E2E Enterprise Security with Traffic Visibility

Radia Perlman

Ken Grewal

Problem Description

- Suppose we have E2E security protocols using encryption (IPsec ESP, TLS)
- Sometimes intermediate devices need to look at more of the packet than IPsec/SSL exposes
  - Firewalls
  - Traffic-shaping tools
  - Load Splitters
  - Network monitoring tools
  - Deep packet inspection and scanning (for worms/ viruses)
  - Intrusion Detection & Prevention Systems (IDS/ IPS)

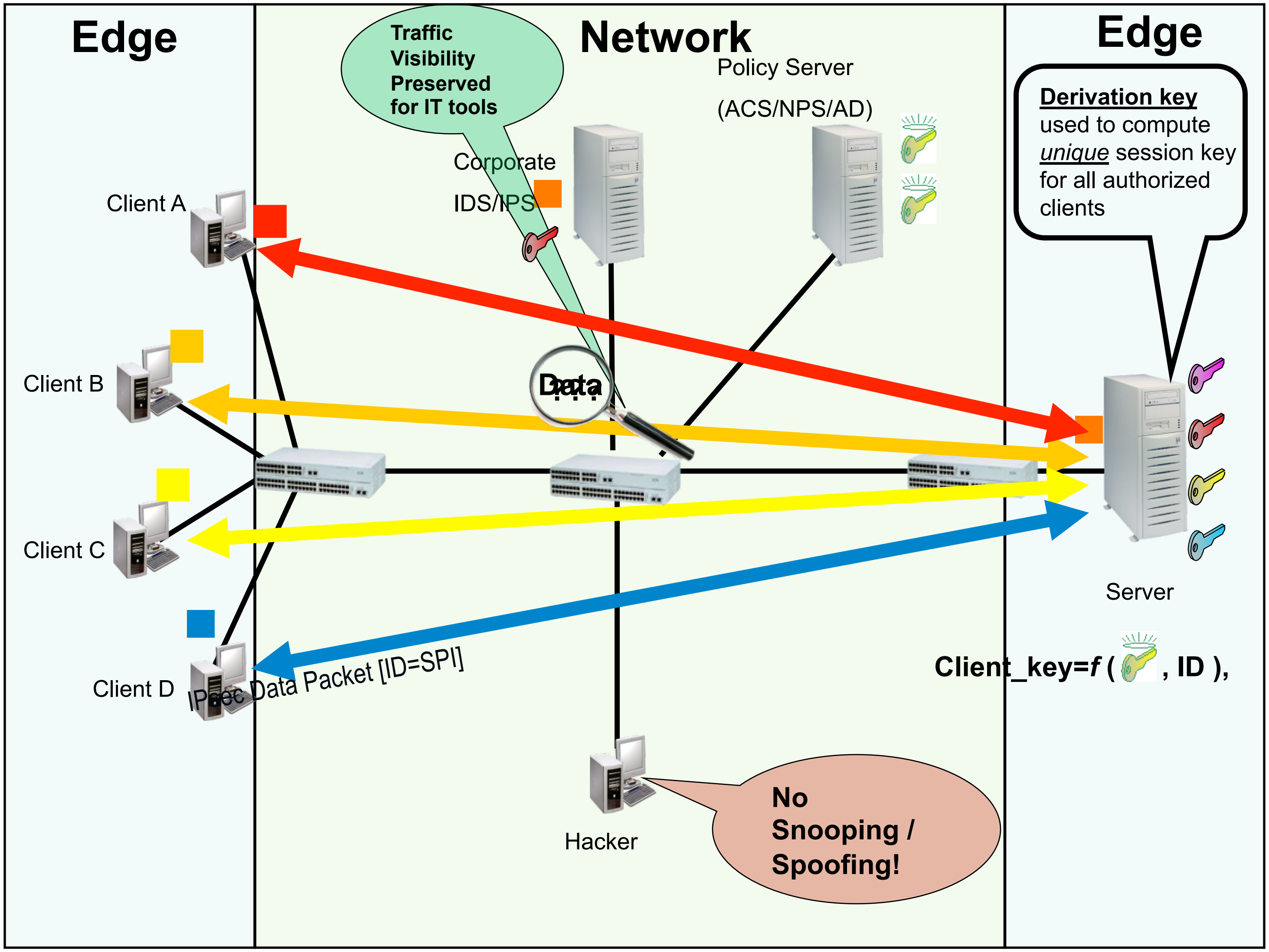
Derived Keys

- A server knows a secret S, from which it derives a session key
- The session key has to be a function of S and things visible in the encrypted packet (e.g., IP address, ports, IPsec SPI)
- The server has to be able to push that key to the client
- If the server wants intermediate boxes to help it, the server gives them S

Pieces of the Puzzle

- Enough information in the unencrypted header to uniquely determine this session's key (e.g., IPsec SPI, IP address)
- A way of pushing the key to the client (e.g., a new method of doing rekeying)
- Modifying the TLS or IPsec header to distinguish packets using derived keys from legacy packets

Enterprise Security



1. Credentials / health check
2. Policy + Key Distribution by central authority
3. Server exchanges auth data and derives session keys, per client
4. Stateless security, scaling with demand
5. Preserve session independence / data authenticity for each client
6. Authorized devices can inspect E2E traffic flows by deriving key

Technology Components

1) Key Derivation

- Use a “Master Key” to create session keys that can be derived per-packet to eliminate data plane cryptographic state maintenance

2) Secure Protocol Requirements

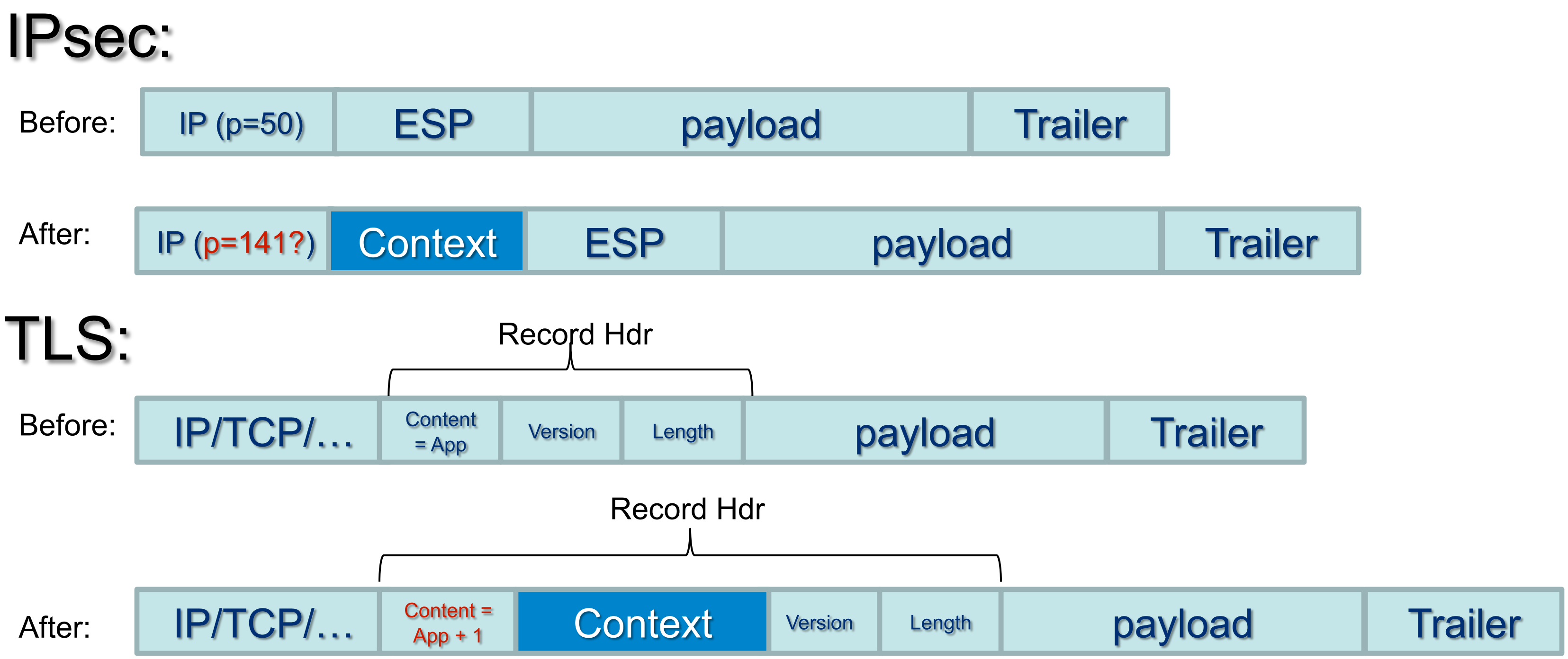
- Data Path
  - Protocol identification for using derived key extensions
  - Additional session context in each packet to allow on-the-fly key derivation
- Control Path
  - Extending the handshake to ‘push’ a derived key from server to client

3) Bifurcated Keys

- Separates the trust boundaries for confidentiality and authenticity
- Provide for separate key material for encryption and integrity while preserving the performance advantages of GCM combined mode operation (single pass confidentiality and integrity)
- IETF draft

Protocol Requirements

- Identification
- Session context in the data packet to allow on-the-fly key derivation



- K_derived = f (K_derivation, *ID*)
- Where ID = session context

Key Distribution

<u>Initiator / client</u> **<u>Responder / Server</u>**

IKE:

➔ Message 1

Message 2 ←

➔ Message 3

Message 4 ← HDR, SK {*SA, Nr, [KEr], Ks, TSi, TSr*}

TLS:

…

[ChangeCipherSpec] ➔

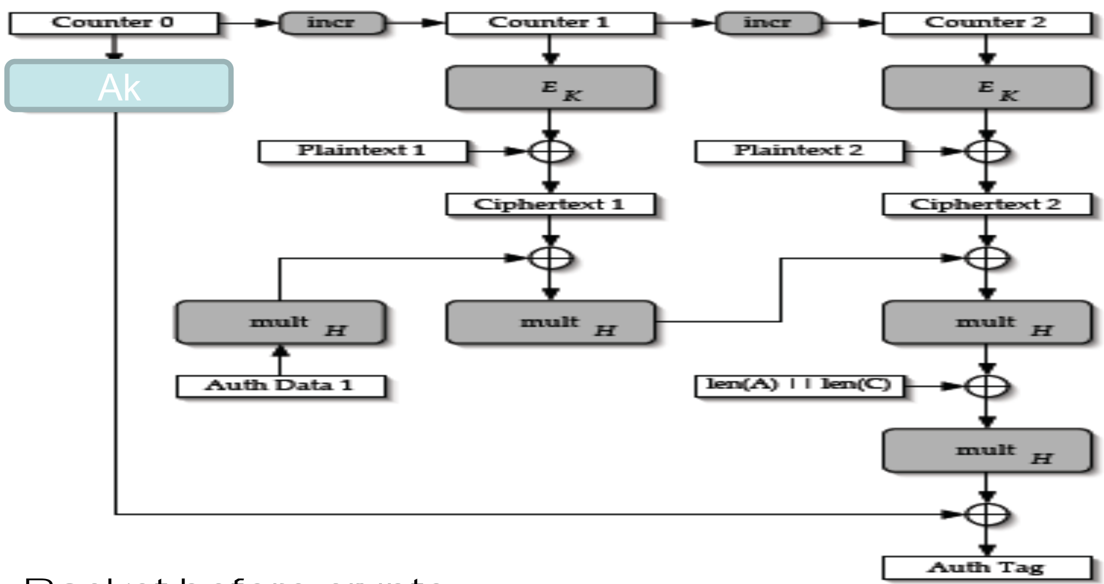
← [ChangeCipherSpec (*Ks*)]

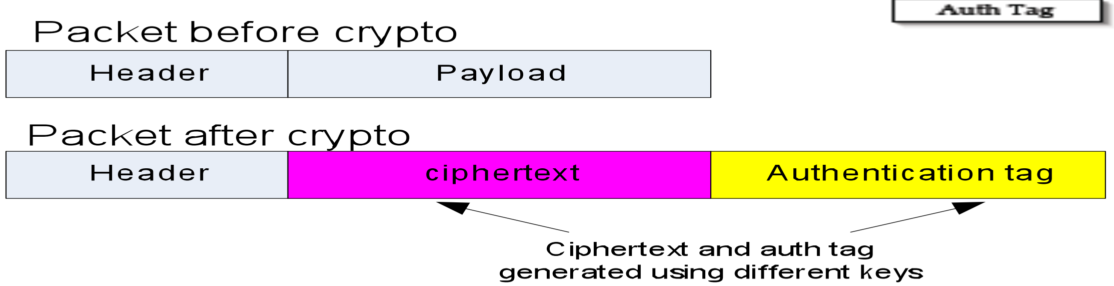
Finished ➔

← Finished

- Observations
  - Key transport / verification happens in the last message(s)
  - Session Key (Ks) is the derived key
- Other permutations possible

Bifurcated Key





- Combined mode algorithm
- Parallelizable
- Highly efficient (10Gb+)
- Two Keys
  1. Encryption Key (Ek)
  2. Integrity Key (Ak)

Enc-Key shared with TIs ; Auth-Key preserves E2E authenticity

Summary / Next Steps

- Traffic visibility is critical to Enterprise environments
- Enterprises will trade security for visibility, unless a solution is provided

---

- Community feedback / interest in solving this problem
- Interested parties – please follow-up via email for further discussion / next steps