UNH-IOL
IPsec Introduction

Timothy Carlin

May 23, 2012

Overview

What you'll learn

- IPsec as a Technology
- IPsec as an Architecture
- What the packets look like
- How to read them
- Tools

ipsec, IPSec, IPSEC, IPsec

What it can mean

- A lot!
- Can refer to:
  - Encryption
  - Protection
  - Keying
  - VPNs
  - Generic Security (Think Firewall)

What it is

- Lots of RFCs!
  - 4301 - IPsec
  - 4303 - ESP
  - 5996 - IKEv2
  - 4835 - Required Algorithms
  - 4945 - Public Key Infrastructure (PKI)

Really

- Infrastructure/Guidelines/Rules to offer Protection of Network Traffic
  - What traffic to protect
  - How protect it
- That's it.
- It's not difficult!
- But it is **detailed**.

IPsec Architecture

RFC4301

Defines:

1. Databases
2. Modes
3. **External** Behavior
4. ...

Databases

- Security Policy Database (SPD)
  - Stores Policies
  - Packet Oriented
  - Specify *Action* (What)
- Security Association Database (SAD)
  - Stores Algorithm Information
  - Linked to by a Policy
  - Specify *Protection* (How)
- Peer Authorization Database (PAD)
  - More on this later (Dynamic Keying)

These Databases and the entries are completely independent, yet inextricably intertwined!

Policy Entry

Stores items associated with processing and diverting traffic.

- Source/Destination (*Data* Endpoints)
- Upper Layer/Next Protocol
  (e.g. TCP/ICMPv6/UDP)
- Source/Dest Port or Protocol Type
  (e.g. Port 21, 80, or ICMP Type 0x80, 0x81
- Direction
- Mode (Transport or Tunnel)
- Action (Bypass, Discard, IPsec)
- Link, Pointer, or index to SA
- And more.

Security Association

Stores items associated with processing traffic for IPsec

- Source/Destination (*Tunnel* Endpoints)
- SPI (Security Parameter Index)
- Encryption Algorithm and Key
- Authentication Algorithm and Key
- Mode (Transport or Tunnel)
- Sequence Numbers
- Protocol (ESP, ...)
- Timers, Counters, etc.

Device Types

Two types of IPsec Devices:

- End-Node
  - Like it sounds, provides services only for itself
  - Hosts are usually End-Nodes
- Security Gateway
  - Provides tunneled IPsec services for other devices
  - Routers can usually be SGWs, Hosts can be, without being a router

Don't think of these as Host/Router!! They are different, and independent device types!

Packet Modes

Two ways to process packets:

- Transport Mode
  - Data Occuring **after** IP is Encrypted
  - End-to-End Encryption
  - Addresses in the Clear
  - End-Node - MUST, SGW - MAY
- Tunnel Mode
  - New IP Header Inserted (Outer/Inner)
  - Like other tunneling
  - End Devices Need not be aware of services
  - End Devices Identity protected
  - Somewhat more complicated
  - MUST for End-Node and SGW

Configuration

Two different methods of Configuration, or Keying:

Conf. Method #1

Manual

- There is a lot to configure, and it's required to support it
- Obviously, this leaves room for error
- Mostly used for debugging (though usually causes it)
- Should **NEVER** be used in production networks (keys never change!)
- Pay attention OSPF.
- Exponentially bad.

Conf. Method #2

Automated

- There is still a lot to configure!
- Still a lot of room for error!
- But! Once it's configured correctly, it's good forever.

Protocols

ESP

ESP (Encapsulating Security Payload)

- Just a packet format
- No handshake, no hello's, no negotiation
- Slides directly above IP in Transport Mode
- Slides between two IP Headers in Tunnel Mode

ESP (cont.)

Visible Fields

- SPI
- Sequence Number

Encoded Fields

- IV (Initialization Vector)
- Payload Data
- TFC Padding
- Padding
- Pad Length
- Next Header
- ICV

IKEv2

Internet Key Exchange

- Second version of the protocol, the first was lousy
- Automatically negotiates algorithms and keys
- No need to worry about correct key length
- Still needs configuration
- Authentication is a huge deal with IKE

IKEv2 (cont.)

Protocal has 3 Parts:

- Protect Negotiation
  - Negotitate Algorithms and Keys
  - Uses Diffie-Hellman, complicated math
- **Authenticate** Identity of Yourself and Peer
  - Pre-Shared Keys - Password
  - Public Key Infrastructure (PKI) - Certs
  - EAP - Something Else
- Negotiate who to protect, and how to protect
  - Another set of Algorithms and Keys
  - Data Endpoint - *Traffic Selectors*
  - Other Things (Configuration, VPN info, Vendor IDs, ...)

IKEv2 (cont.)

Required Configuration

- Remote Tunnel Endpoint (The other guy)
- Authentication Credentials
  - Pre-Shared Key OR
  - Certificate Chain

Optional Configuration

- Mode (Transport/Tunnel)
- Protected Range of Addresses
- Algorithm Limitations
- Different Identification Types
- SA Lifetimes
- Rekeying Timers
- Mobility
- Perfect Forward Secrecy
- Sequence Numbers
- Probably More! (Implementation Dependent)

Algorithms

Encryption

Required (Get used to these)

- 3DES-CBC
- AES-CBC
- NULL

Others

- AES-CTR
- Camellia

Authentication/Integrity

Required (Get used to these)

- HMAC-SHA1
- HMAC-SHA256
- AES-XCBC

Others

- NULL
- HMAC-MD5

USGv6 and Logo

IPsec Test Suites

IPsec

- Conformance and Interoperability
- End-Node and SGW
- Different algorithms
- Different situations/topologies
- Only a couple error condition tests
- Pretty small

IKEv2 Test Suites

IKEv2

- Conformance and Interoperability
- End-Node and SGW
- Lots of protocol testing. (Somewhere between the state-machine tests and DAD)
- Also tests Algorithms and situations/topologies
- Lots of different error condition tests
- Big

USGv6 and Logo

- Both the USGv6 and IPv6Ready Logo Programs use the same test documents.
- Unlike IPv6 Base, and AddrArch, with IPsec the two programs have different requirements.
- This is something to pay attention to, depending on what the vendor is looking for.

USGv6 IPsec

- Conformance - All Tests Required
- Interoperability - 3 Devices
  - 1 End Node
  - 1 SGW
  - 1 More (Either Type)
- This amounts to 2 Rounds

Logo IPsec

- Conformance - Most Tests Required, we test all anyway
- Interoperability - 4 Devices
  - 2 Transport Mode
  - 2 Tunnel Mode
- Tunnel Mode Not Required!

What It Means

- We run everything, as much as we can
- Then figure out what it means later
- Sometimes end up having 5 Interop partners
- A lot of testing is just configuration, everything is in the test suites!!

Tools

Setkey

Setkey is our favorite.

- Linux Based (also similar on FreeBSD)
- Manual Configuration
- Display All Configuration (Manual or Auto)
- We have lots of experience with this, and lots of scripts to make testing easy.
- man setkey for more information!

ip xfrm

Similar to setkey

- Also Linux Based
- Manual Configuration
- Display All Configuration (Manual or Auto)
- Configuration looks different, but has all of the same options as setkey
- The wiki is the best source for more info

Strongswan

Use Strongswan for IKEv2 Go-to-Device

- Linux (of course, others too!)
- Well Documented
- Use it all the time
- Does everything
- http://wiki.strongswan.org/wiki/strongswan/IpsecConf

Racoon2

Another IKEv2 Implementation

- Linux (and others)
- Not a ton of documentation, but some
- Use it when you must
- Certificate support is not fully implemented! (No IKEv2 Interop)
- IOL Wiki https://tommy.iol.unh.edu/wiki/Racoon_Config_Help

Scripts

- Configuration Scripts exist for almost everything that has been tested.
- Depending on the device and if we are using Manual Keys or IKEv2, I usually have something to make it easier, check with me before despair.
- When testing a new device, or a device for the first time, save the config you used for every test!
- Also try to save the keys that were negotiated when testing IKEv2. This is the only way we'll be able to decrypt the packets.
- You'll thank yourself later.

Cryptography

Cryptography 401

- We **aren't** the NSA.
- But it's worth it to know the basics

Types of Protection:

Encryption Provides Confidentiality
Integrity Data Unmodified
Authentication Establishing Identity
Hash/Checksum/CRC Weak - no key needed! **Not IPsec!**

Cryptography 401 cont.

- Unfortunately, Integrity and Authenticity are often used incorrectly.
- For now, we'll worry about Encryption, and Authentication as both Authenticity and Integrity

Caesar Shift

YG OADVUSLK
ideas?

Algorithm called an Alphabetic Rotation

Okay, but how far?
Key is the Distance of rotation

What is the key?

Algorithm=Rot(ation) Key=18 $\Rightarrow$ *Rot-18*

So: $Y \rightarrow g$ $O \rightarrow w$, etc...
This is a form of **encryption**.