AuditBox: Building An Auditable Network Service Chaining Framework

(To Appear at USENIX NSDI 2021)

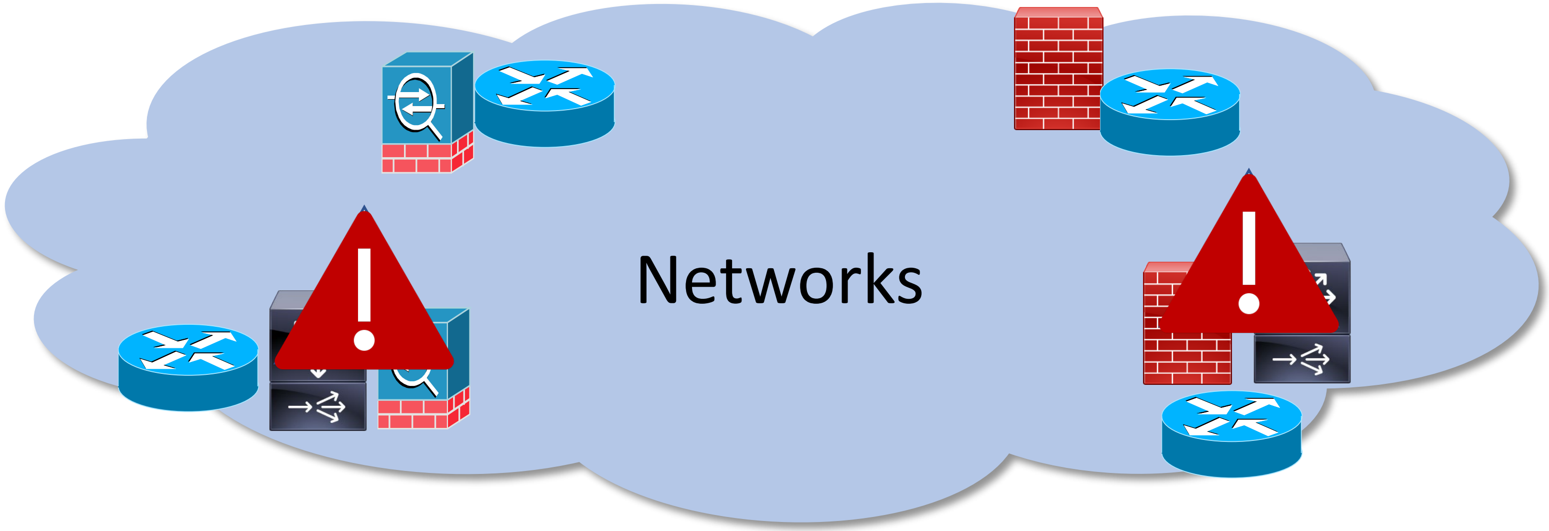
Grace Liu, Hugo Sadok, Anne Kohlbrenner,

Bryan Parno, Vyas Sekar, Justine Sherry

CyLab Carnegie Mellon University Security and Privacy Institute

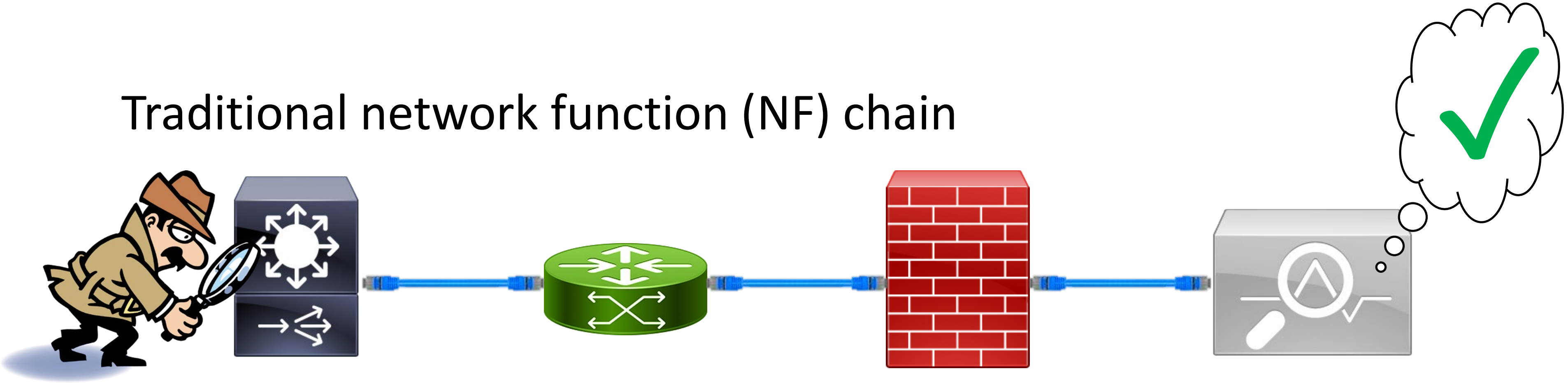
Background: Network Functions

- Network Functions (e.g., Firewalls, Proxies) are crucial for networks.
- Mandated by legal and policy requirements (e.g., HIPAA, GDPR).
- How can we ensure these NFs are operating correctly?

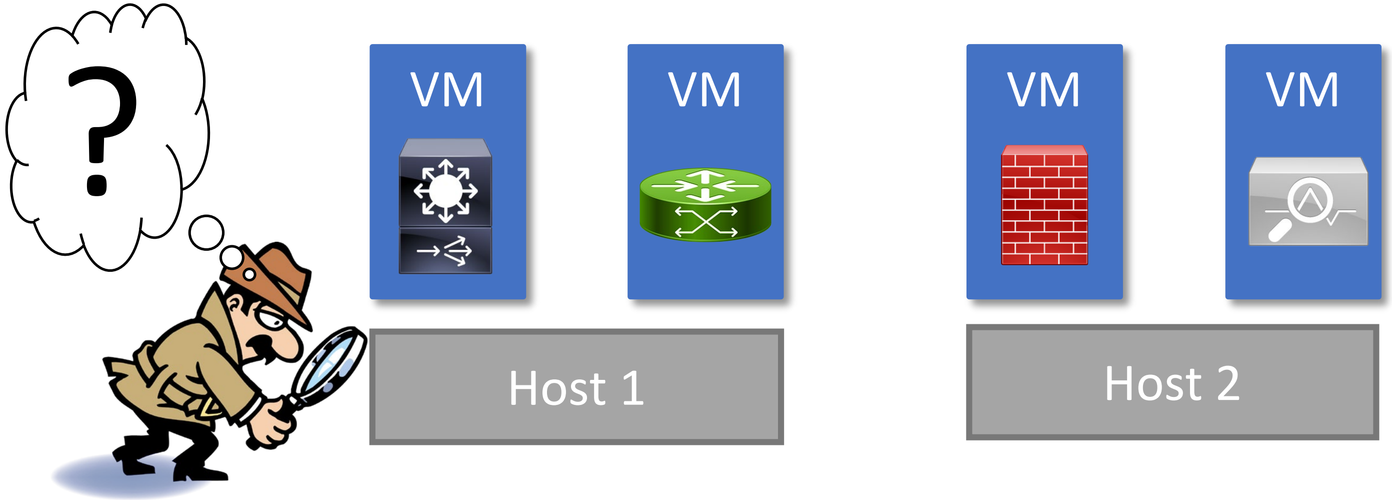


Problem: Virtualized NFs are Hard to Audit

Traditional network function (NF) chain

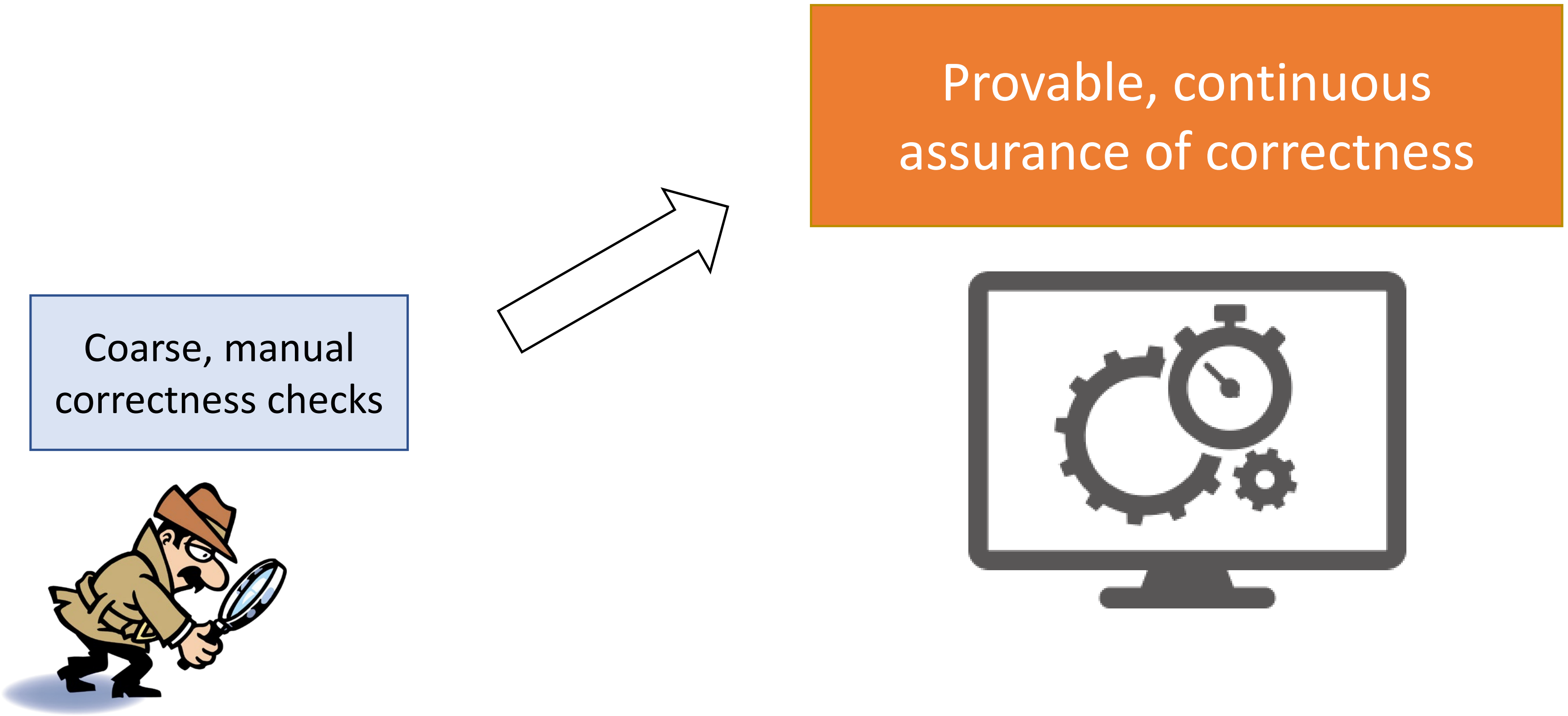


Modern virtualized NF chain



Overarching Goal

- Offer missing capabilities to audit NFV deployments



Coarse, manual correctness checks

Provable, continuous assurance of correctness

Prior Work: Verified Routing Protocols

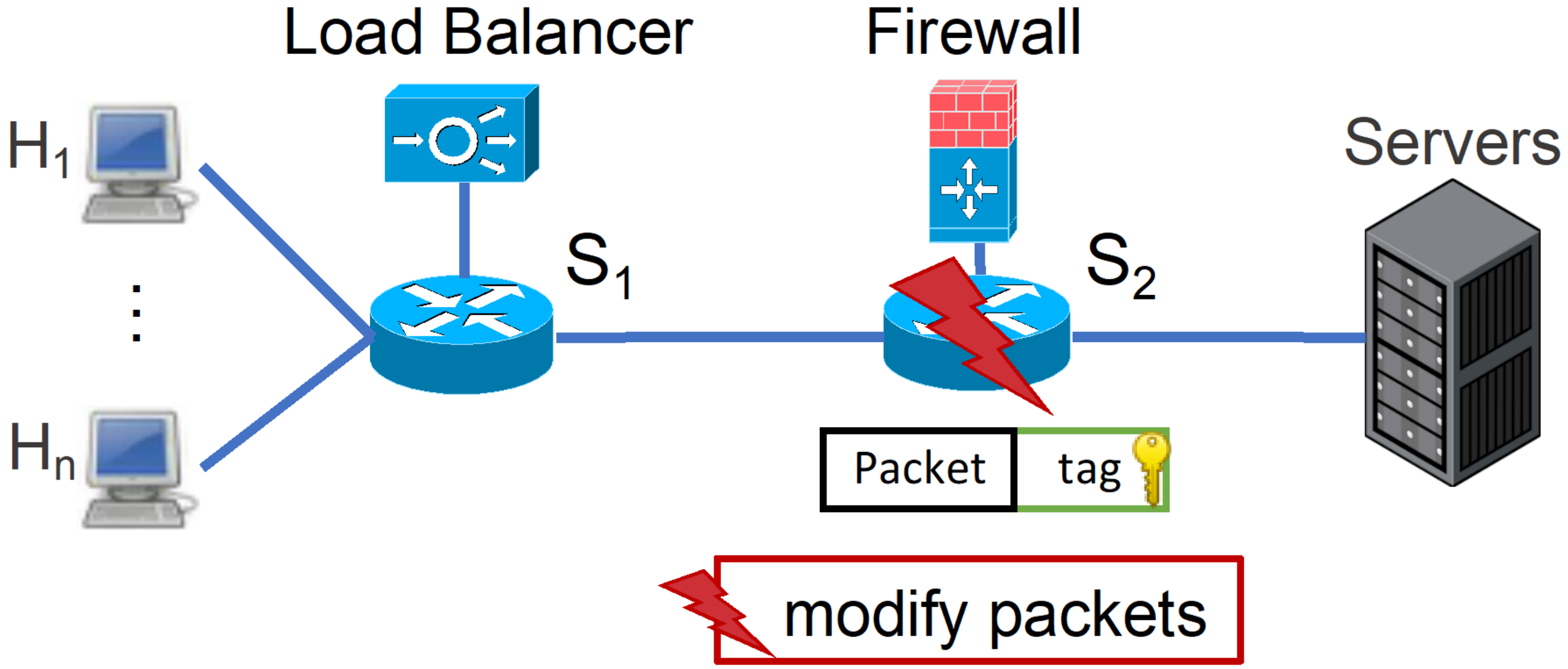
- Long history of work on verifying Internet paths
  [OPT SIGCOMM’14, ICING CoNext’11]

- Too strong:
  - Assumes wild west of the Internet with mutually distrusting ASes

- Too weak:
  - Assumes packet should remain unchanged in transit
  - Assumes intended path known in advance
  - Assumes all forwarding nodes are stateless

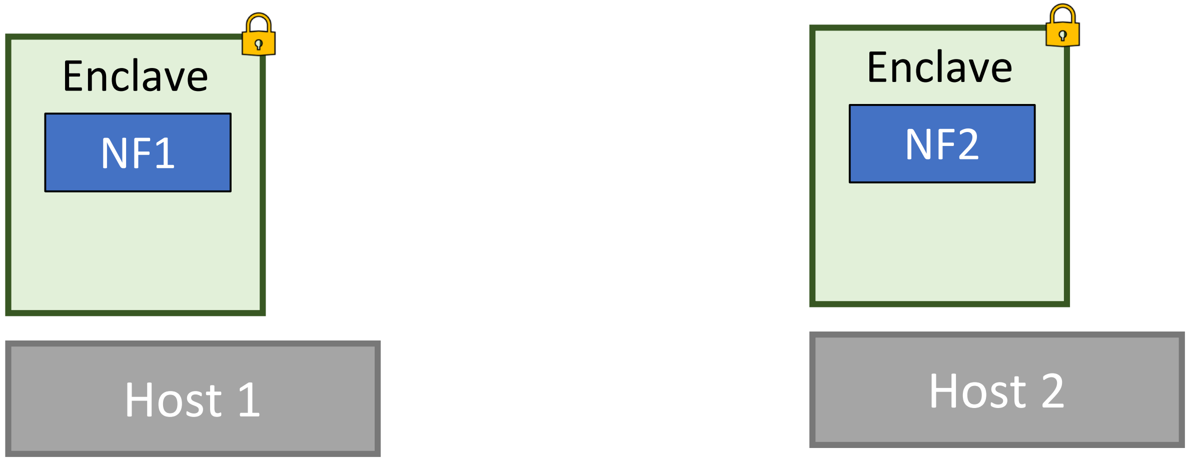
Assumptions
Do not hold
for NFV

Example: Mutable Packets



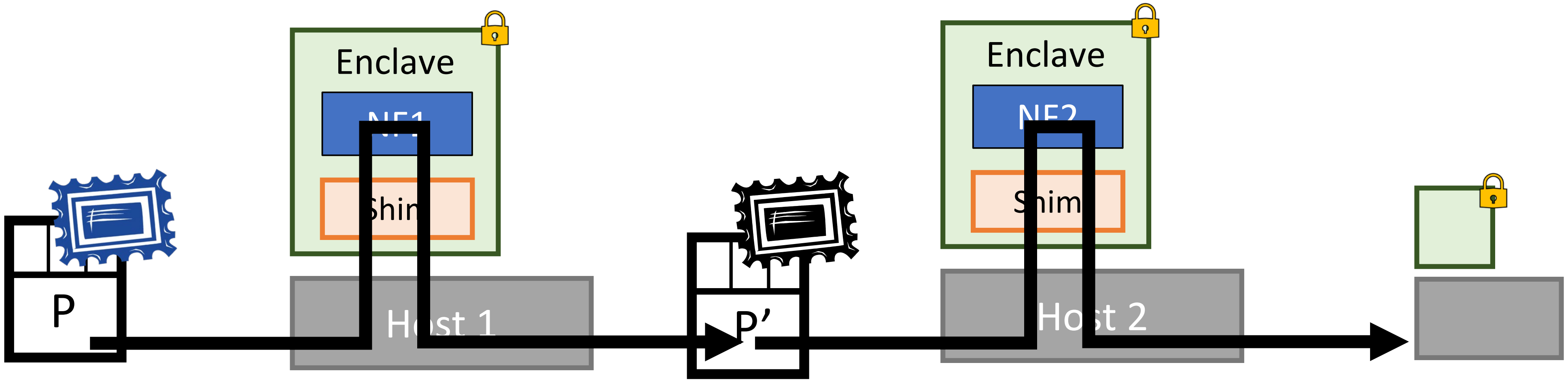
Our Approach: AuditBox

- Run NFs in trusted execution environments – trusted modifications
    - E.g., Intel SGX or Komodo [SOSP 2017]



Our Approach: AuditBox

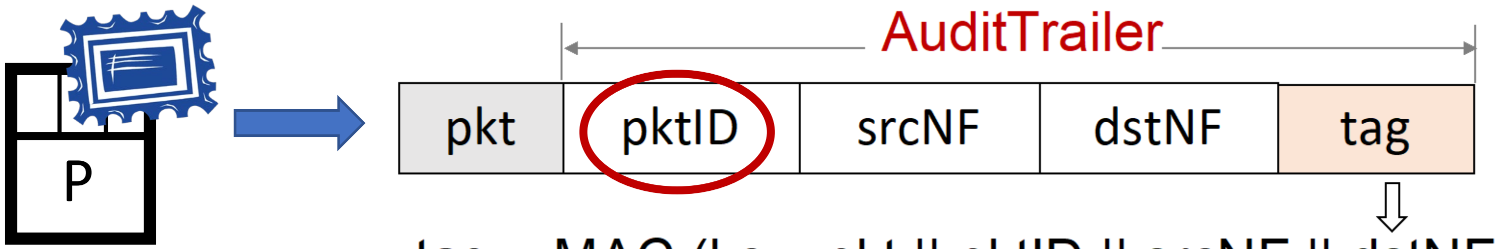
- Run NFs in trusted execution environments – trusted modifications
  - E.g., Intel SGX or Komodo [SOSP 2017]
- NF-hop-by-hop attestation: leverage transitive trust to verifiably enforce policy



Transitive attestation

Design: Audit Trailer

- Created by a trusted entry gateway



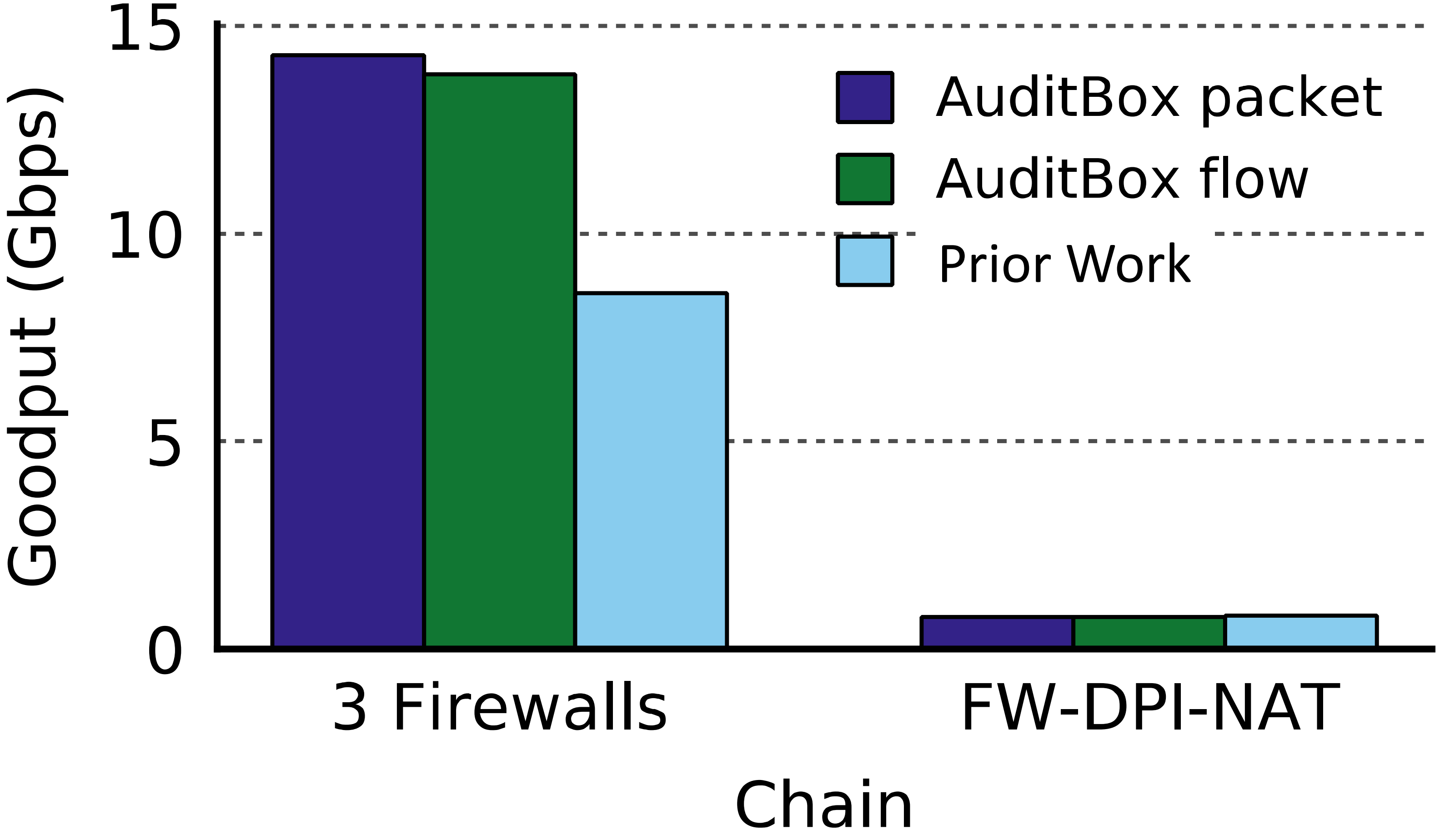
tag = MAC (key, pkt || pktID || srcNF || dstNF)

- Immutable packet ID: create audit trails
- To support flow-level verification:
  - Extend the AuditTrailer with a flow ID and a sequence number

Correctness Guarantees

- Runtime Correctness = Network implements the intended NF forwarding policies
  - Packet correctness: no modification or injection
  - Flow correctness: no modification, injection, reordering, dropping, or duplication.
- Offline Auditability = Must provide an ‘audit trail’
  - Secret logging
- Formal proofs of both runtime and offline properties

Performance Evaluation



Conclusion

- AuditBox offers missing auditing capacities for NFV deployments
- Not only replicates existing manual auditing capacities but *enhances* them with runtime guarantees
- Promote the adoption of NFV

Thank You!

guyuel@andrew.cmu.edu