School of Engineering & Applied Science
THE GEORGE WASHINGTON UNIVERSITY

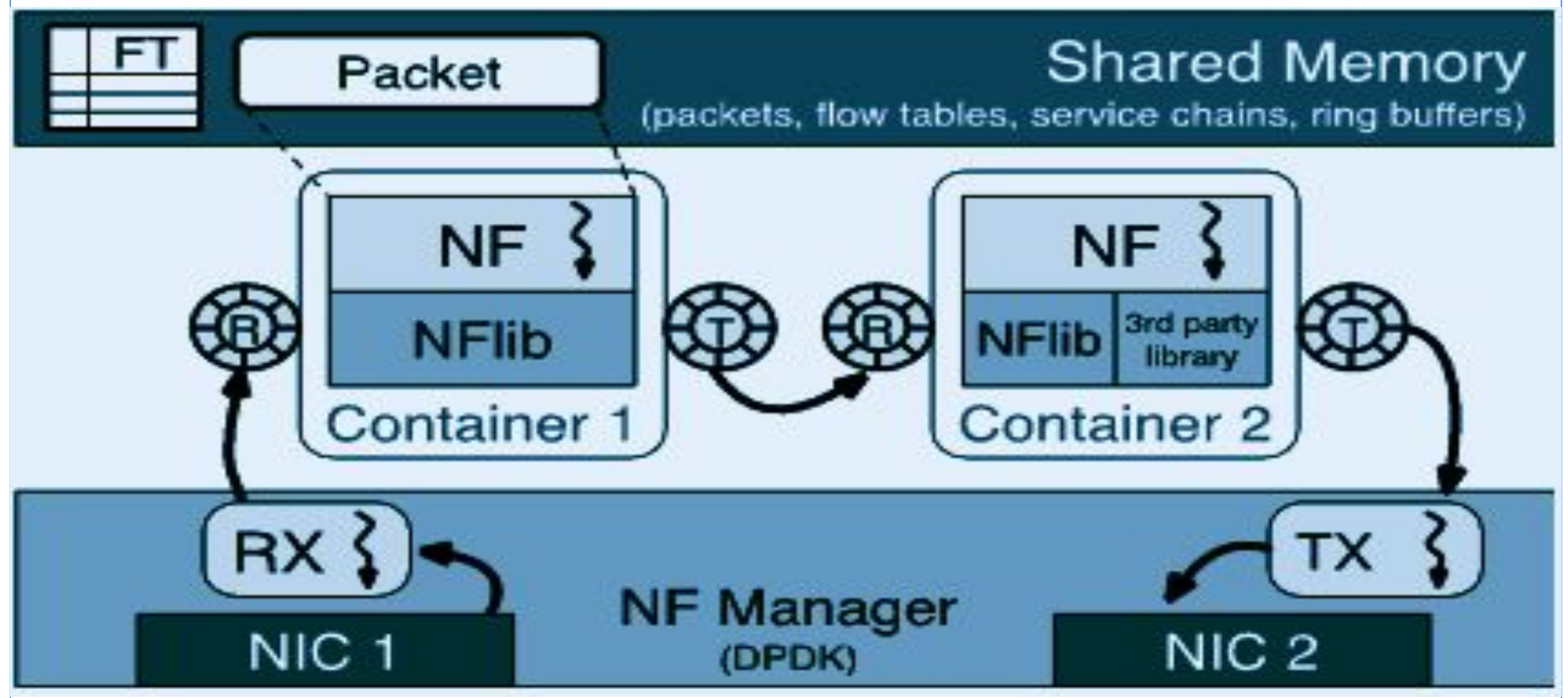
Secure and Scalable Network Packet Processing

Dennis Afanasev, Kevin Deems, Prof. Timothy Wood, Advisor
The George Washington University Computer Science

openNetVM

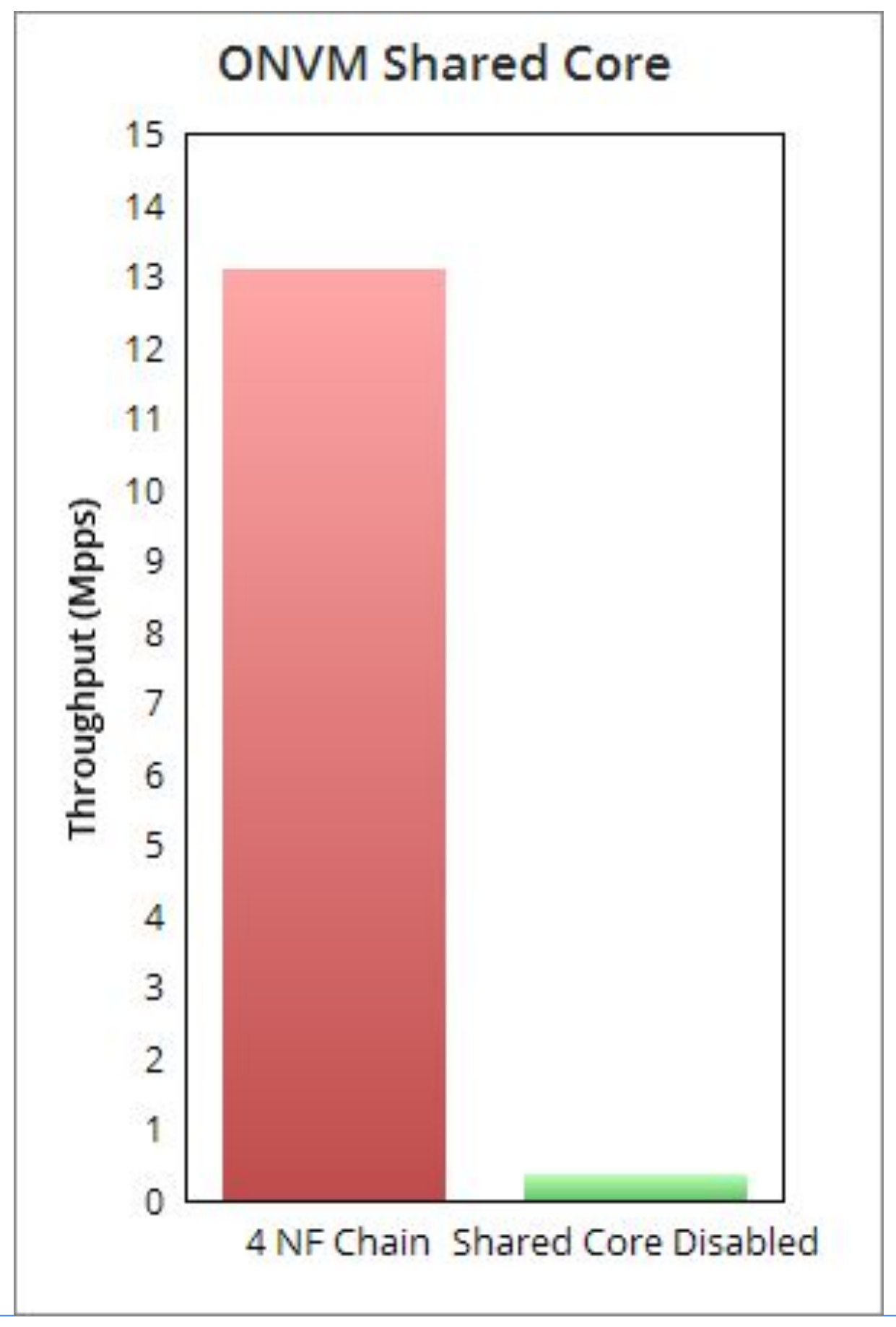
Introduction

- OpenNetVM is a software-based NFV platform for scalable and flexible network computing
- Routers, firewalls or intrusion detection systems, implemented in kernel space perform sub-optimally for variable network loads



Recent Improvements

- Shared core execution of network functions is one of the latest major improvements to the platform
- NFs can be put to sleep when they don't have packets, so that many child NFs can be created to split up the workload over the whole system



Secure TCP Processing Framework

- mTCP can be combined with openNetVM to provide TCP service capabilities
- Normally, virtualized servers use a shared memory pool for data structures, packets, and files to optimize performance
- This leads to fast throughput but decreases level of security
- Example: HTTP WebServer running as network function on OpenNetVM
    - Shared memory pool between all clients
    - All HTTP request parsing is done in the same process

- Proposal: Framework that isolates connection-based network functions from each other while maintaining high throughput

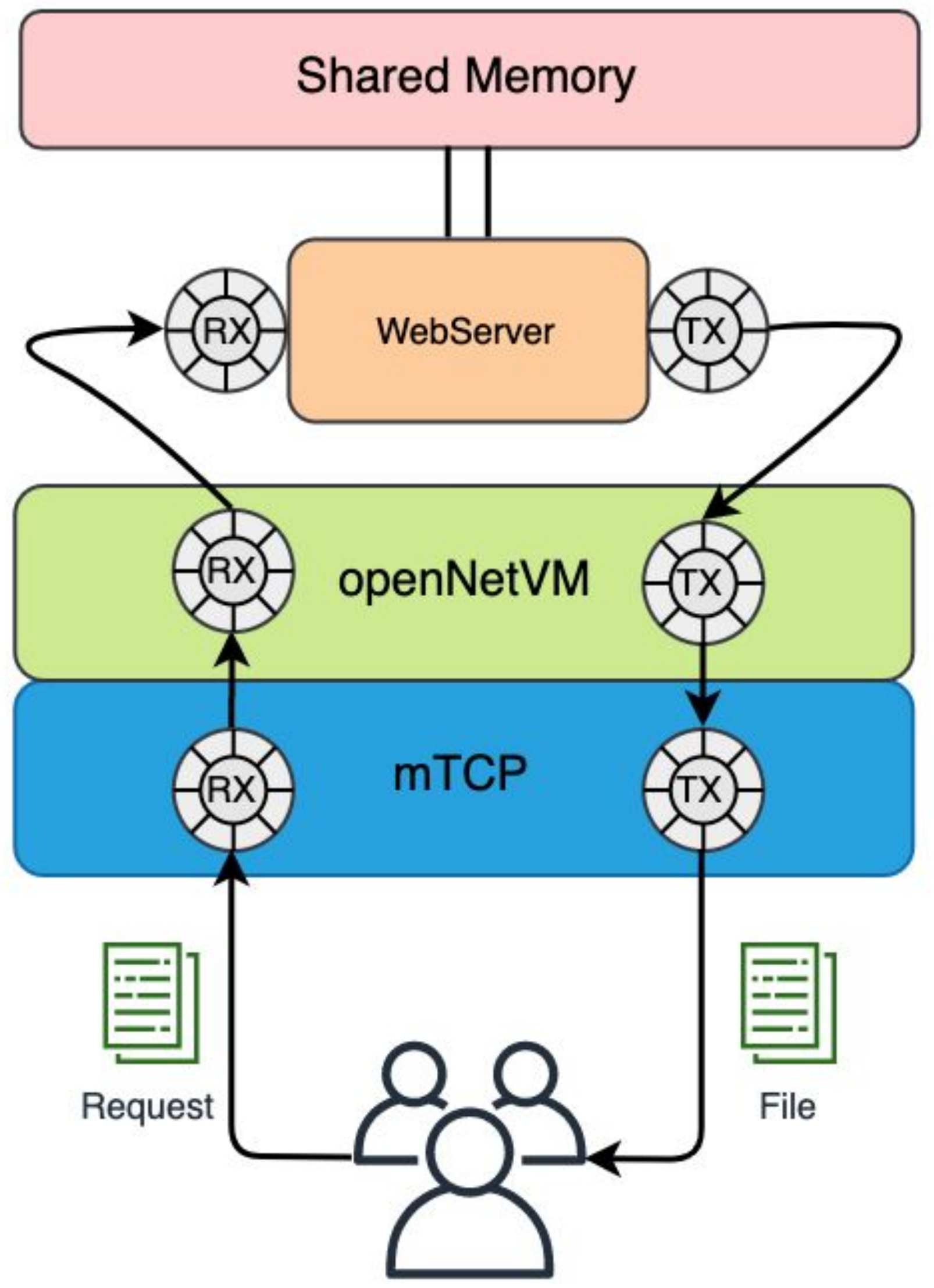


Figure 1: Simple HTTP web server implemented using openNetVM and mTCP. The web server is implemented as a network function that processes GET requests.

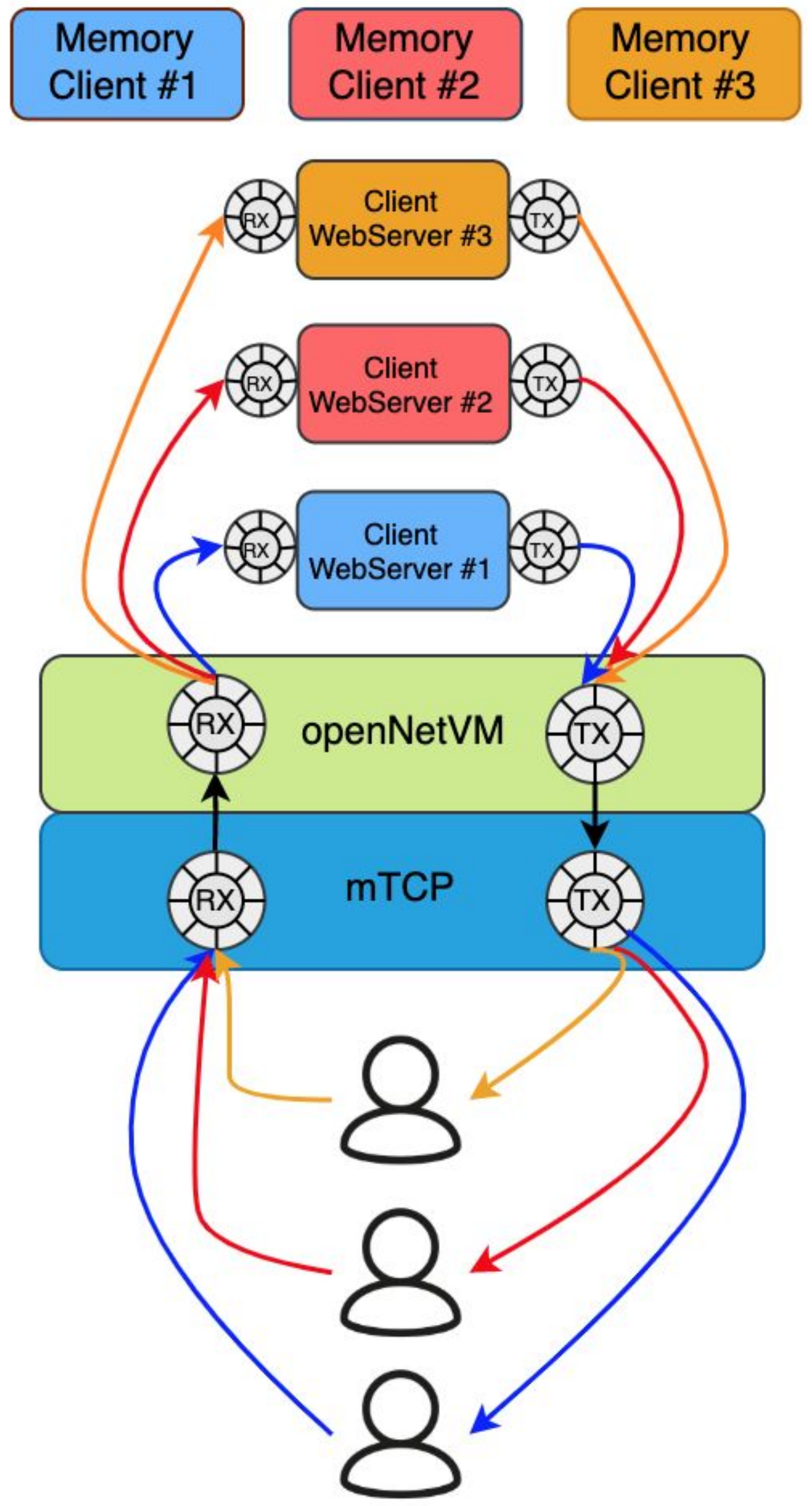
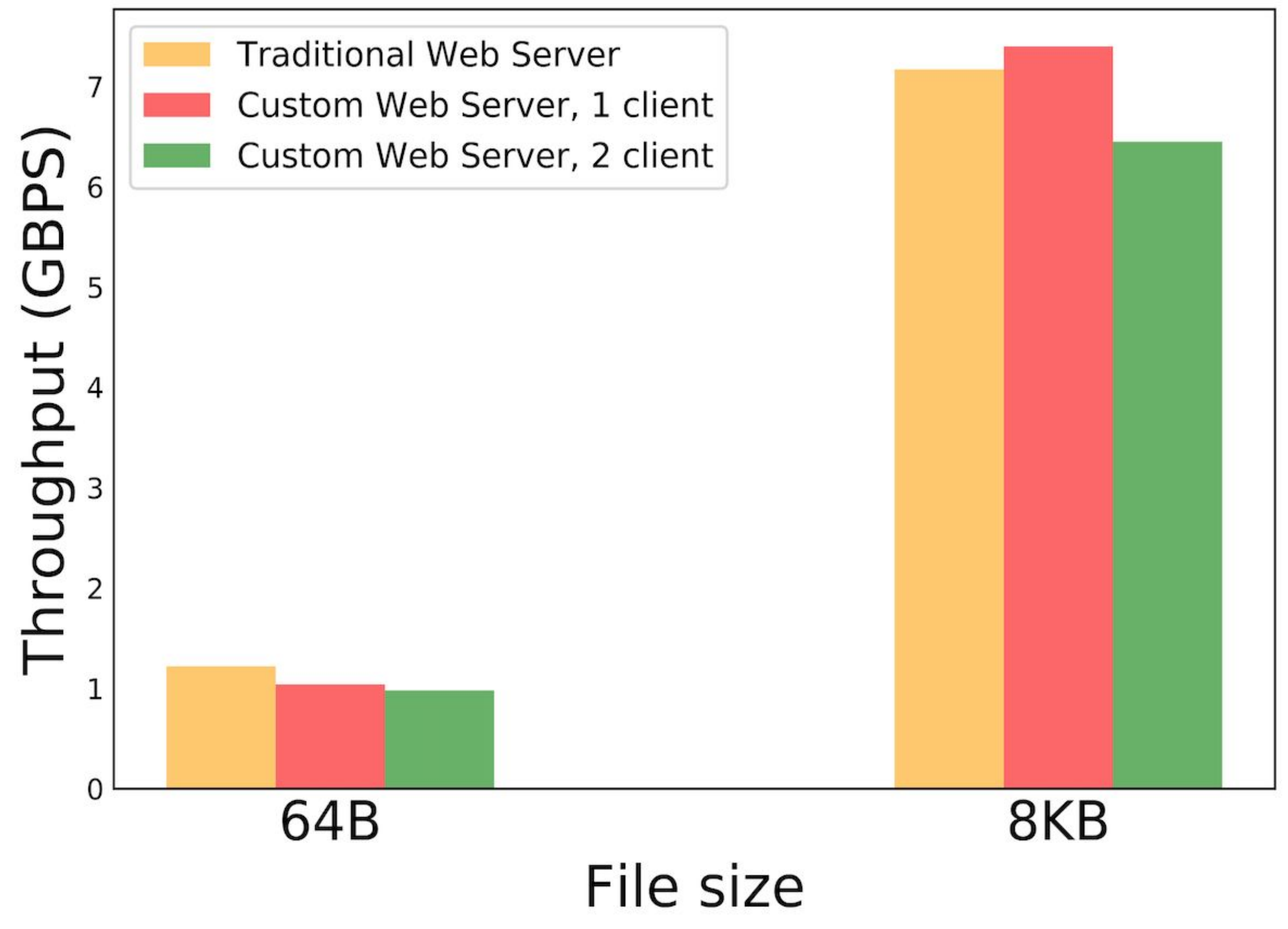


Figure 2: HTTP web servers implemented as isolated network functions serving individual clients. Security is increased as GET request parsing is implemented on seperate processes.

Results

- Performance with small file downloads using proposed architecture is closely maintained with traditional architecture
- Increase in performance with larger sized files using custom architecture



Conclusion

Future research

- ONVM's flexible NF architecture allows for integrations with other platforms, to deliver scalability and communication
- Using new openNetVM features combined with a virtual TCP stack, we can effectively serve multiple clients while preserving security and maintaining high throughput
- Apply custom architecture to different connection based services, such as Redis
- Provide dynamic NF chains, scaling, and lifecycle management from web interfaces