# Enriching Network Security Analysis with Time Travel

Gregor Maier
gregor.maier@tu-berlin.de
TU Berlin / DT Labs

Robin Sommer
ICSI / LBNL

Holger Dreger
Siemens AG, CT

Anja Feldmann
TU Berlin / DT Labs

Vern Paxson
ICSI / UC Berkeley

Fabian Schneider
TU Berlin / DT Labs





## Motivation

- Goal:
  - Enable analysis of network activity that becomes interesting **in retrospect**
- How:
  - Archive **raw network packet** data
  - Full packets, not aggregation
- Problem:
  - Wholesale recording not feasible using commodity hardware
  - Gigabit Networks ⇨ several TB / day

# Motivation: Why?

- Network Intrusion Detection System (NIDS):
  - Suspicious activity ⇨
    Also analyze offender's traffic **from past** in-depth
  - Without archive: traffic is gone
- Forensics:
  - E.g., break-in happened days ago: How? Who?





# Motivation: Proposal

- Common practice at Lawrence Berkeley National Laboratory (LBNL):
  Bulk recording (tcpdump)
  - Omits key services (HTTP, FTP, etc.)
  - Manual analysis of traces after incident
- Our solution:
  **"Time Machine" (TM)** for "Time Travel"
  - Design driven by continuous feedback and live deployments, e.g., at LBNL

# Outline

- Time Machine Design
- Performance Evaluation
- Coupling TM with Network Intrusion Detection System (NIDS)
- Conclusion





# Time Machine Design

# Key Insight: Heavy-Tails

- Minority of connections carry most of volume
  - Bulk data transfer (Video, Audio, etc.)
- Majority of connections is small
  - 91% of connections < 10 KB
  - 94% of connections < 20 KB
- Relevant/interesting data mostly at beginning
  - Application protocol headers
  - Handshakes

[1] PAXSON, V., AND FLOYD, S. Wide-Area Traffic: The Failure of Poisson Modeling. *IEEE/ACM Transactions on Networking 3*, 3 (1995).





# TM: exploits Heavy-Tails

- **Cutoff** heuristic:

  Only store the first **N bytes** per connection

  ⇨ record most connections entirely

  ⇨ record beginning of remainder of conns,

  90% reduction in volume
- Observation:
  - After 10--20KB mostly bulk data
- Evasion risk (future work)

# TM Design

- Capture operation
  - Captures packets from network tap
  - Checks per connection cutoff and determines storage class
  - Updates packet indexes
- Query operation
  - Index lookup
  - Packet retrieval
- Storage management and bookkeeping
  - Memory and disk buffer and indexes





# Experiences → Design

- Multi-threaded design
- Most queries triggered by NIDS
  - Automated query interface
  - Feed historic data back to NIDS for analysis
- Some traffic more important than other
  - Multiple storage/traffic classes
  - Tune parameters dynamically via NIDS

# Performance Evaluation



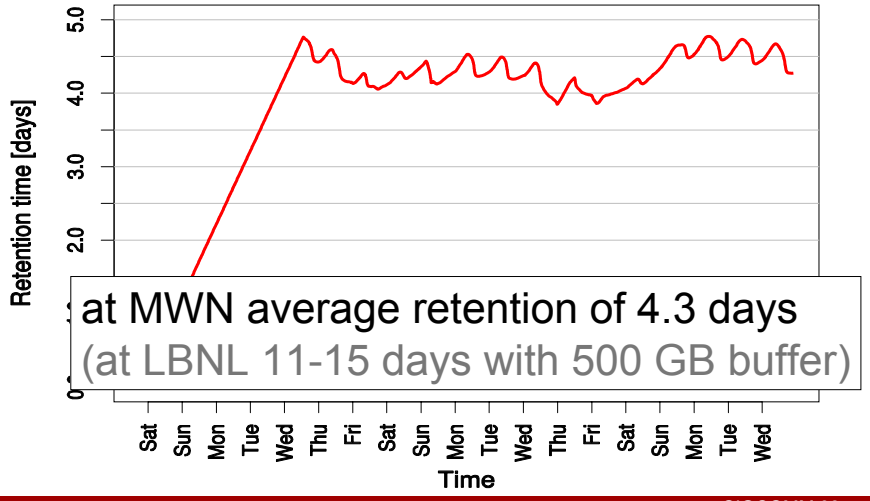


# Setup

- LBNL: Lawrence Berkeley National Laboratory
  - 10 Gbps uplink, 1-2 TB/day
  - 15 KB cutoff, 150 MB mem buffer, 500 GB disk buffer
  - Two dual-core Intel Pentium D, 3.7 GHz, Neterion NIC
- MWN: Munich Scientific Network
  - Two major universities + research institutes
  - 10 Gbps uplink, 3-6 TB/day
  - 1 Gbps monitoring port
  - 15 KB cutoff, 750 MB mem buffer, 2.1 TB disk buffer
  - Dual AMD-Opteron 1.8GHz, 4 GB RAM, Endace NIC

# Retention Time on Disk

at MWN with 2.1 TB disk buffer (Jan'08)




at MWN average retention of 4.3 days

(at LBNL 11-15 days with 500 GB buffer)

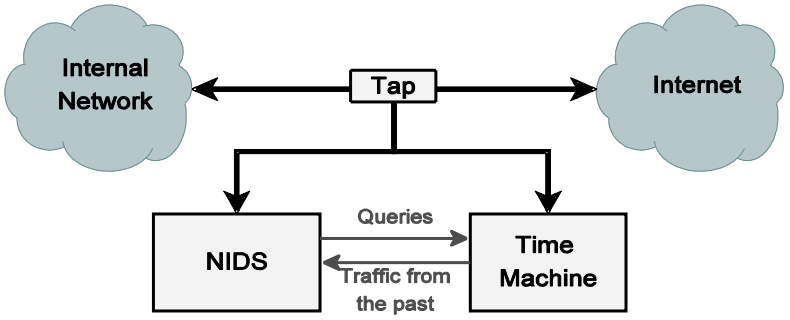




# Coupling TM with a Network Intrusion Detection System (NIDS)

# Setup

- NIDS: Open-source Bro
- Deployed at LBNL (10 Gbps site) for months
  - 15KB cutoff, 150 MB mem buffer, 500GB disk buffer








# Improved Forensics Support

- NIDS: **changes** TM's parameters **dynamically**
- Example:
  - For every NIDS reported incident: Change to more conservative storage class
    - Scanners: 50KB cutoff, 75MB mem, 50GB disk
    - Alarms: no cutoff, 75MB mem, 50GB disk
  - Results: total of 12,532 IPs in scanners, 592 in alarms

# Improved Forensics Support

- NIDS: Preserves incident related data
  - Stores in separate file
  - Not subject to TM's eviction
- Example:
  - Every major non-scan incident (alarm)
    - Store connection's packets on disk
    - Store packets of offending host (last hour)
    - TCP: NIDS reassembles application stream





# Retrospective Analysis

- NIDS: **analyses** traffic from past
- Addresses resource/analysis trade-offs
- Broadens analysis context
  - Suspicious activity
    ⇨ more expensive, in-depth analysis
- Example: HTTP
  - Only analyze requests
  - Suspicious request: retrieve reply from TM
  - 1% retrieved, **CPU util: 40% → 27%**

# Conclusion





# Conclusion

- We build and evaluated efficient Time Machine
  - Commodity hardware for gigabit environments
  - Used operationally
- Cutoff heuristic: keep first x KB of every connection
  - Reduce volume typically by more than 90%
  - Retain days / weeks of full payload traces on disk
  - Retain minutes in memory
- Coupled Time Machine with NIDS
  - Improved forensic support
  - Automatic queries for deeper inspection

# Future Work

- Mitigate evasion risk
  - Use randomized cutoff
  - Keep some packets even after cutoff hit
  - Use NIDS to disable cutoff
- Cutoff processing in hardware
  - e.g., NetFPGA (Shunt)
- Aggregation instead of direct eviction





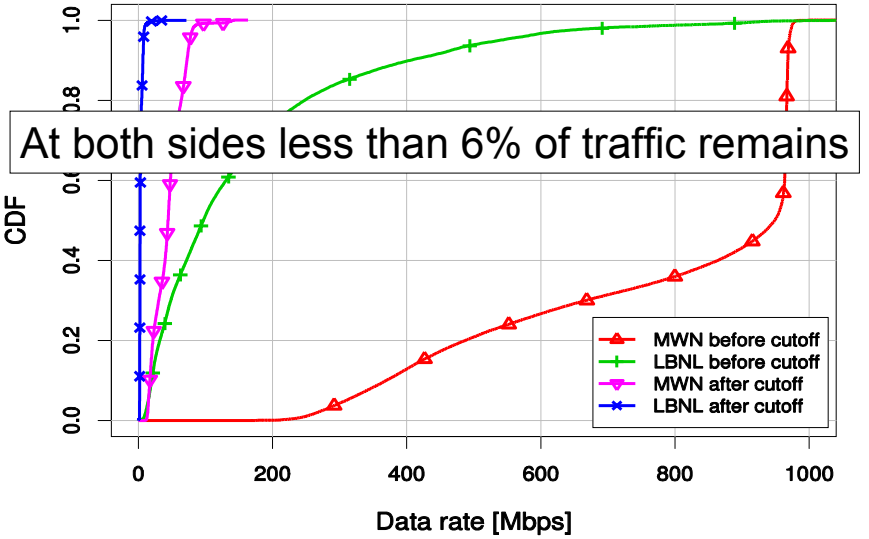
# Questions?

Get your own Time Machine:
http://www.net.t-labs.tu-berlin.de/research/tm

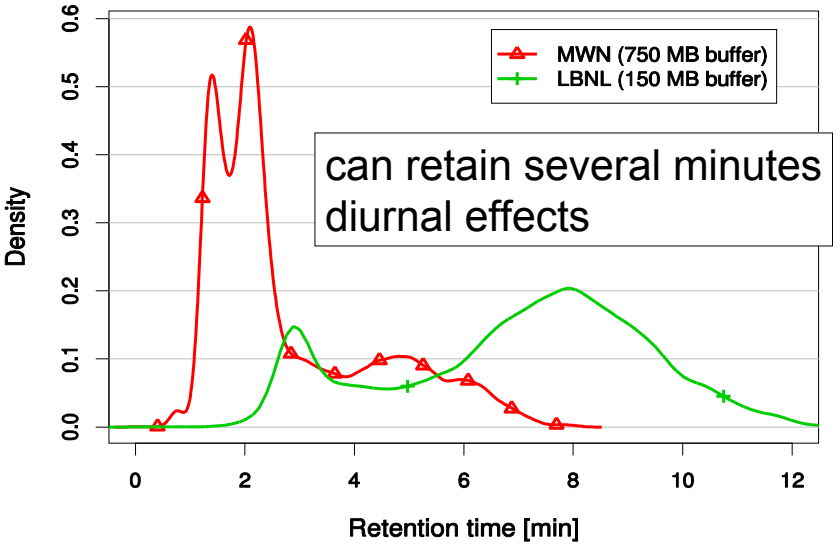
# BACKUP SLIDES



# Effectiveness of Cutoff

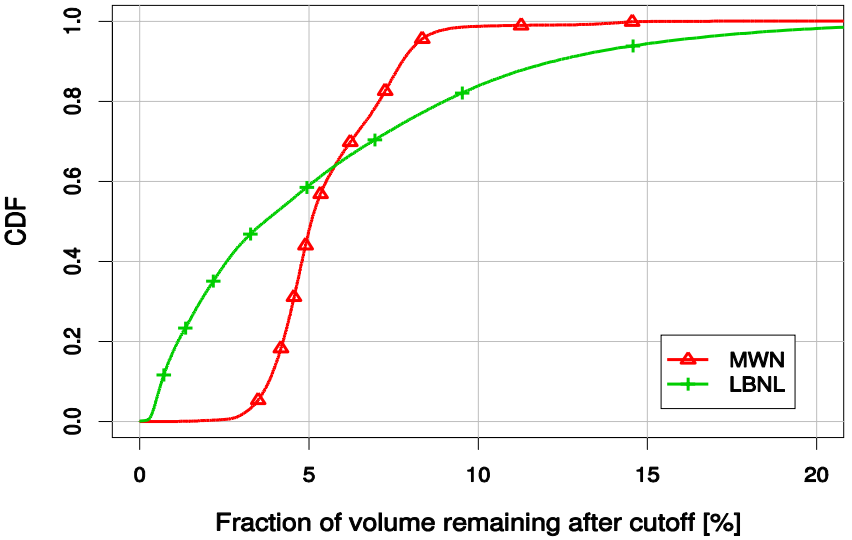


# Retention Time in Memory

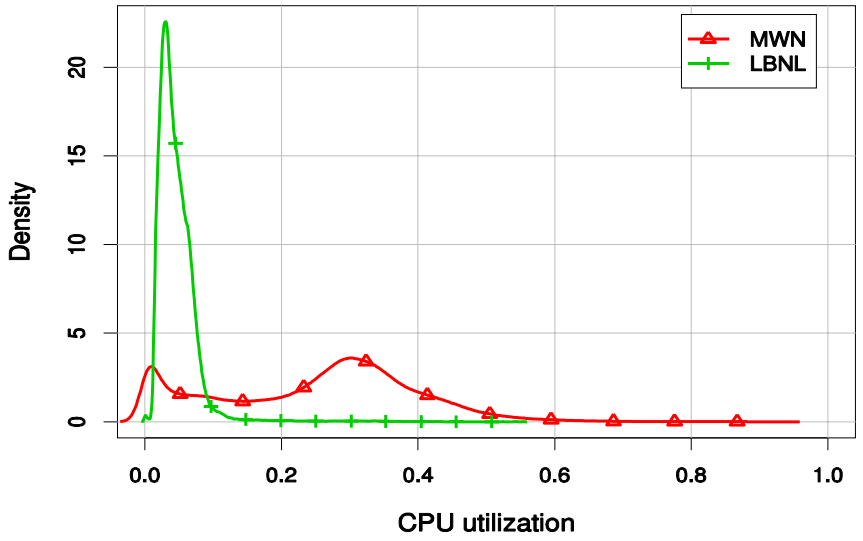





# Traffic after cutoff

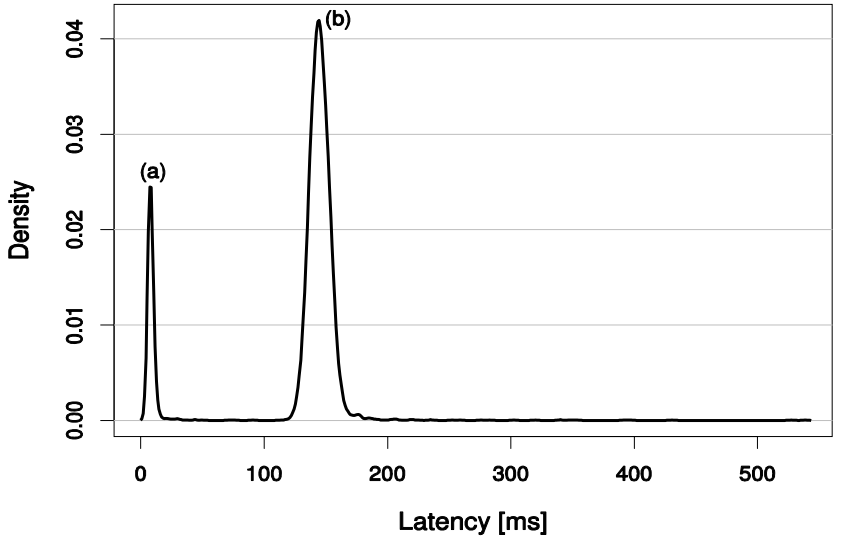


# CPU utilization

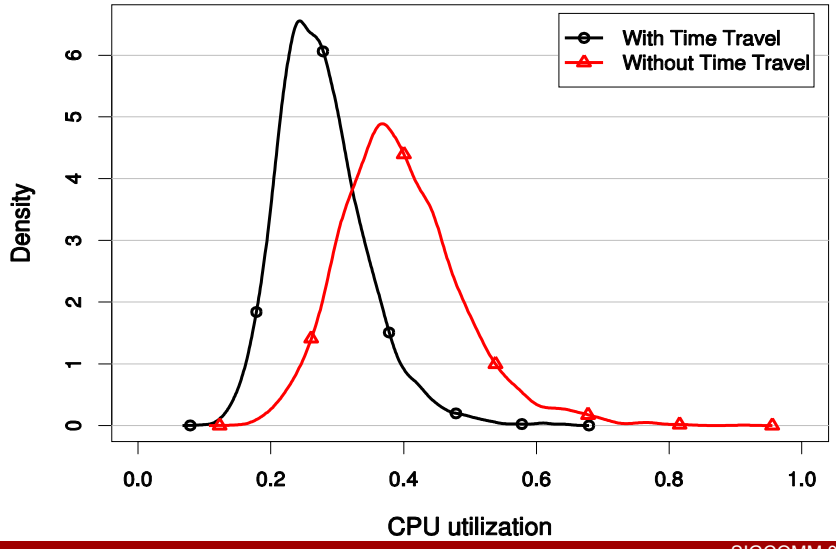





# Query performance



# HTTP Offloading






# TM Architecture

- Classification: Map packet to connection, cutoff enforcement
- Storage Class: cutoff, timeout, buffer budgets
- Index: Header tuples
- Interface:
  - Tune parameters
  - Request packets
    - To disk, to network
    - Specify scope
    - Subscriptions