OpenVPN Customization Schema (Landscape Edition)

■ OpenVPN Customization Layers Overview

Config → Crypto → Provider → Source → System

Architecture Summary

■ Overview

OpenVPN is an open-source VPN system based on OpenSSL for cryptography. It can be customized from simple configuration tweaks to implementing your own ciphers, MACs, or even custom PSK algorithms. This document summarizes all the layers and options visually.

■■ Customization Layers

Layer	What You Control	Typical Customizations	Difficulty
Config Layer	Behavior via .conf files	Change cipher, auth, ports	■ Easy
Crypto Layer (EVP)	Algorithm selection	Use OpenSSL-provided ciphers/digests	■–■ Moderate
Provider Layer (OpenSSL)	Algorithm implementation	Custom cipher/MAC provider	■–■ Hard
Source Layer (OpenVPN Core)	TLS/PSK logic	Modify handshake, PSK routines	■ Expert
System Layer	Network integration	Firewall, routing, scripts	■ Moderate

■ Customization Paths

Depending on your goal, you can modify OpenVPN at different layers. The deeper you go, the more technical skill and code modification are required.

Goal	Approach	Where
Change encryption algorithm (AES→ChaCha20)	Update config only	Config Layer
Use stronger HMAC	Set 'auth SHA512'	Config Layer
Replace PSK/TLS-HMAC	Patch OpenVPN or use tls-crypt	Source/Crypto Layer
Add proprietary cipher	Implement OpenSSL provider	Provider Layer
Custom MAC or PSK	Implement OpenSSL MAC provider	Provider Layer

■ Key Components

Configuration: `/etc/openvpn/server.conf` – defines cipher/auth.
Client configs: `.ovpn` files – control client behavior and keys.
Providers: `/usr/lib/ossl-modules/` – OpenSSL custom modules.
Source files: `src/openvpn/ssl.c` – TLS and PSK logic.
Scripts: `/etc/openvpn/scripts/` – connection/disconnection hooks.

■ Summary

• Configuration tweaks → easiest and safest.
• Custom ciphers/MACs → use OpenSSL provider.
• Modify handshake logic → edit OpenVPN source.
• Custom client app → use OpenVPN3 SDK.

This landscape version gives a clearer overview of OpenVPN’s customizable architecture, making it easier to visualize how configuration, cryptography, and source layers connect.