Stream Splitting
Moving Target Defense

Progress Report Presentation

Objective

Provide moving target defense against man-in-the-middle attacks on communication over the Internet with stream splitting

Team:
Utsav Bhatt
Sri Harsha Lenka
Lauren Murphy
David Stenson

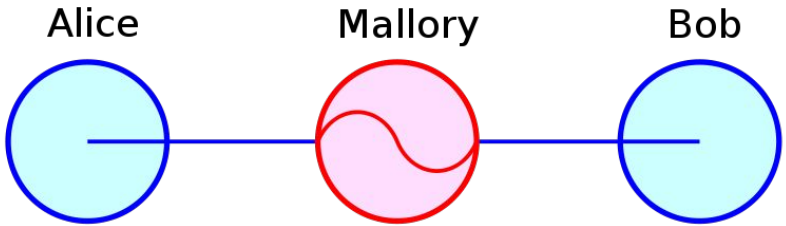
Argonne NATIONAL LABORATORY

Sponsor:
Joshua Lyle

Concepts

Man-in-the-middle Attack on the Internet

- Adversary sits between two parties and passes along their communication
  - Their objective is to inspect or alter the communication
- Methods for maintaining confidentiality and integrity are already implemented in several protocols
  - E.g. authenticated encryption, etc.
  - Increase difficulty of deciphering or tampering with it
- Hosts can’t stop communication from going *through* untrusted nodes entirely

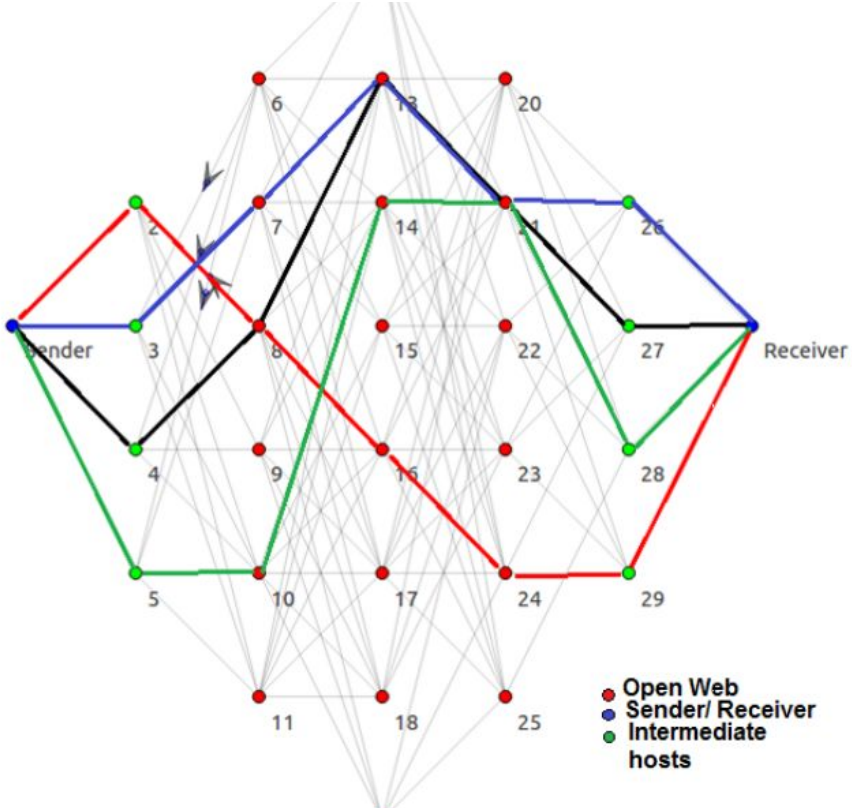


Stream Splitting as a Moving Target Defense

- Split one stream of data into several at the source and send them simultaneously, then receive and reassemble them at the destination
- Routing each stream over geographically and/or physically different paths from other streams increases difficulty of capturing entire transmission
  - Adversaries must control one node on each path
  - They also have to resequence the data
- Changing the paths results in
  - Increased difficulty of controlling nodes on any of the paths
    - Selecting diverse paths
  - Increased efficiency
    - Balancing load on each path according to bandwidth

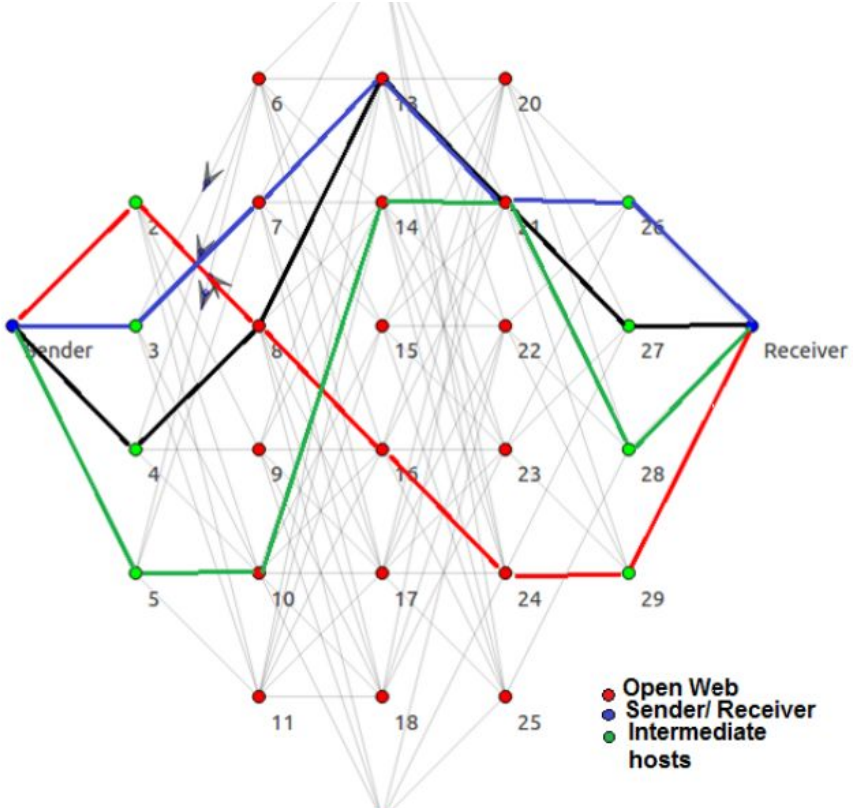
Path Diversity

- We can enforce diversity by deploying **entry / exit nodes** to relay the streams
  - In different geographic locations or on different physical media
- Each host forwards one stream
  - Source to entry
  - Entry to exit
  - Exit to destination



Diversity Quotient

- Source host determines each possible path and selects a subset of paths and load for each according to diversity quotient, Q
  - It increases efficiency to route over the one or a few routes
    - Exploit best bandwidth
  - However, by putting more data on one or a few routes, we decrease path diversity
- Tradeoff between efficiency and confidentiality decided by user



Previous Work

Moving Target Defense

- Making attack surface of system dynamic by changing aspects of system
  - Forces attackers to expend more time and energy on reconnaissance
  - Minimizes usefulness of exploits
- Focus on security
- Multiple Operating System Rotational Environment (MORE)
  - Rotate the operating system running on the host
- Dynamic Application Rotation Environment (DARE)
  - Rotate the platform (Nginx, Apache, etc.)
- These strategies are focused on protecting hosts from intrusion, rather than preserving confidentiality of communication between hosts

Multipath Transport Layer Protocols

- Focus on efficiency
- Multipath TCP (MPTCP)
  - Transmits data over multiple TCP connections
    - If one connection drops, then transmission continues on other connections
  - Maintain connections when switching between networks, i.e. from WiFi to mobile, etc.
  - Apple uses MPTCP for Siri
- Stream Transfer Control Protocol (SCTP)
  - Made for multihomed devices
  - First, they exchange lists of their reachable IP addresses (on different NICs)
  - Next, they select one pair of the possible to communicate with
  - When the current path between two IP addresses fails, the devices switch to a backup pair to utilize an alternative path

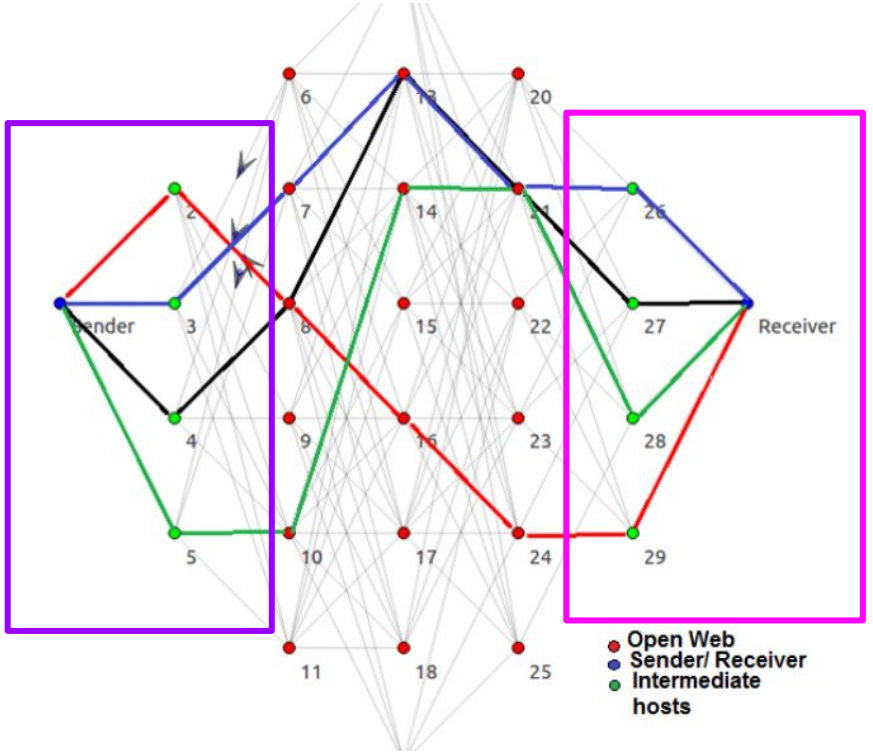
Concurrent Multipath SCTP (CMT-SCTP) VS MPTCP

- CMT-SCTP is a variant of SCTP that utilizes multiple paths at the same time
  - Instead of selecting one and failing back onto others
  - Load balancing over each path
- We chose CMT-SCTP over MPTCP to utilize / modify
  - MPTCP has overhead by maintaining multiple TCP connections
  - CMT-SCTP is geared towards multihomed devices
  - SCTP offers better congestion control and fault detection
  - Chunks in SCTP lend themselves to stream splitting better than streams of MPTCP

Approach

Modifying CMT-SCTP

- CMT-SCTP defines path between two network interfaces, one on each host
- Instead splitting CMT-SCTP streams over network interfaces on the host, we treat entry and exit nodes as network interfaces and form paths between them
  - Source and entry nodes are symbolic client
  - Destination and exit nodes are symbolic server



Plan

Environment

Deploy Common Open Research Emulator (CORE) to emulate network

- Emulation allows us to quickly
  - Add / drop nodes / links
  - Configure services on each node
  - Adjust link bandwidth
  - Scale

Stream Splitting, Load Balancing

Select an implementation of CMT-SCTP to modify to support stream splitting

- We chose a userspace implementation by Tuxen et al.
  - Portable
  - Same throughput and latency as FreeBSD kernel implementation
  - Modifiable as open source

Modify implementation to support load balancing

- Check capacity of node and latency of all paths to select subset of viable
- Balance load across viable paths
  - Maintain geographic diversity and efficiency according to user-defined quotient

Resilience

Test resilience of implementation and implement appropriate safeguards

- Network degradation
  - Drop nodes and decrease bandwidth
  - Track path diversity and efficiency
- Source check
  - Check program is verifying source
- Integrity check
  - Check program can detect streams modified in transit

Encryption

Utilize AES to double encrypt data and encrypt indices for sequencing data

- Only clients should decrypt message
  - Each client should have unique key

Amazon Web Services

- Time permitting, develop program for AWS and test it over the Internet

Progress

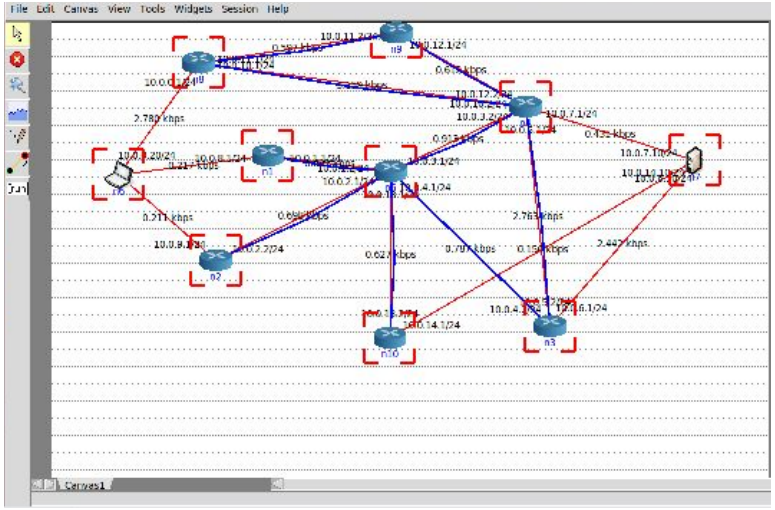
Tasks

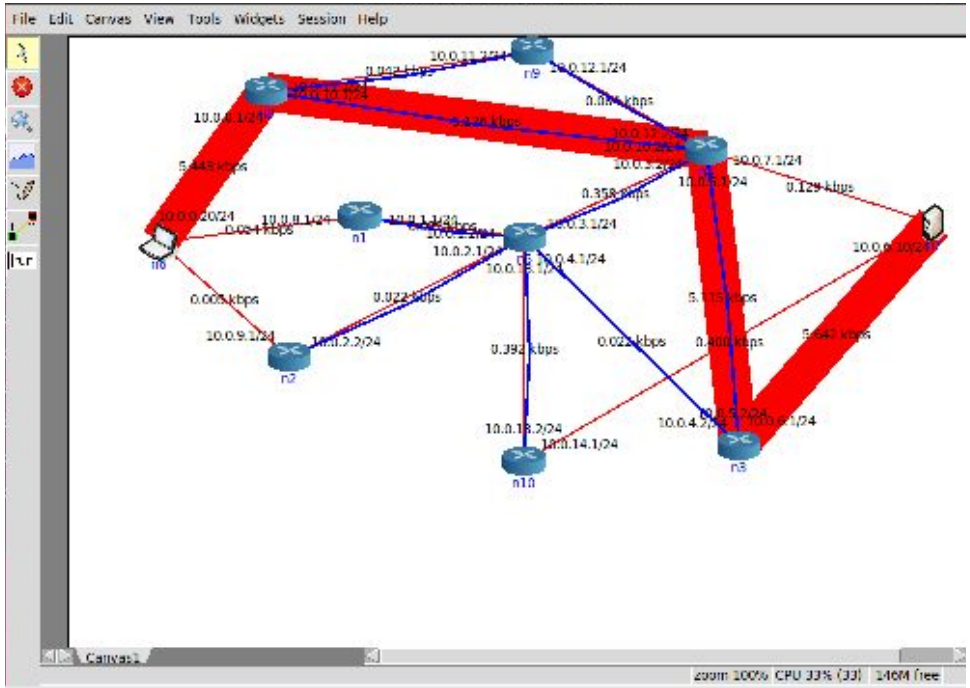
- ✓ Configure and deploy network emulator
- ✓ Select protocol, e.g. MPTCP, SCTP, etc. to utilize or modify
- ☐ Implement stream splitting
  - Thwart MITM attack
- ☐ Implement load balancing and recovery mechanisms
  - Speed
  - Resilience
- ☐ Implement encryption
  - Further confidentiality
  - Integrity

Deliverables

- ✓ Literature review
- ✓ Proposal
- ✓ Midterm report
- ✓ Midterm presentation
- ☐ Implementation
  - ☐ Documentation
- ☐ Final report
- ☐ Final presentation

Current Progress





Issues

Issues

- Efficiency
  - Our modifications to CMT-SCTP shouldn’t reduce its efficiency
- Scaling
  - Multipath protocols show promise in simulations, but not necessarily better in reality
  - Complete disjointedness of paths may not be possible
    - Attempt to detect and resolve fairness issue
- Security
  - Encryption could add overhead

Resources

- Patent on Stream Splitting Moving Target Defense
- Report on Moving Target Defenses by Argonne National Laboratory
  - MORE
  - DARE
  - Stream Splitting