IPSEC and DNSSEC

Josh Wilcox (jw14g24)

April 18, 2025

Table of Contents

Overview

- IPsec is a secure network protocol that can both **authenticate** and **encrypt** packets
- IPsec is extensivley used in VPNs
- Includes protocols for establishing **mutual authentication**
  - Both communicating parties verify each other's identities
  - Prevents unauthorized access by requiring both sides to prove who they are
  - Typically achieved using cryptographic methods (e.g., certificates, pre-shared keys)

Security Architecture

- Initially, IPv4 had major security limitations with few provisions to ensure authentication and security
- IPsec is unique by working on the **internet** layer of the OSI model - where other similar security systems tend to work on the tranport or application layer.
- IPsec uses the following protocols:
  - **Authentication Header**
    - Provides **authentication**
    - Does ***not*** **encrypt**
  - **Encapsulating Security Payload**
    - Provides **authentication**
    - Provides **confidentiality** through **encryption**
  - **Internet Key Exchange**
    - Provides a framework for authentication and key exchange

Authentication Header

Overview of the AH

- Ensures **connectionless** data integrity by using a hash function and a secret shared key in the AH algorithm
  - AH calculates a unique ”fingerprint” (hash) of the packet it is dealing with using a shared key
  - If anyone modifies the data during transmission, the fingerprint won't match when verified
  - The receiver recalculates the fingerprint and compares it to the one sent, rejecting packets that don't match
- AH also guarantees the data **origin** by authenticating IP packets
  - The AH contains a cryptographic signature derived from a secret key known only to sender and receiver
  - This signature proves the packet originated from the expected sender
  - Without knowledge of the shared secret key, an attacker cannot generate a valid AH
  - This prevents IP spoofing attacks where someone might pretend to be a trusted source

Encapsulating Security Payload

What ESP does

- Provides origin authenticity through source authentication
- Provides data integrity through hash functions
- Provides confidentiality through **encryption** of IP packets
- Can also support encryption-only and authentication-only configs
  - This is discouraged for security reasons
- In **Transport Mode**, ESP does not provide integrity and authentication for the **entire** IP packet
  - Only protects the IP payload (data portion)
  - Original IP header remains unprotected
  - Attackers could potentially modify unprotected header fields
  - Header information like source/destination addresses remains visible
- In **Tunnel Mode** - where all of the packet is encapsulated with a new header added - ESP is afforded to the **entire** inner packet while only the outer header remains unprotected

Internet Key Exchange

What is IKE

- Protocol used to set up a **security association**
  - A security association (SA) is the establishment of shared security attributes between two network entities to support secure communication.
- It uses **X.509** certificates of authentication
  - These certificates are either pre-shared or distributed using DNS and a Diffie–Hellman key exchange to set up a shared session secret

Phases

- **Phase 1:** A secure authenticated communication channel is created using Diffie-Hellman Key exchange
  - This generates a shared secret key to encrypt further communications
  - This results in a single bi-directionaly security association
  - Runs in either Main Mode or Aggressive Mode. Main Mode protects the identity of the peers and the hash of the shared key by encrypting them; Aggressive Mode does not.
- **Phase 2:**
  - The secure channel made in phase 1 is used for services like IPsec

Modes of Operation

Transport Mode

- Only the **payload** of the IP packet is encrypted or authenticated
- The original IP header remains intact and visible
- **How to remember**: "Transport" mode only protects the **transported data** (payload), not the delivery information
- **Use cases**:
  - Host-to-host communications where endpoints trust each other
  - Applications requiring end-to-end security without network-level protection
  - Situations where network devices need to see routing information
- **Why these properties?**
  - Lower overhead (no additional headers)
  - Allows for network functionality like QoS and traffic management
  - Compatible with NAT in some configurations
  - Preserves original IP addressing scheme

Tunnel Mode

- The **entire** original IP packet is encrypted and authenticated
- It is then encapsulated into a **new IP packet** with a new header
- **How to remember**: "Tunnel" mode creates a secure **tunnel** that completely hides the original packet (like a train in a tunnel)
- **Use cases**:
  - VPNs (site-to-site or remote access)
  - Communication between security gateways
  - Protection against traffic analysis
  - When endpoints aren't IPsec-aware but gateways are
- **Why these properties?**
  - Hides internal addressing schemes from outside observers
  - Protects against traffic analysis by hiding who is actually communicating
  - Essential for network-level security where endpoints may not support IPsec
  - Allows secure traversal across untrusted networks