

Verify The Address (VTA)

Configuration, Automation & Verification Guide

Salesforce Address Verification Documentation — 360 Degree Cloud

What this document covers

- What Verify The Address is and how it works
- Object & field-level configuration (Object Setup)
- Default country logic and priority rules for callouts
- Address credits, purchase & usage tracking
- Automation Setup and custom Flow-based automation
- VTM History audit trail and field-level data written on verification
- Address verification status values and their meaning
- Status label overrides using the Address_Status__c custom setting
- Object support and custom object enablement

Table of Contents

1. What is Verify The Address (VTA)?
2. Existing Verification Framework — VTM, VTP & VTA Enablement
3. Address Credits, Purchase & Usage Tracking
4. VTM Invoice — Adding Address Credits
5. Address Details on the Client Record
6. App Usage Dashboard — Credits & Status Sync
7. Address Setup — Object & Default Country Configuration
8. Default Country Resolution Logic
9. Automation Setup — Verify on Insert / Update
10. Custom Automation Using Flows
11. VTM History — Verification Audit Trail
12. Fields Written on Verification
13. Address Verification Status Values
14. Overriding Status Labels — Address_Status__c
15. Object Support & Custom Object Enablement

1. What is Verify The Address (VTA)?

Address Verification (VTA) validates addresses and classifies them based on deliverability, helping reduce failed deliveries and improve overall data accuracy. It is part of the same verification suite that also includes Verify The Email (VTM) and Verify The Phone (VTP), and follows a consistent configuration and automation pattern across all three.

Address Status Values

Every verified address is classified into one of the following five deliverability statuses:

- Deliverable
- Likely Deliverable
- Partially Deliverable
- Needs Correction
- Non-Deliverable

How It Works

- Address verification results are shown directly on Contact and Lead records once verification is complete.
- Address verification can also be enabled on other (custom) objects with the required configuration.
- A detailed, field-level verification history is available in the VTM History object for every verification performed.

Availability

- Address Verification currently supports US addresses only.
- Address Credits usage and status can be viewed live in the App Usage section.
- A dedicated Address Setup section allows easy selection of the object and field used for address verification.
- Automated address verification can be configured through the Address Automation section.
- Phone Line Status and Address Verification are seamlessly supported together within the VTA package.

2. Existing Verification Framework — VTM, VTP & VTA Enablement

Verification for Email (VTM), Phone (VTP) and Address (VTA) is controlled at the client-record level on the **Vtm_Details__c** object. Each verification type has its own enablement flag and, where applicable, its own provider key.

Figure 1: VTM Details — Verification Type Enablement

←

Salesforce Home

All

Fields

Relations

Vtm_Details__c(360 PDC Dev1 / a23R700000OCIrQIAX)

is vt

Edit

Delete

Clone

Field API Name	Label	Type	Value
Is_VTA_Enabled__c	Is VTA Enabled	boolean, required	true
Is_VTM_Enabled__c	Is VTM Enabled	boolean, required	true
Is_VTP_Enabled__c	Is VTP Enabled	boolean, required	true
VTM_Key__c	Email Provider Key	Key_Registration__c	a8pR7000003kmZOIAY
VTP_Key__c	Phone Provider Key	Key_Registration__c	a8pR7000000LeTtIAK
Relationship Name	Child Object	Field	Label
Histories	Vtm_Details__History	ParentId	VTM Client History

On the **Vtm_Details__c** record, verification is controlled by three boolean flags: **Is_VTA_Enabled__c** (Is VTA Enabled), **Is_VTM_Enabled__c** (Is VTM Enabled) and **Is_VTP_Enabled__c** (Is VTP Enabled) — all currently set to **true**. Email and Phone verification each have a dedicated provider key field (**VTM_Key__c** — Email Provider Key, and **VTP_Key__c** — Phone Provider Key). The client record also exposes a related **Histories** list (child relationship to **Vtm_Details__History** / VTM Client History), which captures the running audit trail of verification activity. **Note:** On the VTP Client record, VTA needs to be enabled the same way VTM and VTP already are.

3. Address Credits, Purchase & Usage Tracking

Address verification consumes credits, tracked directly on the **Vtm_Details__c** record alongside the Email and Phone credit fields, using the same structure and naming pattern.

Figure 2: VTM Details — Address Credit Fields

←

Salesforce Home

All

Fields

Relations

Vtm_Details__c(360 PDC Dev1 / a23R700000OCIrQIAX)

addr

Edit

Delete

Clone

Export

Field API Name	Label	Type	Value
Address_Api_Key__c	Address Api Key	string (200)	Alza5yDphRgpcR5si0aJ1Lg-VW1oaPOU9ZRfe-g
Address_Purchase_Total__c	Address Purchase Total	double (18, 0), calculated	110
Address_Status__c	Address Status	picklist (255), restricted	Trial
Purchase_Address_Credit__c	Purchase Address Credit	double (18, 0)	125
Remaining_Address_Credits__c	Remaining Address Credits	double (18, 0), calculated	110
Total_Used_Address_Credit__c	Total Used Address Credit	double (18, 0)	0
Relationship Name	Child Object	Field	Label

The client record stores dedicated address credit and configuration fields: **Address_Api_Key__c** (Address Api Key), **Address_Purchase_Total__c** (calculated), **Address_Status__c** (restricted picklist, e.g. "Trial"), **Purchase_Address_Credit__c**, **Remaining_Address_Credits__c** (calculated) and **Total_Used_Address_Credit__c**. This confirms the address-related setup fields exist at the **Vtm_Details__c** level in the same pattern already used for Email and Phone (API key + credit tracking).

4. VTM Invoice — Adding Address Credits

Address credits are added to a client the same way Phone and Email credits are added today — through a **VTM Invoice** record linked to the client's **Vtm Details**.

Figure 3: VTM Invoice — Address Purchase Total

0

be

VTM Invoice

dreamforce

[Printable View](#) | [Help](#) | [Feedback](#)

« Back to List: Environment Hub Members

VTM Invoice Detail

Edit

Clone

VTM Invoice Name	dreamforce	Vtm Details	Whatssync
Purchase Credit		Amount	
Date	27/8/2026	Phone Purchase Total	
		Address Purchase Total	200
		Phone Amount	
		Address Amount	
Created By	CSE.APACEMEA, 27/8/2026 3:01 pm		Last Modified By
			CSE.APACEMEA, 27/8/2026 3:01 pm

Edit

Clone

The VTM Invoice record already includes both **Phone Purchase Total** and **Address Purchase Total** fields (shown here with a value of **200**), alongside **Purchase Credit**, **Amount**, **Phone Amount** and **Address Amount**. VTA credits can be added per client using this same invoice structure by populating the Address Purchase Total / Address Amount fields on a new VTM Invoice linked to the relevant Vtm Details record.

5. Address Details on the Client Record

Figure 4: VTM Details — Address Details Section

Review

☐

Review By

Review Date

▼ Address Details

Address Api Key	AlzaSyDphRgpcR5sl0aJ1Lg-VW1oaPOU9ZRfe-g	Address Status	Trial
Address Purchase Total	110	Purchase Address Credit	125
Total Used Address Credit	0	Remaining Address Credits	110
Created By	360 SMS Integration User, 28/8/2025 5:33 pm		Last Modified By
			360 SMS Integration User, 27/8/2026 6:13 pm

Edit

Clone

Once configured, each client's Address (VTA) details are displayed in a dedicated “▼ Address Details” section on the Vtm_Details__c record layout — showing **Address Api Key**, **Address Status** (e.g. Trial), **Address Purchase Total**, **Purchase Address Credit**, **Total Used Address Credit** and **Remaining Address Credits**, along with Created By / Last Modified By audit fields. This mirrors how Email (VTM) and Phone (VTP) details are already surfaced on the client record.

6. App Usage Dashboard — Credits & Status Sync

Figure 5: App Usage — Address Credits & Status Info

APP USAGE					
Email Credits & Status Info		Phone Credits & Status Info		Address Credits & Status Info	
Verify The Email Status:	Trial	Verify The Phone Status:	Active	Verify The Address Status:	Trial
Total Purchased Credits:	550	Total Purchased Credits:	350	Total Purchased Credits:	110
Total Used Credits:	316	Total Used Credits:	191	Total Used Credits:	101
Total Remaining Credits:	234	Total Remaining Credits:	159	Total Remaining Credits:	9
Refresh					

The **App Usage** admin page presents credit and status information for all three verification types side by side — **Email Credits & Status Info**, **Phone Credits & Status Info** and **Address Credits & Status Info** — each showing Verification Status, Total Purchased Credits, Total Used Credits and Total Remaining Credits. Clicking the **Refresh** button on this page re-syncs the Salesforce org with the corresponding **VTM Client (Vtm_Details__c)** record, and the “Verify The Address” status and credit totals shown in the Address column are reflected here immediately after the sync completes.

7. Address Setup — Object & Default Country Configuration

Figure 6: Address Setup — Object, Address API Name & Default Country

ADDRESS SETUP			
Object	Address API Name	Default Country	Action
search		United States	Save
Object	Address API Name	Default Country	Action
Contact	1. Mailing Address	US-United States	Edit Delete

The **Address Setup** page lets an admin configure, per object, which field should be used for address verification and what the **Default Country** should be for that object (defaulting to “United States” in the entry form). Once saved, each configuration appears in the table below with **Object**, **Address API Name** and **Default Country** columns, and can be edited or deleted. In the example shown, the **Contact** object is configured to use **1. Mailing Address** with a default country of **US-United States**.

8. Default Country Resolution Logic

The Default Country configured in Address Setup acts only as a **fallback**. Whenever a country can be identified from the record's own address text, that detected value always takes priority over the configured default. If nothing is configured and nothing can be detected, the system falls back to **United States** as the default.

Object Setup Default Country	Country on Record	Country Code Sent in Callout
Not set / blank	Not detected	US (system default)
Set (e.g. India)	Detected (e.g. England / UK)	Record's country — record has priority
Not set / blank	Detected	Record's country — record has priority
Set (e.g. India)	Not detected / not passed	Object Setup default — setup has priority

In short: the record's address text is always checked first for a country match; the object-level Default Country is used only when no country can be resolved from the record itself.

9. Automation Setup — Verify on Insert / Update

Figure 7: Automation Setup — Address Verification Triggers

The screenshot shows the 'Automation Setup' section in the application. On the left is a sidebar with navigation links: 'App Usage', 'Object & Field Setup', 'Automation Setup' (highlighted), and 'General Settings'. The main content area has a header 'ADDRESS SETUP' with a note: '(Currently supported for contact and lead, For more objects please contact support.)'. Below this is a table with columns: 'Object', 'Verify on Address Insert', 'Verify on Address Update', 'Verify Both', and 'Action'. The 'Object' column has a dropdown menu currently showing 'Contact'. The 'Verify on Address Update' checkbox is checked. The 'Action' column has a 'Save' button. Above the table, there is a summary bar for the 'Contact' object showing a checkmark for 'Verify on Address Update' and buttons for 'Edit' and 'Delete'.

Automation Setup allows address verification to be triggered automatically whenever a record is **inserted**, **updated**, or **both**, without requiring manual action. For each configured object, admins can independently enable **Verify on Address Insert**, **Verify on Address Update** and **Verify Both**. As shown, this is currently supported for the **Contact** and **Lead** objects (“For more objects please contact support”); in the example, Contact has **Verify on Address Update** enabled.

10. Custom Automation Using Flows

In addition to the standard Automation Setup, users can build their own custom automation using **Salesforce Flows**, allowing fully custom entry conditions for when address verification should run. This is achieved by invoking the **global/invocable Apex action** exposed for verification (the same action historically used with Process Builder).

Steps to automate Address Verification using Flows

- 1 Create a record-triggered flow.
- 2 Select the object and set the entry criteria for the flow.
- 3 Create an action element in the flow.
- 4 Pass the parameters as shown in the figure below.
- 5 To verify the address, create a similar Apex Action as shown in the figure.
- 6 Save and activate the flow.

Figure 8: Flow — Verify Address Apex Action

Flow Element	Configuration
Trigger	Record-Triggered Flow — Run Immediately
Action	Verify Address (Verify_Address) — “Verify Address From Process Builder”, exposed as a global invocable Apex action

Input — Address API	e.g. MailingAddress
Input — Record Id	e.g. {Triggering Contact > Contact ID}

Sample verification result from this flow: **Is Address Verified** — Deliverable, **Address Verification Status** — Deliverable, **Formatted Address** — “123, Mahatma Gandhi Road, Indore, Madhya Pradesh 452001, India”. This confirms address verification can be triggered through Flows using global/invocable Apex methods, giving admins a no-code, condition-driven alternative to standard Automation Setup.

11. VTM History — Verification Audit Trail

Figure 9: Verify The Email/Phone History — Address Verification Record

Verify The Email/Phone History
V-0009

[Edit Layout](#) | [Printable View](#) | [Help for this Page](#)

Verify The Email/Phone History Detail

EditDeleteClone

Sharing (SU Only)

VTM Name	V-0009	Owner	Unnati Rastogi [Change]
Verification Type	Address	Error Code	
Account		Related Object Id	003ai0000u8ljZAAQ
Lead		Error Msg	
Contact	TestVT	Last Modified By	Unnati Rastogi, 8/31/2026, 10:24 PM
Created By	Unnati Rastogi, 8/31/2026, 10:24 PM		

▼ Address Details

Is Address Verified	Likely Deliverable	Address Verification Status	 Likely Deliverable
Formatted Address	B29, Block B, Sector 1, Noida, Uttar Pradesh 201301, India	Verification Details	Some components unconfirmed, but address mostly valid.

Every time an address verification runs — whether via the manual Save action, standard Automation Setup, or a custom Flow — a **VTM_History__c** (“Verify The Email/Phone History”) record is created. This provides a full audit trail per verification, including the related record link (Contact / Lead / Account and Related Object Id), the Verification Type (“Address”), and the address-specific outcome fields: Is Address Verified, Address Verification Status, Formatted Address and Verification Details.

12. Fields Written on Verification

When an address verification completes successfully, two sets of records are updated: the **source record** being verified, and a new **VTM_History__c** audit record.

12.1 Source Object Record (Lead / Contact / Custom Object)

Only a **single field** is updated back on the actual source record via Database.update():

- **Is_Address_Verified__c** → set to the mapped status (e.g. “Deliverable”, “Needs Correction”), lowercased if a custom setting override exists via **Address_Status__c**.

This is confirmed by the FLS/update permission check performed immediately before the DML operation (String[] sObjectFields = new String[] {‘Is_Address_Verified__c’};). Even though other fields may exist on the object, only this one field is written back.

12.2 VTM_History__c (“Verify History” Record)

Field	Value Populated
Formatted_Address__c	formattedAddress returned from the verification API response
Is_Address_Verified__c	finalStatus — the same mapped status written to the source record

tdc_tvte__Source__c	sourceContext, e.g. "Detail Page"
Verification_Type__c	Hardcoded to 'Address'
RecordTypeId	ADDRESS_RECORDTYPE_ID (the Address_Verification record type)
Verification_Details__c	finalInfo — the descriptive message from AddressVerdictInterpreter
Dynamic lookup field (e.g. Lead__c, Contact__c)	sObj.Id — set via objHistory.put(currentObjectName, sObj.Id) inside a try/catch
RelatedObjectId__c	sObj.Id — always set, acting as a fallback if the dynamic lookup assignment fails

13. Address Verification Status Values

The **Is_Address_Verified__c** field (and the corresponding **Verification_Details__c** description) resolves to one of five possible statuses, ranked from highest to lowest confidence:

Status 1: DELIVERABLE

Definition: Address verified at subpremise level and complete.

Is_Address_Verified__c	deliverable
Verification_Details__c	Address verified at subpremise level and complete.

- Address validated down to apartment / unit level
- All components confirmed (street, apartment, city, state, zip, country)
- No missing parts
- No guessed / inferred data

Status 2: LIKELY_DELIVERABLE

Definition: Address appears valid with minor issues but still has good confidence for delivery.

Is_Address_Verified__c	likely_deliverable
Verification_Details__c	Minor issues in subpremise confirmation.

- Apartment / unit number confirmed by API
- Some details (like postal code) couldn't be 100% verified
- Address is mostly correct and can likely be delivered
- Minor verification gaps exist

Status 3: PARTIALLY_DELIVERABLE

Definition: Address has significant quality issues with inferred or unconfirmed components, but may still be deliverable.

Is_Address_Verified__c	partially_deliverable
Verification_Details__c (variant 1)	Inferred and unconfirmed subpremise data.
Verification_Details__c (variant 2)	Address inferred slightly but confirmed at premise level.
Verification_Details__c (variant 3)	Nearby address inferred and unconfirmed.

Variant 1 — Inferred and unconfirmed subpremise data:

- Apartment / suite number was guessed by API (not confirmed)
- Some other components also unconfirmed
- Address might be correct but has uncertain parts
- Should be verified before using for critical shipments
- Delivery possible but risky

Variant 3 — Nearby address inferred and unconfirmed:

- Address found is NEARBY, not an exact match
- Some components were guessed / inferred
- Multiple levels of uncertainty
- Low-to-moderate confidence
- Should be corrected if possible

Status 4: NEEDS_CORRECTION

Definition: Address is incomplete or has missing key components. Must be corrected before use.

Is_Address_Verified__c	needs_correction
Verification_Details__c	Address missing key details but no inferred data.

Scenario A — Missing apartment / suite number:

- Apartment / suite number missing
- Required component not found in address
- Cannot verify at unit level
- User must add missing apartment / suite info

Scenario B — Missing key address component:

- Critical address component missing (house number, street name, city, or state)
- No guessed data, but incomplete
- Cannot confirm delivery without missing info

- Manual entry required

Status 5: NON_DELIVERABLE

Definition: The address could not be validated as deliverable. This is the lowest-confidence outcome, returned when the verification service is unable to confirm the address as a real, deliverable location.

Is_Address_Verified__c	non_deliverable (default fallback status)
-------------------------------	---

This is the default / fallback classification used whenever a verdict cannot confidently be mapped to Deliverable, Likely Deliverable, Partially Deliverable, or Needs Correction — for example, when the address is missing entirely or the verification service cannot locate it at all.

14. Overriding Status Labels — Address_Status__c

A custom setting, **Address_Status__c**, allows admins to override or relabel the status value before it is written to **Is_Address_Verified__c**. One custom setting record is created per status label: **Deliverable**, **Likely Deliverable**, **Non Deliverable**, **Needs Correction** and **Partially Deliverable**.

How the override is resolved

```
Address_Status__c objStatus = Address_Status__c.getInstance(addStatus);
string finalStatus = (objStatus != null && String.isNotBlank(objStatus.value__c)) ?
objStatus.value__c.toLowerCase() : addStatus;
```

- If a matching custom setting record exists for the status, its **value__c** is used (lowercased) as the final status label.
- If **no matching record exists**, the system falls back to using the raw **addStatus** as-is (not even lowercased, since that fallback path only triggers through **objStatus.value__c**).
- This configuration is **not strictly required** — the feature works out-of-the-box with default status values. It is only needed if admins want to display custom / relabeled status text (e.g. “Deliverable” → “Valid Address”) without any code changes.

15. Object Support & Custom Object Enablement

Address Verification is supported out-of-the-box for the standard **Lead** and **Contact** objects. This applies consistently across Object Setup, Automation Setup, and Flow-based automation.

Custom Object Support: For any custom object, additional configuration is required before address verification can work on it. This includes:

- Creating the necessary Visualforce (VF) pages for that object.
- Adding the required custom fields (e.g. the Address field(s) and an **Is_Address_Verified__c**-equivalent field).
- Completing all other related setup (Object Setup entry, FLS/CRUD access, and automation configuration) before verification will function correctly on that object.

End of Document — Verify The Address (VTA) Configuration, Automation & Verification Guide.