

VPN Payload

The VPN payload is used for traditional systemwide VPNs based on L2TP, PPTP, and IPSec. This payload should not be confused with the Per-App VPN, described in [Per-App VPN Payload](#).

The VPN payload is designated by specifying `com.apple.vpn.managed` as the `PayloadType` value. In addition to the settings common to all payload types, the VPN payload defines the following keys:

Key	Type	Value
<code>UserDefinedName</code>	String	Optional. Description of the VPN connection displayed on the device.
<code>VPNTType</code>	String	Determines the settings available in the payload for this type of VPN connection. It can have one of the following values: • L2TP • PPTP • IPSec (Cisco) • IKEv2 (see IKEv2 Dictionary Keys) • AlwaysOn (see AlwaysOn Dictionary Keys) • VPN (solution uses a VPN plugin or <code>NetworkExtension</code>, so the <code>VPNSubType</code> key is required (see below)).
<code>VPNSubType</code>	String	Optional. If <code>VPNTType</code> is VPN, this field is required. If the configuration is targeted at a VPN solution that uses a VPN plugin, then this field contains the bundle identifier of the plugin. Here are some examples: • Cisco AnyConnect: <code>com.cisco.anyconnect.applevpn.plugin</code> • Juniper SSL: <code>net.juniper.sslvpn</code> • F5 SSL: <code>com.f5.F5-Edge-Client.vpnplugin</code> • SonicWALL Mobile Connect: <code>com.sonicwall.SonicWALL-SSLVPN.vpnplugin</code> • Aruba VIA: <code>com.arubanetworks.aruba-via.vpnplugin</code> If the configuration is targeted at a VPN solution that uses a <code>NetworkExtension</code> provider, then this field contains the bundle identifier of the app that contains the provider. Contact the VPN solution vendor for the value of the identifier. If <code>VPNTType</code> is IKEv2, then the <code>VPNSubType</code> field is optional and is reserved for future use. If it is specified, it must contain the empty string.
<code>ProviderBundleIdentifier</code>	String	Optional. If the <code>VPNSubType</code> field contains the bundle identifier of an app that contains multiple VPN providers of the same type (<code>app-proxy</code> or <code>packet-tunnel</code>), then this field is used to specify which provider to use for this configuration.

If `VPNType` is `VPN`, `IPSec`, or `IKEv2`, the following keys may be defined in the corresponding `VPN`, `IPSec`, or `IKEv2` dictionaries to configure VPN On Demand:

Key	Type	Value
<code>OnDemandEnabled</code>	Integer	1 if the VPN connection should be brought up on demand, else 0.
<code>OnDemandMatchDomainsAlways</code>	Array of Strings	<i>Deprecated.</i> A list of domain names. In versions of iOS prior to iOS 7, if the hostname ends with one of these domain names, the VPN is started automatically. In iOS 7 and later, if this key is present, the associated domain names are treated as though they were associated with the <code>OnDemandMatchDomainsOnRetry</code> key. This behavior can be overridden by <code>OnDemandRules</code> .
<code>OnDemandMatchDomainsNever</code>	Array of Strings	<i>Deprecated.</i> A list of domain names. If the hostname ends with one of these domain names, the VPN is <i>not</i> started automatically. This might be used to exclude a subdomain within an included domain. This behavior can be overridden by <code>OnDemandRules</code> . In iOS 7 and later, this key is deprecated (but still supported) in favor of <code>EvaluateConnection</code> actions in the <code>OnDemandRules</code> dictionaries.
<code>OnDemandMatchDomainsOnRetry</code>	Array of Strings	<i>Deprecated.</i> A list of domain names. If the hostname ends with one of these domain names, if a DNS query for that domain name fails, the VPN is started automatically. This behavior can be overridden by <code>OnDemandRules</code> . In iOS 7 and later, this key is deprecated (but still supported) in favor of <code>EvaluateConnection</code> actions in the <code>OnDemandRules</code> dictionaries.
<code>OnDemandRules</code>	Array of Dictionaries	Determines when and how an on-demand VPN should be used. See On Demand Rules Dictionary Keys for details.

If `VPNType` is not `AlwaysOn`, the following key may be defined:

Key	Type	Value
<code>VendorConfig</code>	Dictionary	A dictionary for configuration information specific to a given third-party VPN solution.

There are two possible dictionaries present at the top level, under the keys “`PPP`” and “`IPSec`”. The keys inside these two dictionaries are described below, along with the `VPNType` value under which the keys are used.

PPP Dictionary Keys

The following elements are for VPN payloads of type `PPP`.

Key	Type	Value
<code>AuthName</code>	String	The VPN account user name. Used for L2TP and PPTP.

Key	Type	Value
AuthPassword	String	Optional. Only visible if TokenCard is false. Used for L2TP and PPTP.
TokenCard	Boolean	Whether to use a token card such as an RSA SecurID card for connecting. Used for L2TP.
CommRemoteAddress	String	IP address or host name of VPN server. Used for L2TP and PPTP.
AuthEAPPlugins	Array	Only present if RSA SecurID is being used, in which case it has one entry, a string with value "EAP-RSA". Used for L2TP and PPTP.
AuthProtocol	Array	Only present if RSA SecurID is being used, in which case it has one entry, a string with value "EAP". Used for L2TP and PPTP.
CCMPPE40Enabled	Boolean	See discussion under CCPEnabled. Used for PPTP.
CCMPPE128Enabled	Boolean	See discussion under CCPEnabled. Used for PPTP.
CCPEnabled	Boolean	Enables encryption on the connection. If this key and CCMPPE40Enabled are true, represents automatic encryption level; if this key and CCMPPE128Enabled are true, represents maximum encryption level. If no encryption is used, then none of the CCP keys are true. Used for PPTP.

IPv4 Dictionary Keys

The following element is for VPN payloads of type L2TP or PPTP

Key	Type	Value
OverridePrimary	Integer	Specifies whether to send all traffic through the VPN interface. If 1, all network traffic is sent over VPN. Defaults to 0.

IPSec Dictionary Keys

The following elements are for VPN payloads of type IPSec.

Key	Type	Value
RemoteAddress	String	IP address or host name of the VPN server. Used for Cisco IPSec.
AuthenticationMethod	String	Either SharedSecret or Certificate. Used for L2TP and Cisco IPSec.
XAuthEnabled	Integer	1 if Xauth is on, 0 if it is off. Used for Cisco IPSec.
XAuthName	String	User name for VPN account. Used for Cisco IPSec.
XAuthPassword	String	Required for VPN account user authentication. Used for Cisco IPSec.
LocalIdentifier	String	Present only if AuthenticationMethod is SharedSecret. The name of the group to use. If Hybrid Authentication is used, the string must end with [hybrid]. Used for Cisco IPSec.
LocalIdentifierType	String	Present only if AuthenticationMethod is SharedSecret. The value is KeyID. Used for L2TP and Cisco IPSec.

Key	Type	Value
SharedSecret	Data	The shared secret for this VPN account. Only present if AuthenticationMethod is SharedSecret. Used for L2TP and Cisco IPSec.
PayloadCertificateUUID	String	The UUID of the certificate to use for the account credentials. Only present if AuthenticationMethod is Certificate. Used for Cisco IPSec.
PromptForVPNPIN	Boolean	Tells whether to prompt for a PIN when connecting. Used for Cisco IPSec.

On Demand Rules Dictionary Keys

The `OnDemandRules` key in a VPN payload is associated with an array of dictionaries that define the network match criteria that identify a particular network location.

In typical use, VPN On Demand matches the dictionaries in the `OnDemandRules` array against properties of your current network connection to determine whether domain-based rules should be used in determining whether to connect, then handles the connection as follows:

- If domain-based matching is enabled for a matching `OnDemandRules` dictionary, then for each dictionary in that dictionary's `EvaluateConnection` array, VPN On Demand compares the requested domain against the domains listed in the `Domains` array.
- If domain-based matching is not enabled, the specified behavior (usually `Connect`, `Disconnect`, or `Ignore`) is used if the dictionary otherwise matches.

Note

For backwards compatibility, VPN On Demand also allows you to specify the `Allow` action, in which case the domains to match are determined by arrays in the VPN payload itself (`OnDemandMatchDomainsAlways`, `OnDemandMatchDomainsOnRetry`, and `OnDemandMatchDomainsNever`). However, this is deprecated in iOS 7.

Whenever a network change is detected, the VPN On Demand service compares the newly connected network against the match network criteria specified in each dictionary (in order) to determine whether VPN On Demand should be allowed or not on the newly joined network. The matching criteria can include any of the following:

- DNS domain or DNS server settings (with wildcard matching)
- SSID
- Interface type
- reachable server detection

Dictionaries are checked sequentially, beginning with the first dictionary in the array. A dictionary matches the current network only if *all* of the specified policies in that dictionary match. You should always set a default behavior for unknown networks by specifying an action with no matching criteria as the last dictionary in the array.

If a dictionary matches the current network, a server probe is sent if a URL is specified in the profile. VPN then acts according to the policy defined in the dictionary (for example, allow VPNOnDemand, ignore VPNOnDemand, connect, or disconnect).

Note

Be sure to set a catch-all value. If you do not, the current default behavior is to allow the connection to occur, but this behavior is not guaranteed.

The OnDemandRules dictionaries can contain one or more of the following keys:

Key	Type	Value
Action	String	The action to take if this dictionary matches the current network. Possible values are: • Allow—<i>Deprecated</i>. Allow VPN On Demand to connect if triggered. • Connect—Unconditionally initiate a VPN connection on the next network attempt. • Disconnect—Tear down the VPN connection and do not reconnect on demand as long as this dictionary matches. • EvaluateConnection—Evaluate the ActionParameters array for each connection attempt. • Ignore—Leave any existing VPN connection up, but do not reconnect on demand as long as this dictionary matches.
ActionParameters	Array of Dictionaries	A dictionary that provides rules similar to the OnDemandRules dictionary, but evaluated on each connection instead of when the network changes. These dictionaries are evaluated in order, and the behavior is determined by the first dictionary that matches. The keys allowed in each dictionary are described in Keys in the ActionParameters dictionary . Note: This array is used only for dictionaries in which EvaluateConnection is the Action value.
DNSDomainMatch	Array of Strings	An array of domain names. This rule matches if any of the domain names in the specified list matches any domain in the device's search domains list. A wildcard '*' prefix is supported. For example, *.example.com matches against either mydomain.example.com or yourdomain.example.com.
DNSServerAddressMatch	Array of Strings	An array of IP addresses. This rule matches if any of the network's specified DNS servers match any entry in the array. Matching with a single wildcard is supported. For example, 17.* matches any DNS server in the class A 17 subnet.
InterfaceTypeMatch	String	An interface type. If specified, this rule matches only if the primary network interface hardware matches the specified type. Supported values are Ethernet, WiFi, and Cellular.

Key	Type	Value
SSIDMatch	Array of Strings	An array of SSIDs to match against the current network. If the network is not a Wi-Fi network or if the SSID does not appear in this array, the match fails. Omit this key and the corresponding array to match against any SSID.
URLStringProbe	String	A URL to probe. If this URL is successfully fetched (returning a 200 HTTP status code) without redirection, this rule matches.

The keys allowed in each `ActionParameters` dictionary are:

Key	Type	Value
Domains	Array of Strings	<i>Required.</i> The domains for which this evaluation applies.
DomainAction	String	<i>Required.</i> Defines the VPN behavior for the specified domains. Allowed values are: • ConnectIfNeeded—The specified domains should trigger a VPN connection attempt if domain name resolution fails, such as when the DNS server indicates that it cannot resolve the domain, responds with a redirection to a different server, or fails to respond (timeout). • NeverConnect—The specified domains will not trigger a VPN connection nor be accessible through an existing VPN connection.
RequiredDNSServers	Array of Strings	<i>Optional.</i> An array of IP addresses of DNS servers to be used for resolving the specified domains. These servers need not be part of the device's current network configuration. If these DNS servers are not reachable, a VPN connection is established in response. These DNS servers should be either internal DNS servers or trusted external DNS servers. Note: This key is valid only if the value of <code>DomainAction</code> is <code>ConnectIfNeeded</code> .
RequiredURLStringProbe	String	<i>Optional.</i> An HTTP or HTTPS (preferred) URL to probe, using a GET request. If no HTTP response code is received from the server, a VPN connection is established in response. Note: This key is valid only if the value of <code>DomainAction</code> is <code>ConnectIfNeeded</code> .

IKEv2 Dictionary Keys

If VPNTType is IKEv2, the following keys may be provided in a dictionary:

Key	Type	Value
RemoteAddress	String	Required. IP address or hostname of the VPN server.
LocalIdentifier	String	Required. Identifier of the IKEv2 client in one of the following formats: • FQDN • UserFQDN • Address • ASN1DN
RemoteIdentifier	String	Required. Remote identifier in one of the following formats: • FQDN • UserFQDN • Address • ASN1DN
AuthenticationMethod	String	Required. One of the following: • SharedSecret • Certificate • None To enable EAP-only authentication, the authentication method should be set to None and the ExtendedAuthEnabled key should be set to 1. If this key is set to None and the ExtendedAuthEnabled key is not set, the authentication configuration defaults to SharedSecret.
PayloadCertificateUUID	String	Optional. The UUID of the identity certificate used as the account credential. If the value of AuthenticationMethod is Certificate, this certificate is sent out for IKEv2 machine authentication. If extended authentication (EAP) is used, it is sent out for EAP-TLS authentication.
CertificateType	String	Optional. This key specifies the type of PayloadCertificateUUID used for IKEv2 machine authentication. Its value is one of the following: • RSA (Default) • ECDSA256 • ECDSA384 • ECDSA521 If this key is included, the ServerCertificateIssuerCommonName key is required.
SharedSecret	String	Optional. If AuthenticationMethod is SharedSecret, this value is used for IKE authentication.
ExtendedAuthEnabled	Integer	Optional. Set to 1 to enable EAP-only authentication (see AuthenticationMethod, above). Defaults to 0.
AuthName	String	Optional. Username used for authentication.
AuthPassword	String	Optional. Password used for authentication.

Key	Type	Value
DeadPeerDetectionRate	String	Optional. One of the following: • None (Disable) • Low (keepalive sent every 30 minutes) • Medium (keepalive sent every 10 minutes) • High (keepalive sent every 1 minute) Defaults to Medium.
ServerCertificateIssuer CommonName	String	Optional. Common Name of the server certificate issuer. If set, this field will cause IKE to send a certificate request based on this certificate issuer to the server. This key is required if both the CertificateType key is included and the ExtendedAuthEnabled key is set to 1.
ServerCertificate CommonName	String	Optional. Common Name of the server certificate. This name is used to validate the certificate sent by the IKE server. If not set, the Remote Identifier will be used to validate the certificate.
TLSMinimumVersion	String	Optional. The minimum TLS version to be used with EAP-TLS authentication. Value may be 1.0, 1.1, or 1.2. If no value is specified, the default minimum is 1.0. Availability: Available in iOS 11.0 and macOS 10.13 and later.
TLSMaximumVersion	String	Optional. The maximum TLS version to be used with EAP-TLS authentication. Value may be 1.0, 1.1, or 1.2. If no value is specified, the default maximum is 1.2. Availability: Available in iOS 11.0 and macOS 10.13 and later.
NATKeepAliveOffload Enable	Integer	Optional. Set to 1 to enable or 0 to disable NAT Keepalive offload for Always On VPN IKEv2 connections. Keepalive packets are sent by the device to maintain NAT mappings for IKEv2 connections that have a NAT on the path. Keepalive packets are sent at regular interval when the device is awake. If NATKeepAliveOffloadEnable is set to 1, Keepalive packets will be offloaded to hardware while the device is asleep. NAT Keepalive offload has an impact on the battery life since extra workload is added during sleep. The default interval for the Keepalive offload packets is 20 seconds over WiFi and 110 seconds over Cellular interface. The default NAT Keepalive works well on networks with small NAT mapping timeouts but imposes a potential battery impact. If a network is known to have larger NAT mapping timeouts, larger Keepalive intervals may be safely used to minimize battery impact. The Keepalive interval can be modified by setting the NATKeepAliveInterval key. Default value for NATKeepAliveOffloadEnable is 1.
NATKeepAliveInterval	Integer	Optional. NAT Keepalive interval for Always On VPN IKEv2 connections. This value controls the interval over which Keepalive offload packets are sent by the device. The minimum value is 20 seconds. If no key is specified, the default is 20 seconds over WiFi and 110 seconds over a Cellular interface.

Key	Type	Value
UseConfigurationAttributeInternalIPSubnet	Integer	Optional. If set to 1, negotiations should use IKEv2 Configuration Attribute INTERNAL_IP4_SUBNET and INTERNAL_IP6_SUBNET. Defaults to 0. Availability: Available in iOS 9.0 and later.
DisableMOBIKE	Integer	Optional. If set to 1, disables MOBIKE. Defaults to 0. Availability: Available in iOS 9.0 and later.
DisableRedirect	Integer	Optional. If set to 1, disables IKEv2 redirect. If not set, the IKEv2 connection would be redirected if a redirect request is received from the server. Defaults to 0. Availability: Available in iOS 9.0 and later.
EnablePFS	Integer	Optional. Set to 1 to enable Perfect Forward Secrecy (PFS) for IKEv2 Connections. Default is 0. Availability: Available in iOS 9.0 and later.
EnableCertificateRevocationCheck	Integer	Optional. Set to 1 to enable a certificate revocation check for IKEv2 connections. This is a best-effort revocation check; server response timeouts will not cause it to fail. Availability: Available in iOS 9.0 and later.
IKESecurityAssociationParameters	Dictionary	Optional. See table below. Applies to child Security Association unless ChildSecurityAssociationParameters is specified.
ChildSecurityAssociationParameters	Dictionary	Optional. See table below.

The IKESecurityAssociationParameters and ChildSecurityAssociationParameters dictionaries may contain the following keys:

Key	Type	Value
EncryptionAlgorithm	String	Optional. One of: • DES • 3DES • AES-128 • AES-256 (Default) • AES-128-GCM (16-octet ICV) • AES-256-GCM (16-octet ICV)
IntegrityAlgorithm	String	Optional. One of: • SHA1-96 • SHA1-160 • SHA2-256 (Default) • SHA2-384 • SHA2-512
DiffieHellmanGroup	Integer	Optional. One of: 1, 2, 5, 14 (Default), 15, 16, 17, 18, 19, 20, or 21.
LifeTimeInMinutes	Integer	Optional SA lifetime (rekey interval) in minutes. Valid values are 10 through 1440. Defaults to 1440 minutes.

DNS Dictionary Keys

If VPNTType is IKEv2, the following DNS keys may be provided:

Key	Type	Value
ServerAddresses	Array of Strings	Required. An array of DNS server IP address strings. These IP addresses can be a mixture of IPv4 and IPv6 addresses. Availability: Available in iOS 10.0 and later and macOS 10.12 and later.
SearchDomains	Array of Strings	Optional. A list of domain strings used to fully qualify single-label host names. Availability: Available in iOS 10.0 and later and macOS 10.12 and later.
DomainName	String	Optional. The primary domain of the tunnel. Availability: Available in iOS 10.0 and later and macOS 10.12 and later.
SupplementalMatchDomains	Array of Strings	Optional. A list of domain strings used to determine which DNS queries will use the DNS resolver settings contained in ServerAddresses. This key is used to create a split DNS configuration where only hosts in certain domains are resolved using the tunnel's DNS resolver. Hosts not in one of the domains in this list are resolved using the system's default resolver. If SupplementalMatchDomains contains the empty string it becomes the default domain. This is how a split-tunnel configuration can direct all DNS queries first to the VPN DNS servers before the primary DNS servers. If the VPN tunnel becomes the network's default route, the servers listed in ServerAddresses become the default resolver and the SupplementalMatchDomains list is ignored. Availability: Available in iOS 10.0 and later and macOS 10.12 and later.
SupplementalMatchDomainsNoSearch	Integer	Optional. Whether (0) or not (1) the domains in the SupplementalMatchDomains list should be appended to the resolver's list of search domains. Default is 0. Availability: Available in iOS 10.0 and later and macOS 10.12 and later.

Proxies Dictionary Keys

The Proxies dictionary may contain the following keys:

Key	Type	Value
ProxyAutoConfigEnable	Integer	Optional. Set to 1 to enable automatic proxy configuration. Defaults to 0.

Key	Type	Value
ProxyAutoConfigURLString	String	Optional. URL to the location of the proxy auto-configuration file. Used only when ProxyAutoConfigEnable is 1.
SupplementalMatchDomains	Array of Strings	Optional. If set, then only connections to hosts within one or more of the specified domains will use the proxy settings

If ProxyAutoConfigEnable is 0, the dictionary may also contain the following keys:

Key	Type	Value
HTTPEnable	Integer	Optional. Set to 1 to enable proxy for HTTP traffic. Defaults to 0.
HTTPProxy	String	Optional. The host name of the HTTP proxy.
HTTPPort	Integer	Optional. The port number of the HTTP proxy. This field is required if HTTPProxy is specified.
HTTPProxyUsername	String	Optional. The username used for authentication.
HTTPProxyPassword	String	Optional. The password used for authentication.
HTTPSEnable	Integer	Optional. Set to 1 to enable proxy for HTTPS traffic. Defaults to 0.
HTTPSProxy	String	Optional. The host name of the HTTPS proxy.
HTTPSPort	Integer	Optional. The port number of the HTTPS proxy. This field is required if HTTPSProxy is specified.

AlwaysOn Dictionary Keys

If VPNTType is AlwaysOn, the following keys may be provided in a dictionary:

Key	Type	Value
UIToggleEnabled	Integer	Optional. If set to 1, allows the user to disable this VPN configuration. Defaults to 0.
TunnelConfigurations	Array of Dictionaries	Required. See below.
ServiceExceptions	Array of Dictionaries	Optional. See below.
AllowCaptiveWebSheet	Integer	Optional. Set to 1 to allow traffic from Captive Web Sheet outside the VPN tunnel. Defaults to 0.
AllowAllCaptiveNetworkPlugins	Integer	Optional. Set to 1 to allow traffic from all Captive Networking apps outside the VPN tunnel to perform Captive network handling. Defaults to 0.

Key	Type	Value
AllowedCaptiveNetworkPlugins	Array of Dictionaries	Optional. Array of Captive Networking apps whose traffic will be allowed outside the VPN tunnel to perform Captive network handling. Used only when AllowAllCaptiveNetworkPlugins is 0. Each dictionary in the AllowedCaptiveNetworkPlugins array must contain a BundleIdentifier key of type string, the value of which is the app's bundle identifier. Captive Networking apps may require additional entitlements to operate in a captive environment.

Each dictionary in a TunnelConfigurations array may contain the following keys:

Key	Type	Value
ProtocolType	String	Must be IKEv2.
Interfaces	Array of Strings	Optional. Specify the interfaces to which this configuration applies. Valid values are Cellular and WiFi. Defaults to Cellular, WiFi.

In addition, all keys defined for the IKEv2 dictionary, such as RemoteAddress and LocalIdentifier may be present in a TunnelConfigurations dictionary.

Each dictionary in a ServiceExceptions array may contain the following keys:

Key	Type	Value
ServiceName	String	Required. The name of a system service which is exempt from Always On VPN. Must be one of: - VoiceMail - AirPrint - CellularServices (Available in iOS 11.3 and later.)
Action	String	Required. One of the following: - Allow - Drop

Per-App VPN Payload

The Per-App VPN payload is used for configuring add-on VPN software, and it works only on VPN services of type 'VPN'. It should not be confused with the standard VPN payload, described in [VPN Payload](#).

This payload is supported only in iOS 7.0 and later and macOS v10.9 and later.

The VPN payload is designated by specifying `com.apple.vpn.managed.applayer` as the `PayloadType` value. The Per-App VPN payload supports all of the keys described in [VPN Payload](#) plus the following additional keys:

Key	Type	Value
<code>VPNUUID</code>	String	A globally-unique identifier for this VPN configuration. This identifier is used to configure apps so that they use the Per-App VPN service for all of their network communication. See App-to-Per-App VPN Mapping .
<code>SafariDomains</code>	Array	This optional key is a special case of App-to-Per App VPN Mapping. It sets up the app mapping for Safari (Webkit) with a specific identifier and a designated requirement. The array contains strings, each of which is a domain that should trigger this VPN connection in Safari. The rule matching behavior is as follows: • Before being matched against a host, all leading and trailing dots are stripped from the domain string. For example, if the domain string is <code>".com"</code> the domain string used to match is <code>"com"</code>. • Each label in the domain string must match an entire label in the host string. For example, a domain of <code>"example.com"</code> matches <code>"www.example.com"</code>, but not <code>"foo.badexample.com"</code>. • Domain strings with only one label must match the entire host string. For example, a domain of <code>"com"</code> matches <code>"com"</code>, not <code>"www.example.com"</code>.
<code>OnDemandMatchAppEnabled</code>	Boolean	If <code>true</code>, the Per-App VPN connection starts automatically when apps linked to this Per-App VPN service initiate network communication. If <code>false</code>, the Per-App VPN connection must be started manually by the user before apps linked to this Per-App VPN service can initiate network communication. If this key is not present, the value of the <code>OnDemandEnabled</code> key is used to determine the status of Per-App VPN On Demand.
<code>ProviderType</code>	String	Optional. Either <code>packet-tunnel</code> or <code>app-proxy</code> . The default is <code>app-proxy</code> . If the value of this key is <code>app-proxy</code> , then the VPN service will tunnel traffic at the application layer. If the value of this key is <code>packet-tunnel</code> , then the VPN service will tunnel traffic at the IP layer.

App-to-Per-App VPN Mapping

The App-to-Per-App mapping payload is designated by specifying `com.apple.vpn.managed.appmapping` as the `PayloadType` value.

This payload is supported only in macOS 10.9 and later. It is not supported in iOS.

Key	Type	Value
<code>AppLayerVPNMapping</code>	Array of Dictionaries	An array of mapping dictionaries.

Each dictionary in the array must contain the following keys:

Key	Type	Value
<code>Identifier</code>	String	Required. The app's bundle ID.
<code>VPNUUID</code>	String	Required. The VPNUUID of the Per-App VPN defined in a Per-App VPN payload.
<code>DesignatedRequirement</code>	String	Required. The code signature designated requirement of the app that will use the per-app VPN.
<code>SigningIdentifier</code>	String	Required. The code signature signing identifier of the app that will use the per-app VPN.

Web Clip Payload

The Web Clip payload is designated by specifying `com.apple.webClip.managed` as the `PayloadType` value.

A Web Clip payload provides a web clipping on the user's home screen as though the user had saved a bookmark to the home screen.

In addition to the settings common to all payloads, this payload defines the following keys:

Key	Type	Value
<code>URL</code>	String	The URL that the Web Clip should open when clicked. The URL must begin with HTTP or HTTPS or it won't work.
<code>Label</code>	String	The name of the Web Clip as displayed on the Home screen.
<code>Icon</code>	Data	Optional. A PNG icon to be shown on the Home screen. Should be 59 x 60 pixels in size. If not specified, a white square will be shown.
<code>IsRemovable</code>	Boolean	Optional. If <code>false</code> , the web clip is not removable. Defaults to <code>true</code> . Not available in macOS.