  - Currently, they are tuned for burst errors
  - Are the burst errors from the ‘80s still relevant?

- See more here
  - “Huge Data is outgrowing the Internet’s file transfer protocols” by Craig Partridge
  - “https://tigerprints.clemson.edu/cgi/viewcontent.cgi?article=1075&context=hugedata”

Discussions from Reviews

- Overall observations – mostly good reviews
  - In the summary section, DON’T summarize the paper.
  - Summarize YOUR UNDERSTANDING of the paper.
- Criticizing papers
  - Think of a problem you are working on. If you apply the technique/method to your problem, does that benefit you?
- Future research directions
  - This needs some work
  - Use your imagination!

Discussions from Reviews – A Great Summary!

1. Summary of the Paper

End-To-End Arguments in System Design [1] lays out a straightforward argument for placing responsibility on end system applications for ensuring their desired application-level protocol guarantees. End To End refers to two systems connected through the internet or other series of networks that have some sort of application running on them. The paper lays out 4 possible paths to meet ensuring system design properties, which I like to think of as 'guarantees':

1. Only client applications provide the guarantee
2. Only the underlying infrastructure provides the guarantee
3. Both client and infrastructure provide the guarantees, but independently of each other
4. The client and infrstructure work together to provide the capability guarantee

Saltzer et al argue that reliability and assurance guarantees must *at least* be enforced by the two end systems, or option 1. Because of that, it is more expensive and complex to also implement the same guarantees in lower layers unless there is some sort of extra consideration (Such as errors that are caught and retransmitted at lower layers may prevent higher level, larger, and more expensive retransmissions). The main claim of the paper is that is not feasible for many guarantees to be provided solely by the interconnecting infrastructure, especially when knowledge and help of the application layer is required.

Discussions from Review – Critical Review

Today’s Internet is no longer end-to-end transparent. The backbone network is no longer limited to purely forwarding packets. The current Internet environment is dangerous, and the Internet we all rely on faces many significant challenges. The more we rely on the Internet, the more dangerous we will find, and the end-to-end argument of the Internet must be focused on solving the current challenges, and some changes need to be made in time.

-Improve the endpoint to meet can be part of the demand. Such as spam filtering can be handled at both ends of the server-side or client-side, malicious programs can be controlled through the client‘s firewall, etc.

-Arrange some proprietary services in the core of the network, and transfer some software involving security needs (such as PayPal, Gmail, etc.) to be carried out in that service, combined with the current end-to-end communication. And other software without similar needs remains in the end-to-end communication.

-Legal constraints to improve self-restraint. Fulfilling your responsibilities will reduce the appearance of malicious messages.

-Conditional trust. No longer trusting users unconditionally, the network itself needs to implement security features.

-Openness, but the need to increase the control of network services.

The end-to-end argument of the Internet is gradually transitioning to a set of mutually cooperative principles - the Internet is no longer a blanket end-to-end argument of best-effort services regardless of user behavior, and a new network-centric model can be gradually established in the middle of the network. If the bottom layer is designed to provide more functionality than is necessary for the core business, as in the hourglass model, and if the layer cannot be "completely and correctlyïmplemented, then consider designing for such functionality in other layers. The most important benefit of end-to-end arguments is that they preserve the flexibility and versatility of the Internet, allowing applications to be added without requiring changes to the backbone network deployment.

The success of the Internet is a direct result of its core design philosophy, so changes to the core design philosophy must be made very carefully and only minor changes can be made while keeping the Internet alive and active. As for whether the end-to-end argument will survive, it is too early to make predictions. All we can do now is drive some specific results. The flexibility and openness promoted by the end-to-end argument will continue to be important in guiding future network development, and this flexibility should be preserved. It is unlikely that future network development will be limited to the architectural ideas of the end-to-end argument, but it is also unlikely that such doctrinal guidance will be completely abandoned.

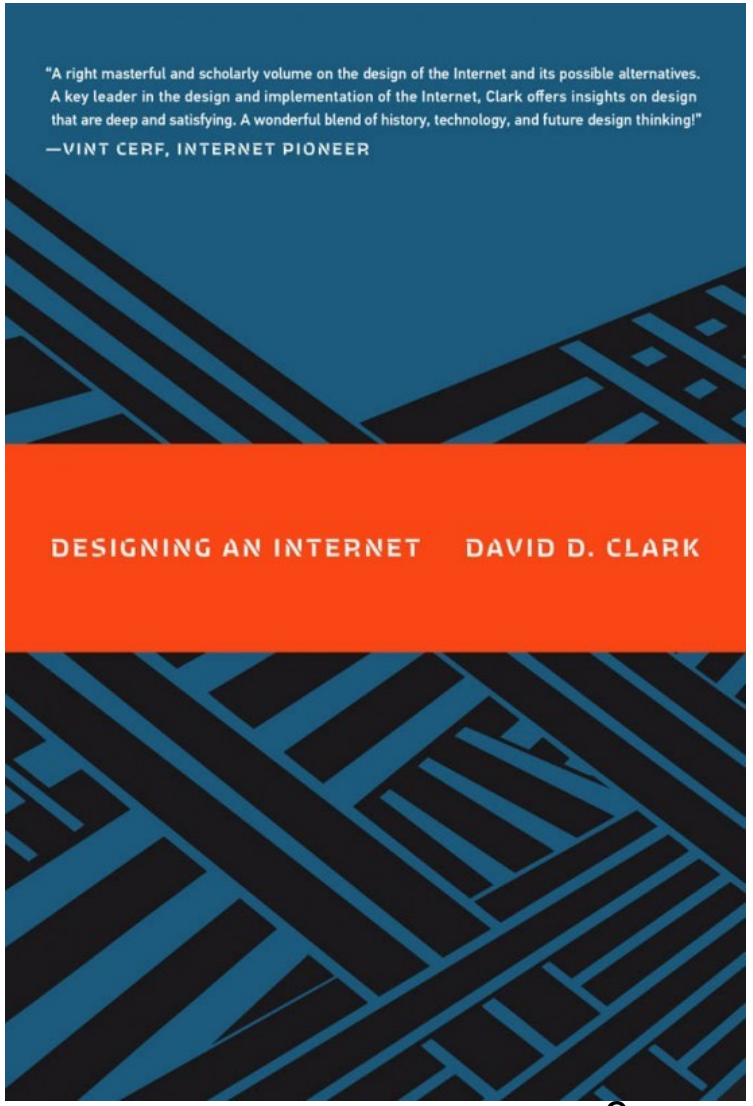
Discussions from Reviews – Critical Review

The paper is outdated. Priority has shifted from the efficiency of the program to the efficiency of the programmer and maintainability and reliability of the program. This shift has made the arguments made in this paper unconvincing. Having the functionality in the networking layer allows for reusability and reliability of the features. It does highlight the appreciation we should have for the tool sets we have as developers. There are instances where an end-to-end check will still need to be performed to ensure data transferred completely successfully or the full action was taken by the remote host, but the network layer already offered is efficient enough and helps facilitate the program.

The Design of the DARPA Internet Protocol

- David D. Clark - MIT
  - One of the most influential protocol architect for the Internet
  - Continues to be a visionary today

- If you enjoy reading the history of the Internet, read his book
  - Designing an Internet – MIT Press 2018



Main Focus

- Why the Internet was designed this way?
  - Captures some of the reasoning
  - Connectionless
  - Packet switching
  - TCP over IP
- Why do we care?
  - Still need to connect different networks!

Fundamental Goal 0 – Connect the networks

- “Develop an effective technique for multiplexed utilization of existing interconnected networks”
  - Interconnect networks – backward compatibility
  - Keep the administrative control/boundaries
  - Packet switched networks since they fit the applications
  - Store and forward packets - the concept of gateways

Secondary Goals 1-7

- **Goal 1**: Internet communication must continue despite loss of networks or gateways.
- **Goal 2**: The Internet must support multiple types of communication service.
- **Goal 3**: The Internet architecture must accommodate a variety of networks.
- **Goal 4**: The Internet architecture must permit distributed management of its resources.
- **Goal 5**: The Internet architecture must be cost effective.
- **Goal 6**: The Internet architecture must permit host attachment with a low level of effort.
- **Goal 7**: The resources used in the internet architecture must be accountable.

Goal 1 – Network failure should not affect the applications

- **Goal 1**: Internet communication must continue despite loss of networks or gateways. The failure should be transparent to the applications.

  - **Fate sharing** – if your endpoints fail, there is no harm is losing the state. Otherwise, the applications should perform as usual. Example?
  - Stateless packet switching → Datagrams

- Does the Internet still support this goal?

Goal 2

- **Goal 2**: The Internet must support multiple types of communication service.

- **How do we accomplish this?**
  - Example application - Remote login
  - TCP – reliable sequenced data delivery
  - UDP – basic datagram service – a debugger should not be reliable, real time speech
  - TCP/IP was a single layer, what was then broken into two to achieve goal #2

Goal 3

- **Goal 3**: The Internet architecture must accommodate a variety of networks.
  - A variety of networks – long haul, local area, satellite, and more
  - Is this still true?

  - Minimum set of assumptions

  - Job of the transport layer to provide additional services – TCP or UDP
    - Not the job of IP
    - End-to-end – only that your endpoint here is the transport layer

Secondary Goals 4-7

- **Goal 4**: The Internet architecture must permit distributed management of its resources.
  - Tools for distributed management
  - Still does not exist
  - Is this still necessary?

- **Goal 5**: The Internet architecture must be cost effective.
  - Not always money – also overhead
  - How many bytes can you fit in?
  - Retransmission? At which layer?

- **Goal 6**: The Internet architecture must permit host attachment with a low level of effort.
  - Somehow easy but not too difficult once you know how

- **Goal 7**: The resources used in the internet architecture must be accountable.
  - Does it exist?

Conclusions

- The Internet goals are mostly successful
- Datagram works for 1st three goals, not for the rest
- Maybe there is a better building block for the next architecture?
  - How about content names?

Deliverable for the next class

- Fill out the survey - https://tntech.co1.qualtrics.com/jfe/preview/SV_bjT8BpnRABUxZ7o

- **Critical review due on 09/08**
  - P. Mockapetris, K. Dunlap, “Development of Domain Name System”, ACM SIGCOMM Computer Communication Review, 1988

- **Presentation – Max Layer**
  - “Is Internet routing undermining DNS anycast benefits?"