DPDK

DATA PLANE DEVELOPMENT KIT

rte_security: enabling IPsec hw acceleration

Boris Pismenny (Mellanox)

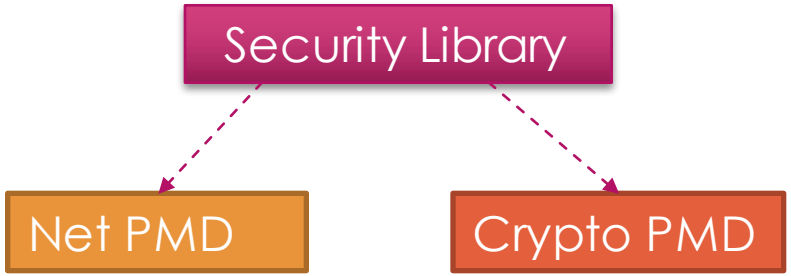
Declan Doherty (Intel)

Hemant Agrawal (NXP)

DPDK Summit - San Jose – 2017

Introduction

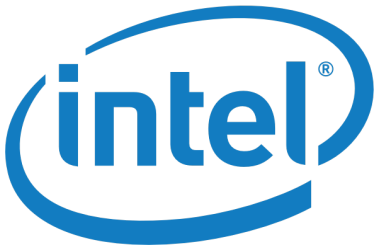
- Framework for management and provisioning of hardware acceleration of security protocols.
- Generic APIs to manage security sessions.
- Security acceleration functions are accessed through security instances which can instantiated on any device type, current supports security instances on Crypto and Ethernet devices.
- Rich capabilities discovery APIs
- Current only targets the support of IP Security (IPsec) protocol.
- Could support a wide variety of protocols/applications
  - Enterprise/SMB VPNs — IPsec
  - Wireless backhaul — IPsec, PDCP
  - Data-center — SSL
  - WLAN backhaul — CAPWAP/DTLS
  - Control-plane options for above — PKCS, RNG



Community Collaboration

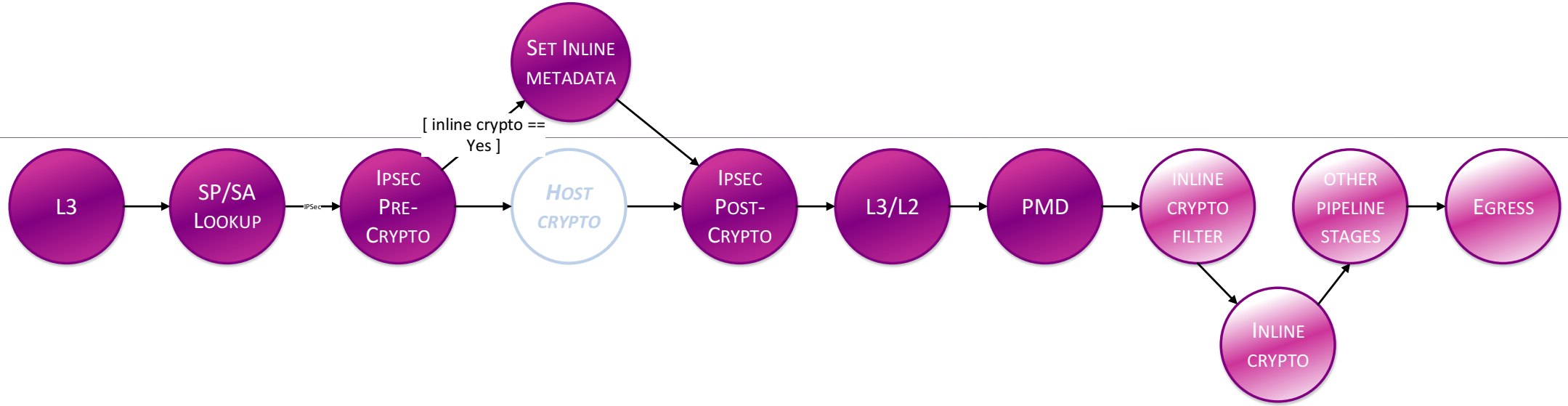
- Collaborative work between Intel, Mellanox and NXP with contributions from:
  - Hemant Agrawal, Declan Doherty, Akhil Goyal, Radu Nicolau, Boris Pismenny, and Aviad Yehezkel.

- rte_security is now part of DPDK 17.11 as *Experimenal* API



Inline Crypto Acceleration

- IO based acceleration performed on the physical interface as packet ingress/egress the system.
- No packet headers modifications on the hardware, only encryption/decryption and authentication operations are preformed.
  - Hardware may support extra features like payload padding, setting of etc.



Lookaside Protocol Acceleration

- Lookaside acceleration model where packet is given to an accelerator for processing and then returned to the host after processing is complete.
- Security function is provided as an extension of a librte_cryptodev crypto PMD.
  - Security session is used in place of crypto session in crypto op when enqueuing and dequeuing packets to the crypto PMD.
- Supports full protocol (IPsec) processing on the accelerator. Including:
  - Add/remove protocol headers
  - Handling SA state information

Library Features

- Protocol agnostic session API for the management of protocol state on underlying hardware.
- Definitions of supported protocols, currently only IPsec, and the parameters for configuring the options. For IPsec this includes:
  - Acceleration type – inline crypto/lookaside protocol/inline protocol
  - Defining security association (SA) parameters such as Tunnel/Transport, ESP/AH, Ingress/Egress as well as associated crypto processing and key material
- Crypto operations are defined using primitives defined in librte_cryptodev limit any redefinition of parameters within DPDK.
- Capabilities APIs to allow dynamic discovery of a instances features.

Session Management

- Session APIs support
  - Create Session

struct rte_security_session *
rte_security_session_create(uint16_t id,

struct **rte_security_session_conf** *conf,

struct **rte_mempool** *mp);

  - Update
  - Destroy
  - Query (Get Stats)

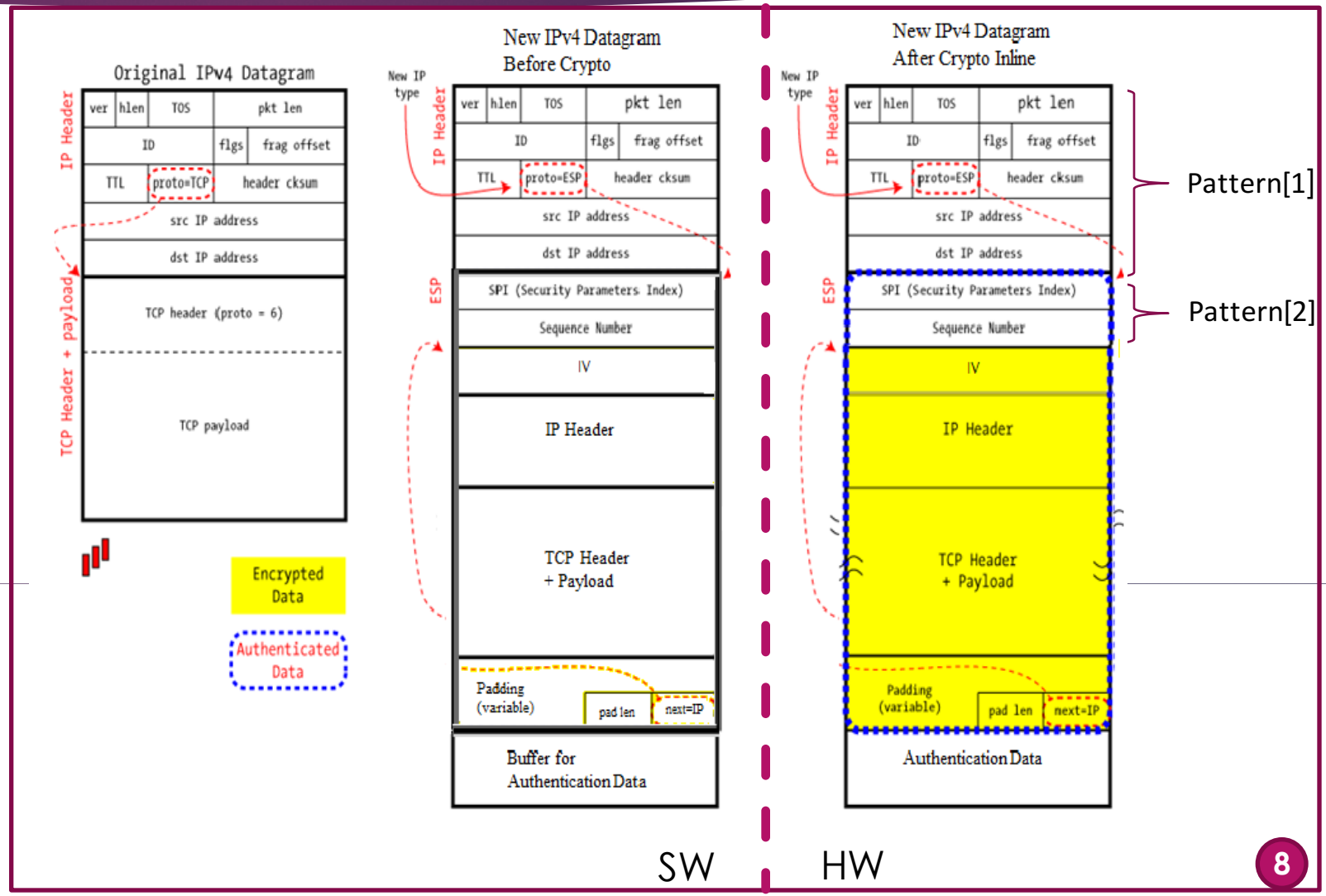
```
/** security session configuration parameters */
struct rte_security_session_conf config = {
        .action_type = RTE_SECURITY_ACTION_TYPE_INLINE_CRYPTO,
        /**< Type of action to be performed on the session */
        .protocol = RTE_SECURITY_PROTOCOL_IPSEC,
        /**< Security protocol to be configured */
        .ipsec = {
                .spi = /**< Security Protocol Index */,
                .salt = /** Salt value */,
                .direction = RTE_SECURITY_IPSEC_SA_DIR_INGRESS,
                .proto = RTE_SECURITY_IPSEC_SA_PROTO_ESP,
                .mode = RTE_SECURITY_IPSEC_SA_MODE_TUNNEL
        },
        /**< Configuration parameters for security session */
        .crypto_xform = /** crypto transforms*/
        /**< Security Session Crypto Transformations */
};
```

Flow Action Programming (Inline Crypto)

```
/** flow parameters */
attr->ingress = 1; /** attr->egress = 1 */

pattern[0].type = RTE_FLOW_ITEM_TYPE_ETH;
pattern[1].type = RTE_FLOW_ITEM_TYPE_IPV4;
pattern[2].type = RTE_FLOW_ITEM_TYPE_ESP;
pattern[3].type = RTE_FLOW_ITEM_TYPE_END;

action[0].type = RTE_FLOW_ACTION_TYPE_SECURITY;
action[0].conf = sa->sec_session;
action[1].type = RTE_FLOW_ACTION_TYPE_ PASSTHRU;
action[2].type = RTE_FLOW_ACTION_TYPE_END;
```



Summary

- Provides an abstraction for provisioning security hw accelerations, initially targeting IPsec.
- Can be used with ethdev and cryptodev
- rte_security + rte_flow = powerful control plane
- Agnostic API to allow applications to use different security accelerations.
- IPsec Security Gateway Sample application is available today using rte_security to support inline crypto (on Intel's IXGBE_NET PMD) and lookaside protocol acceleration (on NXP's DPAA2 CRYPTO PMD).
  - Go try it out!

Future Work

- Further IPsec enablement
  - Further encapsulations
  - LSO + checksum
  - IPsec inline protocol offload
- Further protocol enablement
  - MACsec, PDCP, DTLS, etc would fit under this model.
- Software equivalent enablement
  - It could be possible to offer software equivalent processing under this API, may or may not be desirable depending on protocol and it's processing overhead.

Questions?

Boris Pismenny (Mellanox)

Declan Doherty (Intel)

Hemant Agrawal (NXP)