

Phase 7: Integration & External Access – Must-Have Documentation

Goal: Enable Salesforce to securely communicate with external systems, exchange data, and automate processes for the Telecom Service Hub project.

1. Named Credentials

Purpose: Store external API authentication securely for callouts. **Steps:**

1. Setup → **Named Credentials** → **New Named Credential**

Fill in: Label: TelecomAPI

URL: https://external-service.example.com

Identity Type: Named Principal

Authentication: Password or OAuth


Impact: Makes external API calls secure and easy to manage.

Named Credentials:

The screenshot shows the Salesforce 'Named Credentials' configuration page. At the top, there's a blue header with the Salesforce logo, a breadcrumb 'SETUP > NAMED CREDENTIALS', and the title 'TelecomAPI'. To the right of the title are 'Edit' and 'Delete' buttons. Below the header, the configuration details are organized into sections:

- Label:** TelecomAPI
- Name:** TelecomAPI
- URL:** https://external-service.example.com
- Enabled for Callouts:** A toggle switch that is currently turned on (indicated by a checkmark).
- Authentication:** A section with two fields:
 - External Credential:** Telecom_Auth
 - Client Certificate:** (This field is empty)
- Callout Options:** A section with three checkboxes:
 - Generate Authorization Header:** (unchecked)
 - Allow Formulas in HTTP Header:** (checked)
 - Allow Formulas in HTTP Body:** (unchecked)

SETUP > NAMED CREDENTIALS



Telecom_Auth

EditDelete

Label

Telecom_Auth


Name

Telecom_Auth

Authentication Protocol

Custom

Managed Package Access


Created By Namespace 

Related Named Credentials

Label	Name	URL
TelecomAPI	TelecomAPI	https://external-service.example.com

Principals

New

Se...	Parameter Name	A...	Authentication Status	Actions
1	NamedPrincipal	2	Unknown	

Developer Console (Apex code)

Developer Console - Google Chrome

orgfarm-11cac0a6a9-dev-ed.develop.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage

File Edit Debug Test Workspace Help < >

Log executeAnonymous @9/26/2025, 8:32:03 PM

Execution Log

Timestamp	Event	Details
20:32:03:002	USER_INFO	[EXTERNAL] 005gL0000087PrW savitrinikoparde116@agentforce.com (GMT-07:00) Pacific Daylight Time (America/Los_An...
20:32:03:002	EXECUTION_ST...	
20:32:03:002	CODE_UNIT_ST...	[EXTERNAL] execute_anonymous_apex
20:32:03:002	VARIABLE_SCO...	[6] http System.Http true false
20:32:03:002	VARIABLE_SCO...	[1] req System.HttpRequest true false
20:32:03:002	VARIABLE_SCO...	[7] res System.HttpResponse true false
20:32:03:002	HEAP_ALLOCATE	[95] Bytes:3
20:32:03:002	HEAP_ALLOCATE	[100] Bytes:152
20:32:03:002	HEAP_ALLOCATE	[417] Bytes:408
20:32:03:002	HEAP_ALLOCATE	[430] Bytes:408
20:32:03:002	HEAP_ALLOCATE	[317] Bytes:6
20:32:03:002	HEAP_ALLOCATE	[EXTERNAL] Bytes:15
20:32:03:002	STATEMENT_EX...	[1]
20:32:03:002	STATEMENT_EX...	[1]

☐ This Frame ☐ Executable ☐ Debug Only ☐ Filter

Logs Tests Checkpoints Query Editor View State Progress Problems

User	Application	Operation	Time	Status	Read	Size
savitri koparde	Unknown	/services/data/v64...	9/26/2025, 8:32:0...	Success		8.5 KB

2. Remote Site Settings

Purpose

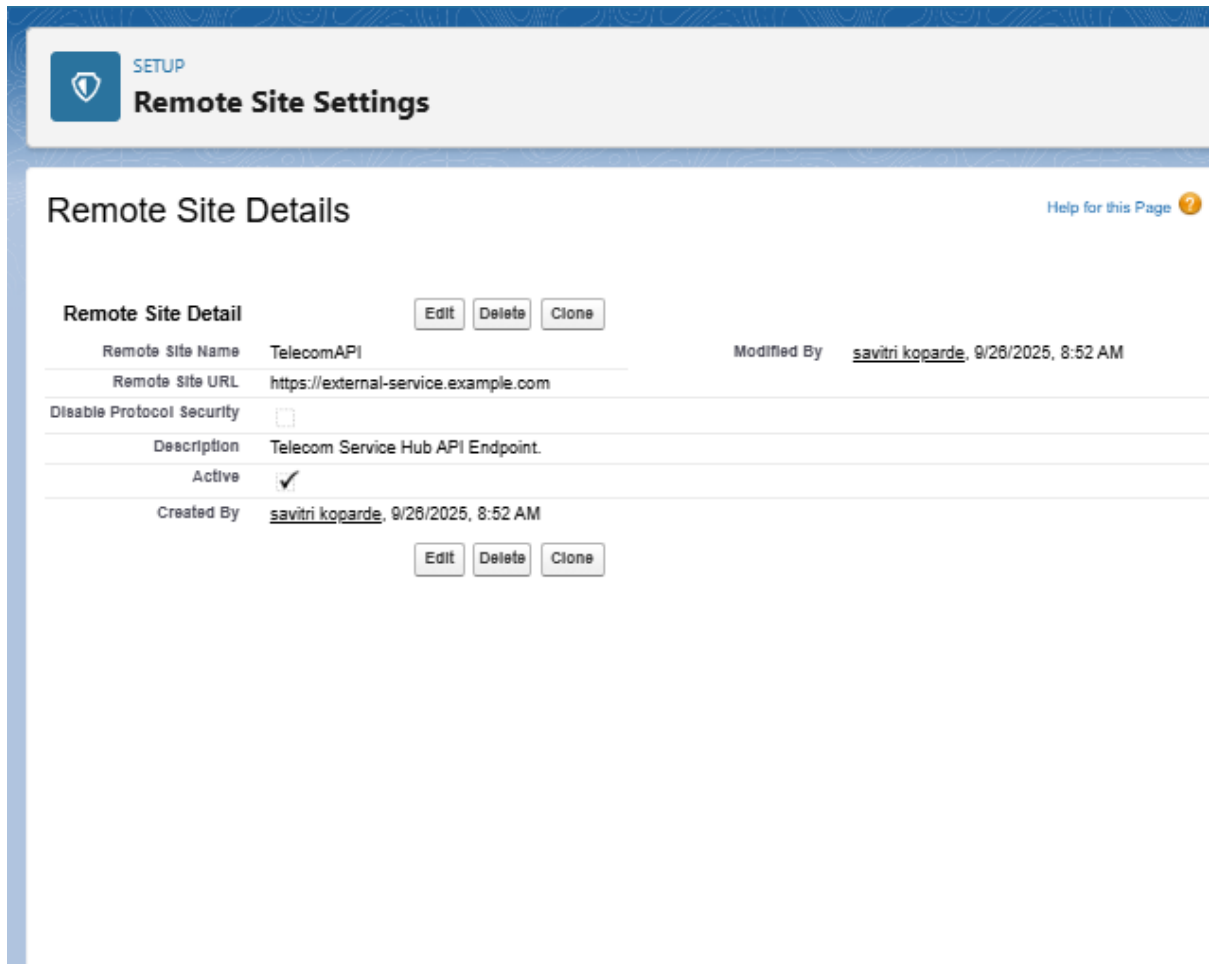
If using legacy authentication (without Named Credentials), Salesforce requires whitelisting the external API domain.

Steps

1. Go to **Setup → Remote Site Settings → New Remote Site**.
2. Enter:
 - **Remote Site Name:** TelecomAPI
 - **Remote Site URL:** https://external-service.example.com
3. Save

Impact

Ensures Salesforce allows outbound calls to the external API.



The screenshot displays the 'Remote Site Settings' page in Salesforce. At the top, there is a header with the 'SETUP' icon and the title 'Remote Site Settings'. Below this, the 'Remote Site Details' section is visible, featuring a table with the following information:

Remote Site Detail		Edit	Delete	Clone
Remote Site Name	TelecomAPI	Modified By <u>savitri koparde</u> , 9/26/2025, 8:52 AM		
Remote Site URL	https://external-service.example.com			
Disable Protocol Security	<input type="checkbox"/>			
Description	Telecom Service Hub API Endpoint.			
Active	<input checked="" type="checkbox"/>			
Created By	<u>savitri koparde</u> , 9/26/2025, 8:52 AM			

At the bottom of the table, there are three buttons: 'Edit', 'Delete', and 'Clone'.

3. Sample REST Callout (Apex Integration)

Purpose

Demonstrates how Salesforce can fetch data from an external system.

✂ Code Example

```
public class TelecomAPIService {  
    public static void fetchCustomerData() {  
        Http http = new Http();  
        HttpRequest request = new HttpRequest();  
        // Use Named Credential  
        request.setEndpoint('callout:TelecomAPI/customers');  
        request.setMethod('GET');  
  
        HttpResponse response = http.send(request);  
  
        if (response.getStatusCode() == 200) {  
            System.debug('Response: ' + response.getBody());  
        } else {  
            System.debug('Error: ' + response.getStatus());  
        }  
    }  
}
```

} Impact

- Shows Salesforce making a secure outbound call.
- This can later be extended to fetch Service Requests, Plans, or SIM status.

Developer Console - Google Chrome

orgfarm-11cac0a6a9-dev-ed.develop.my.salesforce.com/_ui/common/apex/debug/ApexCSIPage

Log executeAnonymous @9/26/2025, 9:29:20 PM

Code Coverage: None API Version: 64

```

1 public class TelecomAPIService {
2     public static void fetchCustomerData() {
3         // Step 1: Create HTTP instance
4         Http http = new Http();
5
6         // Step 2: Prepare the request
7         HttpRequest request = new HttpRequest();
8
9         // IMPORTANT: "TelecomAPI" must match your Named Credential Label/N
10        request.setEndpoint('callout:TelecomAPI/customers');
11        request.setMethod('GET');
12
13        // Step 3: Send the request
14        HttpResponse response = http.send(request);
15    }

```

ogs Tests Checkpoints Query Editor View State Progress Problems

ier	Application	Operation	Time	Status	Read	Size
vitri koparde	Unknown	/services/data/v64...	9/26/2025, 9:29:2...	Success		4.69 KB

By completing this phase, our **Telecom Service Hub** can:

- Securely connect to external APIs.
- Demonstrate real-time data exchange.
- Keep credentials secure with **Named Credentials**.