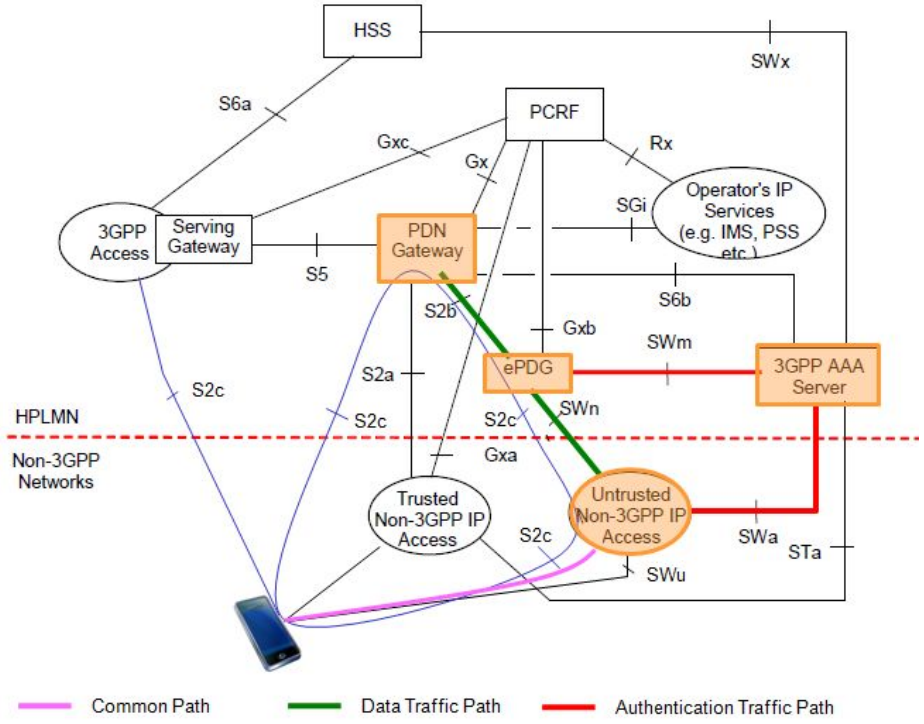
IPsec in Android

android

IMS and IWLAN

- IMS supports TLS and IPsec; IR.92 (VoLTE standard) specifies IPsec. (also IR.51 and IR.94)
- IWLAN uses IPsec for the data plane and IKE (with 3GPP extensions) for configuration.
  - TS 23.234 (requirements), TS 24.234 (stage 3 - protocol), TS 33.234 (security)
  - The prefix assigned to the phone is carried between LTE/UMTS and IWLAN within the tunnel.
- IMS requires Transport mode inside Tunnel mode

IMS and IWLAN



Design Goals/Needs

- Initial Focus on the data plane:
  - IPsec transport mode security on a per-socket basis
  - IPsec tunnel mode security to create encrypted `Networks` (e.g., IWLAN)
  - Combined: encrypted socket over an encrypted `Network` (e.g., SIP over IWLAN)
- UDP encapsulation for IPv4
  - Provide IKE/encap socket to userspace with guarantees of safety
- For IMS (transport), keys are manually generated from EAP-AKA.
- For IWLAN (tunnel), keying is performed by IKEv2 with 3GPP extensions.

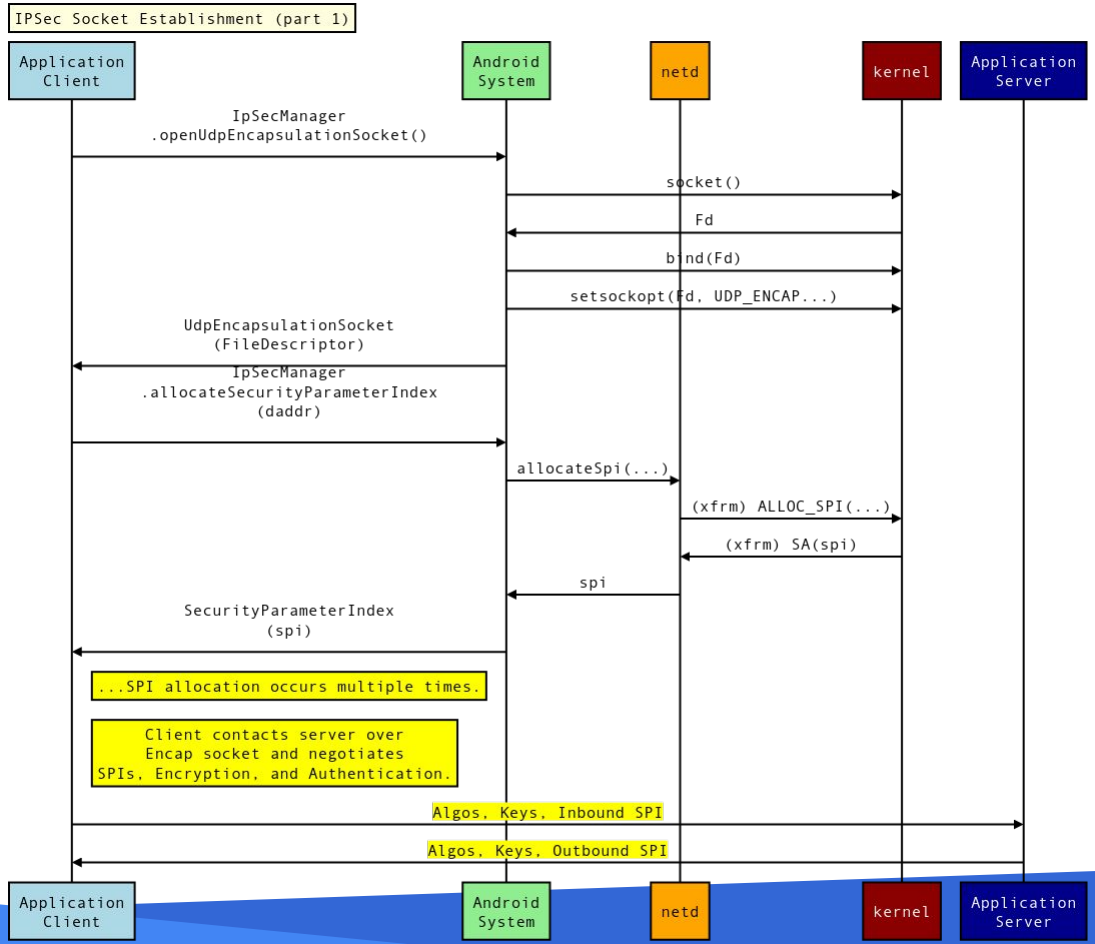
Goals and Constraints

- Multiple entities on-device concurrently configuring IPsec, such as VPNs and IMS.
  - They have no coordination mechanism
- Global policies are difficult in a multi-app environment
  - Per-socket policies are safe for individual socket owners
  - No acquires sent to userspace
- No dropped packets allowed during keying due to latency limits (esp. for transport mode)

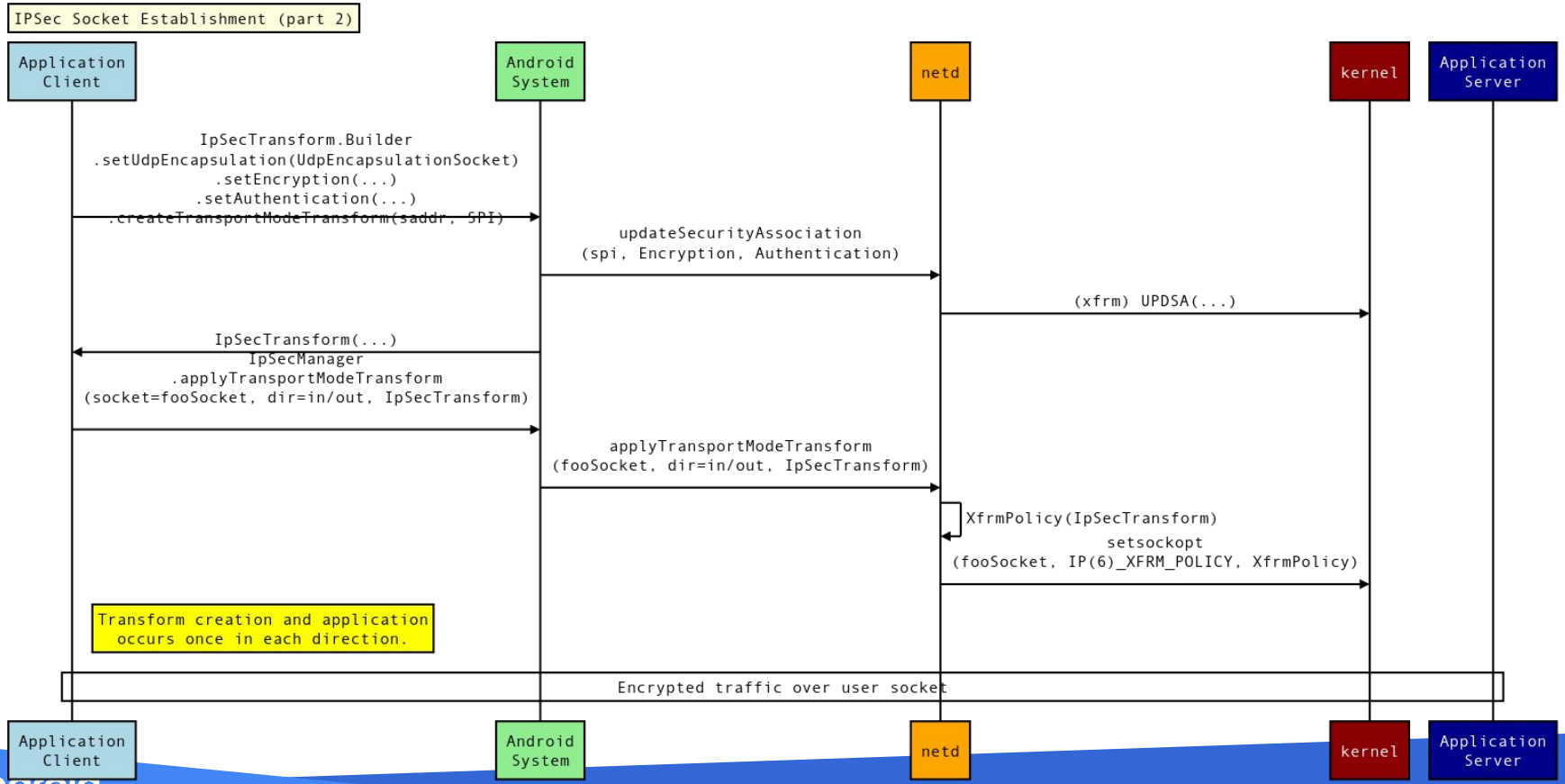
Approach

- SAs are provided by management object - ***IpSecTransform***
- VTIs (in the future, XFRMIs) are created along with two pairs of policies(v4/6, in/out) and collected in a management object - ***IpSecTunnelInterface***
- Association of an SA with a socket or tunnel is performed by calling an **`apply()`** method
  - **`applyXYZ()`** does re-key as follows:
    - SPI in the template selects the SA in the outbound direction by either setting a new socket policy or calling **`UPDPOLICY`**
    - Input direction “just works”™
- **`XFRMA_OUTPUT_MARK`** used to bind the tunnel to an underlying interface

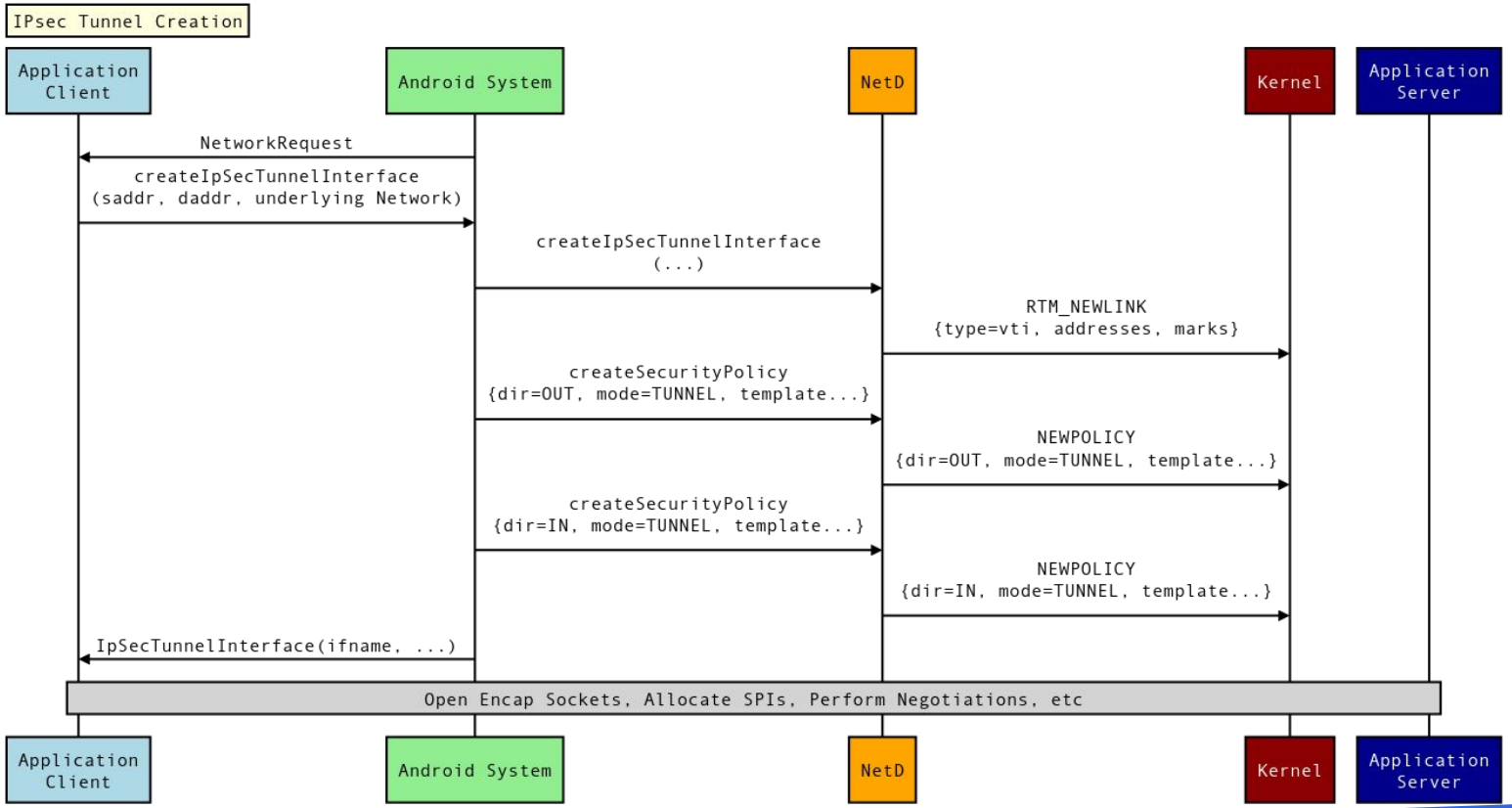
Transport Mode Socket Establishment



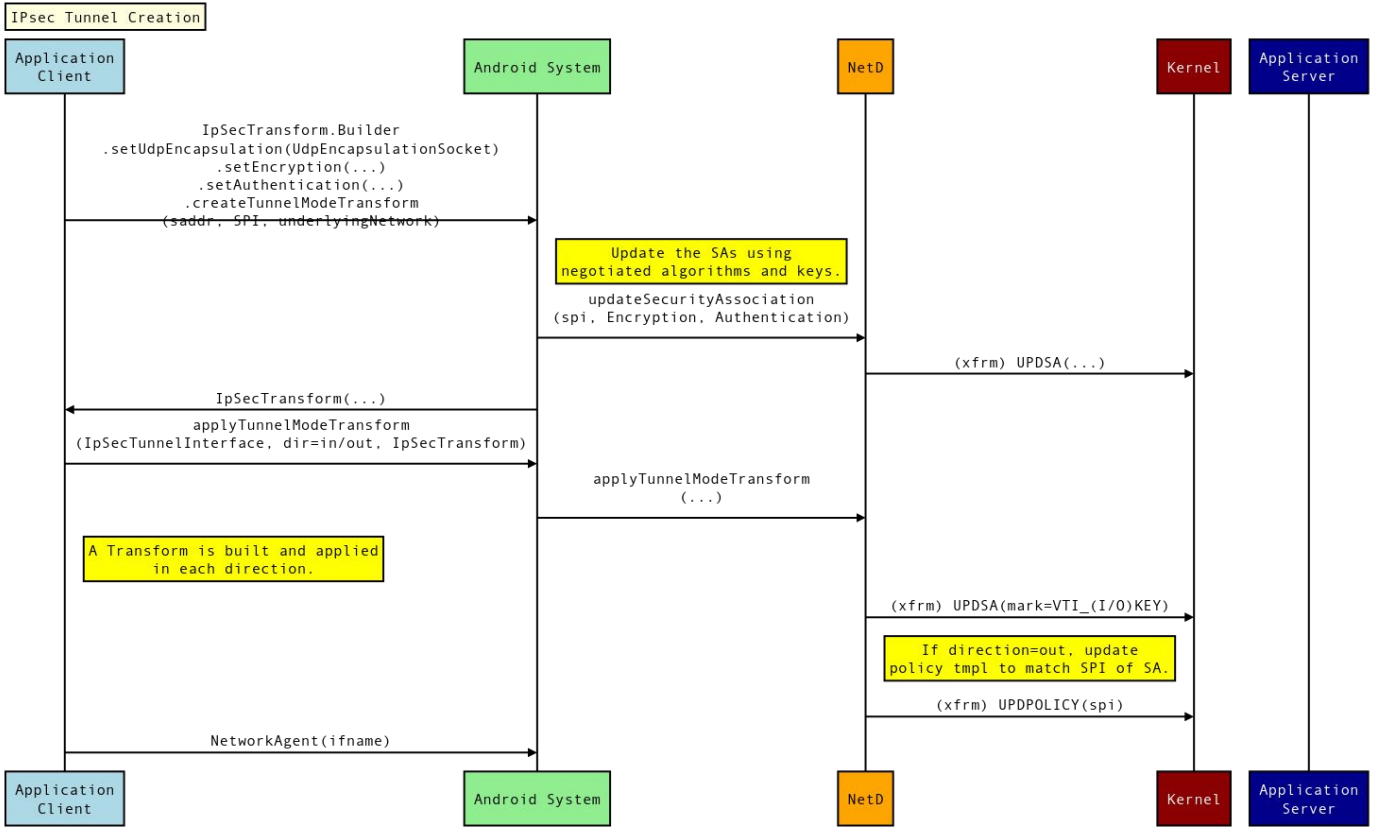
Transport Mode Socket Establishment Contd.



Tunnel Mode Interface Establishment



Tunnel Mode Interface Establishment Contd.



IPsec data usage and firewalling

- Major headache

- Solution:
  - Dependent on `xt_policy` module
  - `uidBillingDone` bit in packet mark (fwmark)
  - Exempt ESP, and count inner packets only
  - For firewalling, allow ESP packets, and packets that have no socket through

- Bypassing powersaving firewalls
  - ESP traffic blanket-exempted (regular apps can’t send/receive it)
  - Forwarding with UDP-encap-ESP allowed

IPsec data usage and firewalling - cases

- Cases, for reference:

			Inbound							Outbound					
			Outer		Inner		Outer	Inner		Inner		Outer		Inner	Outer
			raw-PRE	filter-INPUT	raw-PRE	filter-INPUT	doze	doze	filter-FWD	filter-OUTPUT	mangle-POST	filter-OUTPUT	mangle-POST	doze	doze
Non-IPSec traffic	INBOUND		COUNT	BILL	-	-	ORIGINAL UID	-	-	-	-	-	-	-	-
	OUTBOUND		-	-	-	-	-	-	-	-	-	BILL	COUNT	-	ORIGINAL UID
	FORWARD		COUNT	-	-	-	-	-	-	-	-	-	COUNT		-
Transport Mode	INBOUND	IKE	COUNT	BILL	-	-	ORIGINAL UID	-	-	-	-	-	-	-	-
		ESP	COUNT	BLANKET IGNORE	IGN based on policy module	BILL	BLANKET ALLOW	ORIGINAL UID	-	-	-	-	-	-	-
		UDP-encap-ESP	COUNT	BILL		IGN based on BILLING_DONE mark	ENCAP UID	ORIGINAL UID	-	-	-	-	-	-	-
	OUTBOUND	IKE	-	-	-	-	-	-	-	-	-	BILL	COUNT	-	ORIGINAL UID
		ESP	-	-	-	-	-	-	-	IGN based on policy module	IGN based on policy module	BILL	COUNT	ORIGINAL UID	BLANKET ALLOW
		UDP-encap-ESP	-	-	-	-	-	-	-			BILL	COUNT	ORIGINAL UID	ORIGINAL UID
	FORWARD	Plaintext passthrough	COUNT	-	-	-	-	-		-	-	-	COUNT	-	-
		ESP passthrough	COUNT	-	-	-	-	-	IGNORED BY QTAGUID	-	-	-	COUNT	-	-
		UDP-encap-ESP passthrough	COUNT	-	-	-	-	-		-	-	-	COUNT	-	-
Tunnel Mode	INBOUND	IKE	COUNT	BILL	-	-	ORIGINAL UID	-	-	-	-	-	-	-	-
		ESP	COUNT	BLANKET IGNORE	IGN based on VTI (-i vti+)	BILL	BLANKET ALLOW	ORIGINAL UID	-	-	-	-	-	-	-
		UDP-encap-ESP	COUNT	IGN based on port-specific U32 rule		BILL	ENCAP UID	ORIGINAL UID	-	-	-	-	-	-	-
	OUTBOUND	IKE	-	-	-	-	-	-	-	-	-	BILL	COUNT	-	ORIGINAL UID
		ESP	-	-	-	-	-	-	-	IGN based on VTI (-o vti+)	IGN based on VTI (-o vti+)	BILL	COUNT	ORIGINAL UID	BLANKET ALLOW
		UDP-encap-ESP	-	-	-	-	-	-	-			BILL	COUNT	ORIGINAL UID	ORIGINAL UID
	FORWARD - TUNNEL MODE	Decrypt ESP	COUNT	IGN	IGN based on VTI (-i vti+)	-	BLANKET ALLOW	-	IGNORED BY QTAGUID	-	-	-	COUNT	-	-
		Decrypt UDP-encap-ESP	COUNT	IGN based on port-specific U32 rule		-	ENCAP UID	-		-	-	-	COUNT	-	-
		Encrypt ESP	COUNT	-	-	-	-	-		-	IGN based on VTI (-o vti+)	BILL TO ANDROID-OS	COUNT	-	BLANKET ALLOW
		Encrypt UDP-encap-ESP	COUNT	-	-	-	-	-		-			COUNT	-	IF NO SOCKET, ALLOW

IPsec data usage and firewalling - cases

- Cases, for reference:

		Inbound									Outbound								
		Tunnel		Transport		Plaintext		Doze			Plaintext		Transport		Tunnel		Doze		
		raw-PRE	filter-INPUT	raw-PRE	filter-INPUT	raw-PRE	filter-INPUT	Tunnel	Transport	Plaintext	filter-OUTPUT	mangle-POST	filter-OUTPUT	mangle-POST	filter-OUTPUT	mangle-POST	Plaintext	Transport	Tunnel
INBOUND	IKE-Tunnel	COUNT	BILL	-	-	-	-	TUNNEL UID	-	-	-	-	-	-	-	-	-	-	-
	IKE-Transport		Is ESP / UDP-encap-ESP	IGN based on VTI (-i vti+)	BILL	-	-	TUNNEL UID	TRANSPORT UID	-	-	-	-	-	-	-	-	-	-
	[ESP [ESP]]		BLANKET IGN		BLANKET IGN	IGN based on VTI (-i vti+)	BILL	BLANKET ALLOW	BLANKET ALLOW	ORIGINAL UID	-	-	-	-	-	-	-	-	-
	[UDP-encap [UDP-encap]]		IGN based on port-specific U32 rule		BILL		IGN based on BILLING_DONE mark	TUNNEL ENCAP UID	TRANSPORT ENCAP UID	ORIGINAL UID	-	-	-	-	-	-	-	-	-
	[UDP-encap [ESP]]				BLANKET IGN		BILL	TUNNEL ENCAP UID	BLANKET ALLOW	ORIGINAL UID	-	-	-	-	-	-	-	-	-
	[ESP [UDP-encap]]		BLANKET IGN		BILL		IGN based on BILLING_DONE mark	TUNNEL ENCAP UID	TRANSPORT ENCAP UID	ORIGINAL UID	-	-	-	-	-	-	-	-	-
OUTBOUND	IKE-Tunnel	-	-	-	-	-	-	-	-	-	-	-	-	-	BILL	COUNT	-	-	TUNNEL UID
	IKE-Transport	-	-	-	-	-	-	-	-	-	-	-	IGN based on VTI (-o vti+)	IGN based on VTI (-i vti+)	BILL		-	TRANSPORT UID	TUNNEL UID
	[ESP [ESP]]	-	-	-	-	-	-	-	-	-	IGN based on VTI (-o vti+)	IGN based on VTI (-o vti+)			BILL		ORIGINAL UID	ORIGINAL UID	ORIGINAL UID
	[UDP-encap [UDP-encap]]	-	-	-	-	-	-	-	-	-					BILL		ORIGINAL UID	ORIGINAL UID	ORIGINAL UID
	[UDP-encap [ESP]]	-	-	-	-	-	-	-	-	-					BILL		ORIGINAL UID	ORIGINAL UID	ORIGINAL UID
	[ESP [UDP-encap]]	-	-	-	-	-	-	-	-	-					BILL		ORIGINAL UID	ORIGINAL UID	ORIGINAL UID

Questions

- Policy check isn’t performed for SAs in transport mode with socket policy on input.
  - Minor security issue?
  - Removing socket policies has no effect on inbound direction.
- Will XFRMI match against ‘0’?
  - Can we update the XFRMI on SAs while they are in ACTIVE state? Same question for the mark?
- Can XFRMI support multiple policies?