IP, IPsec, and IKEv2

Luca Campa

University of Klagenfurt – Introduction to Cybersecurity

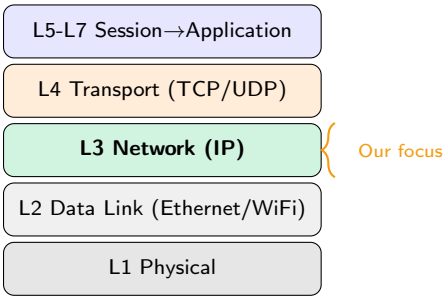
Outline

1. IP Fundamentals
2. IPv6 and MTU Considerations
3. IP Security Problems
4. IPsec
5. IKEv2 (Internet Key Exchange)

IP Fundamentals

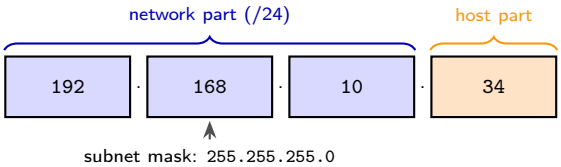
The IP Layer in the ISO/OSI Stack

- IP operates at **Layer 3 (Network layer)**.
- Main role: logical addressing and packet forwarding across multiple networks.
- IP is **connectionless**:
  - no built-in delivery guarantee
  - no built-in ordering guarantee
  - no built-in confidentiality/integrity
- Transport protocols (TCP, UDP, QUIC) run above IP.



Reading an IPv4 Address

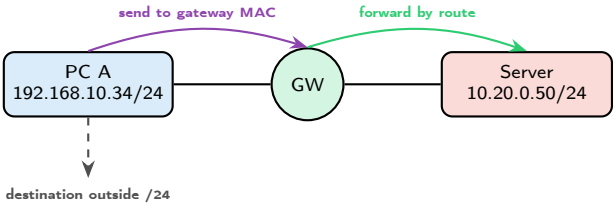
- Example IPv4 address: **192.168.10.34/24**
- Dots split the address into **4 octets** (8 bits each): total **32 bits**.
- Prefix **/24** means:
  - first 24 bits = network part
  - last 8 bits = host part



Result: this host is number 34 in network `192.168.10.0/24`.

Addressing Basics: Network, Host, and Gateway

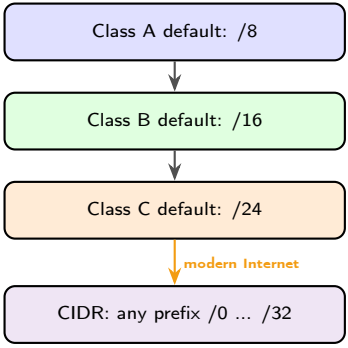
- Every host needs:
  - an IP address
  - a subnet mask/prefix
  - a default gateway
- Same subnet: direct delivery at Layer 2.
- Different subnet: send to default gateway.



IP Classes vs CIDR

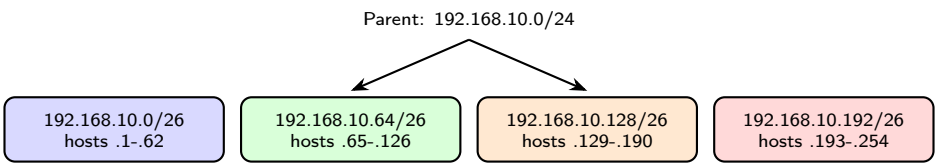
Class	First octet	Default mask	Default prefix
A	1–126	255.0.0.0	/8
B	128–191	255.255.0.0	/16
C	192–223	255.255.255.0	/24

- Classful addressing was rigid and wasted space.
- **CIDR** is classless: choose any prefix length (for example /20, /27, /31).



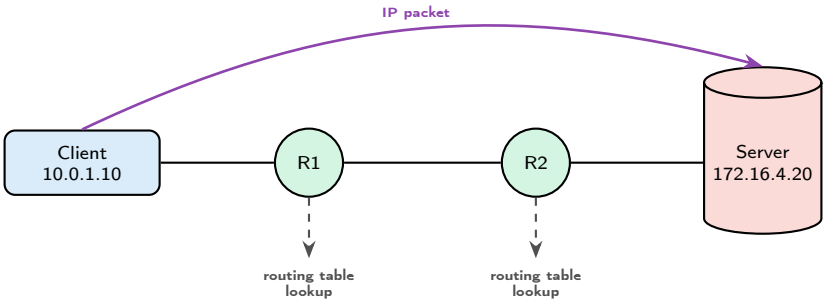
Subnetting with CIDR: Example

- Start with network `192.168.10.0/24` (256 total addresses).
- Split into four /26 subnets: each has 64 addresses.
- Host bits in /26: $32 - 26 = 6$ so total addresses are $2^6 = 64$.
- Usable hosts per /26: $2^{(32-26)} - 2 = 62$.
- Why -2? One address is the subnet ID (all host bits 0) and one is broadcast (all host bits 1).

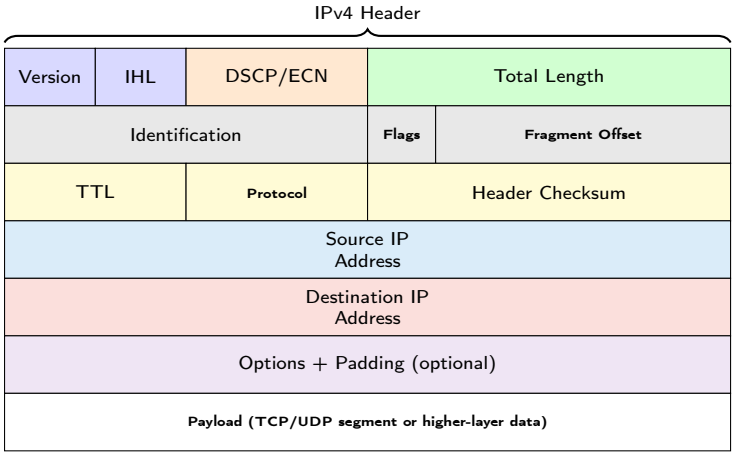


IP Addressing and Forwarding Logic

- Hosts use **source IP** and **destination IP** in each packet.
- Each router makes forwarding decisions based on destination IP and routing table.
- Forwarding path is hop-by-hop until packet reaches destination network.
- Default gateway is used when destination is outside local subnet.

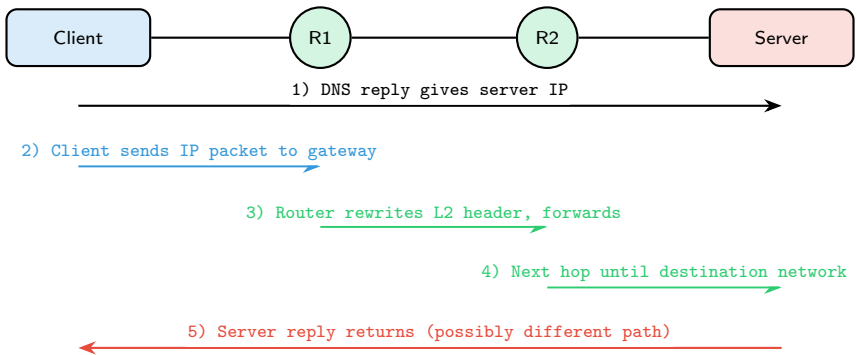


IPv4 Packet Format



- **TTL** limits loop duration; each router decrements it.
- **Protocol** identifies next layer (TCP=6, UDP=17, ESP=50, AH=51).
- **Fragmentation** may occur when packet exceeds MTU (Maximum Transmission Unit).

How IP Exchanges Work

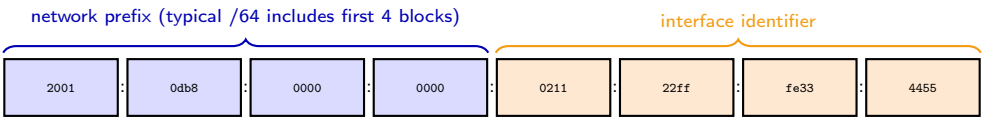


IP itself does not create a session state like TCP; it forwards independent packets.

IPv6 and MTU Considerations

Reading an IPv6 Address

- IPv6 has **128 bits**, written as 8 groups of 16-bit hex blocks.
- Example full form: `2001:0db8:0000:0000:0211:22ff:fe33:4455`
- Compression rules:
  - drop leading zeros in each block
  - use `::` once for one contiguous run of zero blocks



compressed: `2001:db8::211:22ff:fe33:4455`

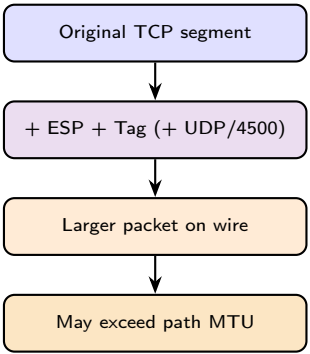
IPv4 vs IPv6

	IPv4	**IPv6**
Header size	Variable (20–60 bytes)	Fixed 40 bytes + extension headers
Checksum in IP header	Yes	No (removed for efficiency)
Fragmentation	Routers and hosts may fragment	Only source host fragments
Address space	32-bit	128-bit
NAT dependency	Frequent in enterprise/home	Less architectural pressure for NAT
IPsec support	Optional	Designed with native support mindset

- Security consequence: IPv6 extension headers must be monitored in filtering policies.
- In real deployments, policy consistency across dual stack is often harder than protocol mechanics.

MTU, Fragmentation, and VPN Overhead

- IPsec adds overhead (ESP header, IV/nonce, tag, optional UDP encapsulation).
- Effective MTU decreases inside tunnels.
- Without proper MSS clamping or PMTUD, symptoms include:
  - stalled HTTPS sessions
  - large-file transfer instability
  - intermittent application failures



Rule: always validate MTU/MSS during VPN rollout and after topology changes.

IP Security Problems

Why Plain IP Is Not Enough for Security

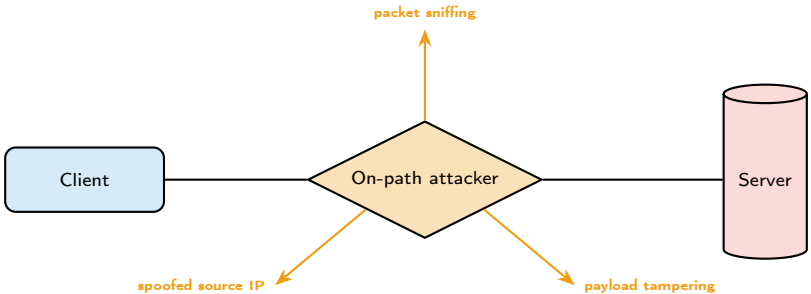
- **No origin authentication**: source addresses can be spoofed.
- **No integrity protection**: on-path modification is possible.
- **No confidentiality**: payload visible to network observers.
- **No anti-replay**: old packets can be resent in some scenarios.
- **Metadata leakage**: addresses, timing, size, and flow patterns remain visible.

These weaknesses motivate security at/around Layer 3: **IPsec**.

IP Threat Taxonomy and Typical Defenses

Threat	Example attacks	Common defenses
Spoofing	Source-IP spoofed packets, reflection	Ingress/egress filtering, ACLs
Tampering	On-path payload/header modification	End-to-end integrity (ESP AEAD, TLS)
Eavesdropping	Passive traffic capture	Encryption (IPsec, TLS, QUIC)
Replay	Re-injection of valid captured packets	Sequence numbers + anti-replay windows
Control-plane abuse	Route hijack, policy manipulation	Authenticated routing, monitoring, anomaly detection

Threat Examples on IP Networks



- Typical scenarios: MITM, route hijack segments, rogue middleboxes, hostile ISP path.
- Goal: protect traffic even when transit networks are untrusted.

IPsec

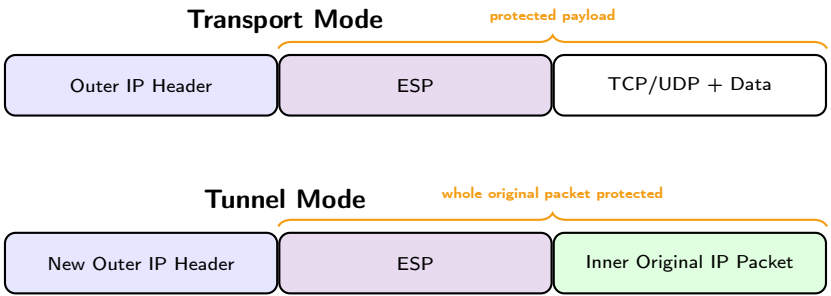
IPsec Overview

- IPsec is a suite of protocols for securing IP at Layer 3.
- Main security services:
  - confidentiality (encryption)
  - integrity and origin authentication
  - anti-replay protection
- Core protocol components:
  - **AH (Authentication Header)**
  - **ESP (Encapsulating Security Payload)**
- Key management is usually done via **IKEv2**.

AH vs ESP

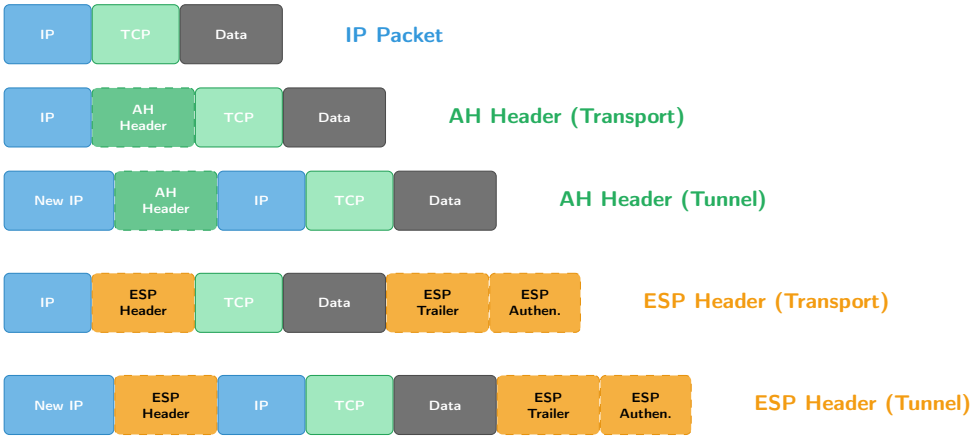
Feature	AH	ESP
IP protocol number	51	50
Confidentiality	No	Yes (encryption)
Integrity/authentication	Yes	Yes (with AEAD or auth option)
Protects IP header fields	Parts of immutable fields	Mainly payload (and inner packet in tunnel mode)
NAT compatibility	Poor	Better (with NAT-T/UDP encapsulation)
Deployment	Rare today	Dominant in practice

Transport Mode vs Tunnel Mode

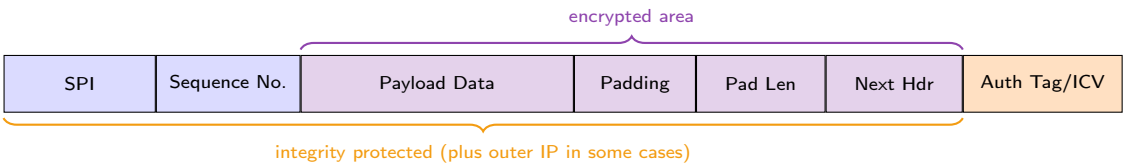


- **Transport mode**: host-to-host protection.
- **Tunnel mode**: gateway-to-gateway or remote-access VPN (most common).

AH and ESP Placement (Transport vs Tunnel)



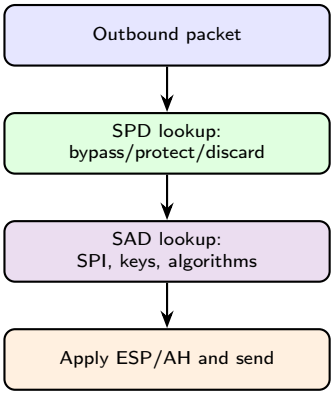
ESP Packet Format



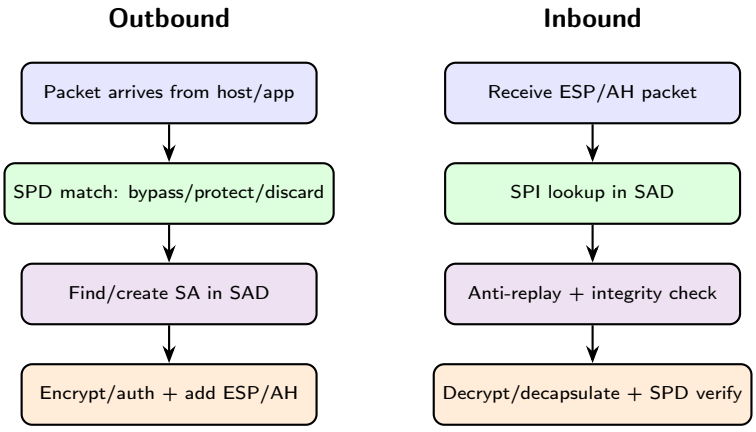
- **SPI** (Security Parameter Index) identifies the Security Association (SA).
- Sequence number supports anti-replay with sliding window checks.

Security Associations, SPD, and SAD

- A **Security Association (SA)** is a one-way logical channel (like a state) with:
  - algorithms and keys
  - SPI for identification
  - lifetime and anti-replay state
- **SPD** (Security Policy Database):
  - policies that match traffic selectors (src/dst IP, port, protocol)
  - rules that say bypass/protect/discard
- **SAD** (Security Association Database):
  - each row corresponds to an active SA
  - indexed by SPI for inbound processing
  - contains all crypto parameters and state for active SAs

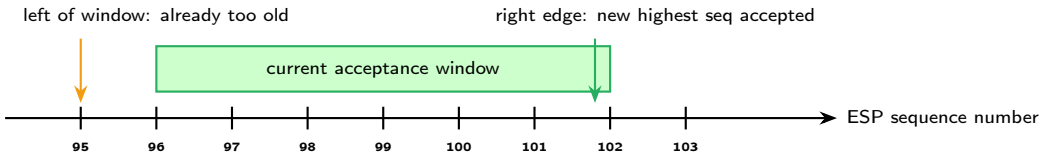


IPsec Processing Pipeline (Outbound and Inbound)



Drops commonly happen at SPD mismatch, replay window failure, or authentication-tag verification.

Anti-Replay Window (Overview)



- Duplicate packets inside the window are dropped.
- Packets older than the window are dropped.
- Window advances as higher valid sequence numbers arrive.

Cryptography Recap for IPsec

- **Symmetric encryption**: protects confidentiality.
- **Integrity/authentication**: MACs or AEAD authentication tags.
- **AEAD** (e.g., AES-GCM, ChaCha20-Poly1305): combine encryption and integrity.
- **Anti-replay**: uses monotonically increasing sequence numbers and replay windows.

Example with AEAD:

$$\text{ciphertext}, \text{tag} = \text{AEAD-Encrypt}(K, \textit{nonce}, \textit{plaintext}, \textit{AAD})$$

$$\text{valid?} = \text{AEAD-Verify}(K, \textit{nonce}, \textit{ciphertext}, \textit{tag}, \textit{AAD})$$

AAD may include selected headers that must be authenticated but not encrypted.

Algorithm Choices and Policy Baseline

Component	Recommended baseline	Notes
ESP encryption/integrity	AES-GCM-128/256 or ChaCha20-Poly1305	Prefer AEAD suites
PRF in IKEv2	HMAC-SHA2 family	Avoid SHA-1 (deprecated)
Diffie-Hellman group	Elliptic Curve groups	
Authentication	Usually via Certificates or EAP methods	PSK (Pre-Shared Key) only with strong secrets
SA lifetime	Time/volume based re-key policy	Balance security and operational stability

IPsec Deployment Patterns

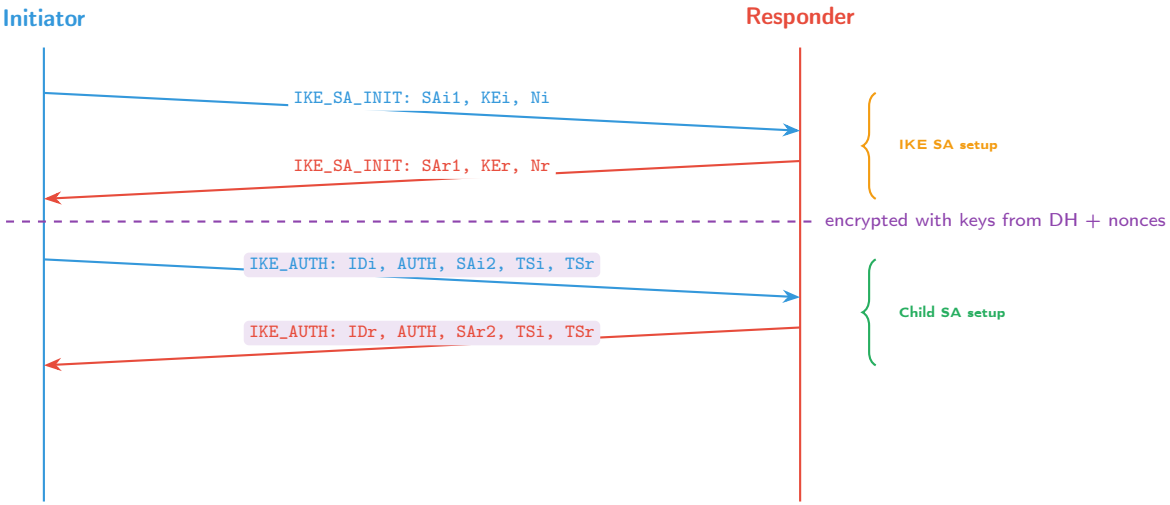
- Site-to-site
- Host-to-site

IKEv2 (Internet Key Exchange)

Why IKE Is Needed

- IPsec needs fresh keys, algorithms, and lifetimes to build SAs.
- Manual key configuration does not scale and is error-prone.
- **IKEv2** automates:
  - peer authentication
  - Diffie-Hellman key agreement
  - SA negotiation and rekeying
  - liveness checks and mobility support

IKEv2 High-Level Exchange



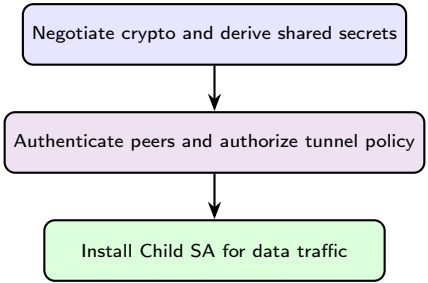
IKEv2 Messages

IKE_SA_INIT

- SA proposals (algorithms)
- DH public values (KEi/KEr)
- nonces (Ni/Nr)

IKE_AUTH

- peer identities (IDi/IDr)
- authentication proofs (AUTH)
- first Child SA proposals
- traffic selectors (TSi/TSr): define what traffic the Child SA will protect



IKEv2 Cryptography Recap

- **Diffie-Hellman (DH)**: peers derive a shared secret over an untrusted network.
- **Nonces**: random values that ensure freshness and key separation.
- **PRF** (pseudo-random function): derives multiple session keys from shared material.
- **Authentication**: certificates (RSA/ECDSA), raw public keys, or EAP methods.

Example key derivation from DH Shared Key and Nonces:

$$SKEYSEED = \text{prf}(N_i \| N_r,\ g^{ir})$$

$$\{K_1, K_2, \ldots\} = \text{prf+}(SKEYSEED,\ \text{context})$$

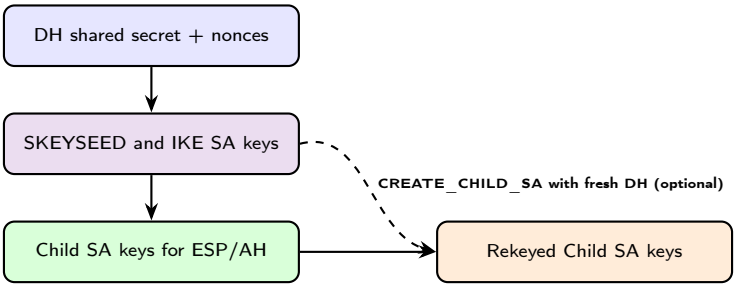
These keys protect IKE messages and IPsec Child SAs. What is g^{ir}?

Authentication Methods in IKEv2

Method	Typical usage	Strengths / Caveats
Pre-Shared Key (PSK)	Small controlled environments	Simple, but weak secrets and reuse are risky
X.509 certificates	Site-to-site and enterprise VPN	Strong identity model with PKI management overhead
EAP-based auth	Remote access (users/devices)	Flexible integration with AAA systems

- AAA stands for Authentication, Authorization, and Accounting, often used in enterprise environments for user/device management.

IKEv2 Key Hierarchy and Perfect Forward Secrecy



- With fresh DH during rekey, compromise of one key epoch does not expose past sessions (**Perfect Forward Secrecy**).

Rekey, Mobility, and Operational Notes

- **CREATE_CHILD_SA**: rekey existing IPsec SAs without full re-authentication.
- **INFORMATIONAL**: delete SAs, report errors, liveness checks.
- **MOBIKE** extension: supports client IP changes (e.g., WiFi to LTE) with tunnel continuity.
- NAT traversal typically encapsulates ESP in UDP/4500.

Stable VPNs need correct timers, MTU/MSS tuning, and clean certificate lifecycle handling.

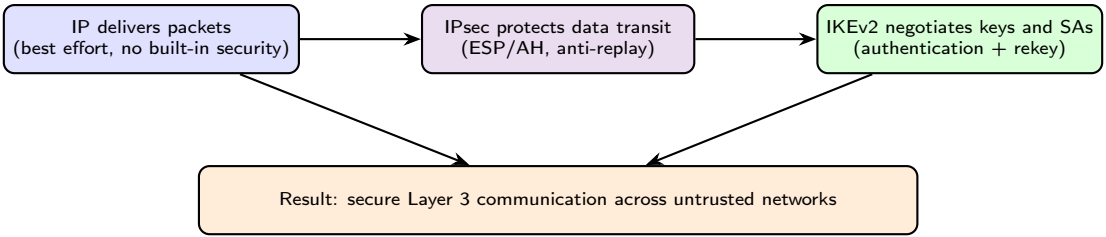
Common Deployment Issues

- **Proposal mismatch**: peers share no common algorithm set.
- **Identity mismatch**: certificate does not match expected ID.
- **Selector mismatch**: traffic selectors are different or too narrow.
- **NAT-T issues**: UDP/4500 filtered or fragmented packets dropped.
- **Clock skew**: certificate validation failures and re-key instability.

Diagnostic:

1. Is Phase 1/IKE SA established?
2. Is Child SA installed?
3. Do policy selectors match expected subnets?
4. Are ESP packets and replay counters increasing?

Secure Layer 3



CLI Example

- Inspect routes and interfaces:
  - `ip route`
  - `ip addr`
- Observe IPsec SAs and policies (Linux with strongSwan):
  - `ip xfrm state`
  - `ip xfrm policy`
  - `swanctl -list-sas`
- Capture IKE/IPsec traffic:
  - `sudo tcpdump -ni <iface> 'udp port 500 or udp port 4500 or proto 50'`

Only run in authorized lab or enterprise environments.

CLI Example (Cont'd)

1) Verify IKE and Child SAs:

```
swanctl --list-sas
```

2) Verify kernel xfrm states and policies:

```
ip xfrm state
ip xfrm policy
```

3) Capture control and data traffic:

```
sudo tcpdump -ni <iface> 'udp port 500 or udp port 4500 or proto 50'
```

References (Selected)

- RFC 791: Internet Protocol (IPv4).
- RFC 8200: Internet Protocol, Version 6 (IPv6).
- RFC 4301: Security Architecture for IP.
- RFC 4303: IP Encapsulating Security Payload (ESP).
- RFC 7296: Internet Key Exchange Protocol Version 2 (IKEv2).
- RFC 8247: Algorithm implementation requirements for IKEv2.

Questions?