

Access Setup for Google Cloud and Google Analytics 4 (GA4) Integration

Access Setup for Google Cloud & Google Analytics 4 (GA4)

What the Client Needs to Do (One Time)

The client only needs to do **one thing** — add our email to their GA4 properties.

1. Go to [Google Analytics](#)
2. Click **Admin** (gear icon, bottom left)
3. Under the **Account** column → click **Account Access Management**
4. Click **+** in the top right
5. Fill in the following:

Field	Value
Email	[YOUR TEAM EMAIL]
Role	Editor/admin

6. Click **Add**
7. Repeat for each property they want to connect



That's all the client needs to do. No Google Cloud access, no JSON files, nothing else.

What We Do Afterward (One Time Setup Per Client)

Step 1 — Enable the GA4 API in Your Google Cloud Project

1. Go to [Google Cloud Console](#)
2. Switch to your Google Cloud project
3. Go to **APIs & Services → Library**
4. Search **"Google Analytics Data API"** → Enable it



This only needs to be done once per Google Cloud project, not once per client.

Step 2 — Create OAuth App (One Time Ever)

1. Go to **APIs & Services** → **OAuth consent screen**
2. Click **Get Started** → fill in app name and support email → select **External** → Save
3. Go to **Audience** → click **Publish App** → Confirm
4. Go to **Clients** → **Create OAuth Client** → **Desktop App** → Download JSON
5. Save the file as `oauth_client.json`



This only needs to be done once. The same OAuth app works for all clients.

Test Script

Step 1 — Run Login Once

Run this once — a browser will open, sign in with your Google account that the client gave access to:

python

```
1 from google_auth_oauthlib.flow import InstalledAppFlow
2 import pickle
3
4 flow = InstalledAppFlow.from_client_secrets_file(
5     "/path/to/oauth_client.json",
6     scopes=["https://www.googleapis.com/auth/analytics.readonly"]
7 )
8
9 creds = flow.run_local_server(port=8090)
10
11 with open("ga4_creds.pickle", "wb") as f:
12     pickle.dump(creds, f)
13
14 print("✅ Credentials saved!")
15 print("Refresh token:", creds.refresh_token[:20], "...")
```



This only needs to be run once per Google account. The pickle file saves credentials permanently.

Run this to confirm the connection is working for all client properties:

python

```
1 import pickle
2 import requests
3 from google.auth.transport.requests import Request
4
5 # Load and auto-refresh credentials
6 with open("ga4_creds.pickle", "rb") as f:
7     creds = pickle.load(f)
8
```

```

9  if creds.expired and creds.refresh_token:
10     creds.refresh(Request())
11     with open("ga4_creds.pickle", "wb") as f:
12         pickle.dump(creds, f)
13
14  def ga4_report(property_id, metrics, start_date="30daysAgo",
15                 end_date="today"):
16     response = requests.post(
17         f"https://analyticsdata.googleapis.com/v1beta/properties/{property_id}:runReport",
18         headers={"Authorization": f"Bearer {creds.token}", "Content-Type": "application/json"},
19         json={
20             "dateRanges": [{"startDate": start_date, "endDate": end_date}],
21             "metrics": [{"name": m} for m in metrics]
22         }
23     )
24     return response.json()
25
26  # Add client property IDs here
27  properties = {
28      "Property Name 1": "PROPERTY_ID_1",
29      "Property Name 2": "PROPERTY_ID_2",
30  }
31
32  for name, property_id in properties.items():
33      try:
34          result = ga4_report(property_id, ["activeUsers", "sessions"])
35          rows = result.get("rows", [])
36          if rows:
37              users = rows[0]["metricValues"][0]["value"]
38              sessions = rows[0]["metricValues"][1]["value"]
39              print(f"✅ {name}: {users} active users, {sessions} sessions")
40          else:
41              print(f"✅ {name}: connected but no data")
42      except Exception as e:
43          print(f"❌ {name}: {e}")

```

How to Find Property IDs

Run this once to list all properties your account has access to:

python

```

1  from google.analytics.admin_v1alpha import AnalyticsAdminServiceClient
2  import pickle
3
4  with open("ga4_creds.pickle", "rb") as f:
5      creds = pickle.load(f)
6
7  admin_client = AnalyticsAdminServiceClient(credentials=creds)
8
9  for account in admin_client.list_account_summaries():
10     print("Account:", account.display_name)
11     for prop in account.property_summaries:
12         print(" → Property:", prop.display_name, "|", prop.property)

```

How the Token Works

Token	Expires	What Happens
Access token	1 hour	Auto-refreshed silently using refresh token
Refresh token	Never (app is published)	Permanent — login never needed again

Troubleshooting

Error	Cause	Fix
403 Permission Denied	Email not added to GA4 property	Ask client to add email with Viewer role
403 Insufficient scope	Wrong OAuth scope	Re-run login flow with correct scope
Port already in use	Port busy	Change port number in <code>run_local_server()</code>
File not found	Wrong path to JSON or pickle	Check file paths

Summary

Who	What	When
Client	Add our email as Viewer in GA4	Once per client
Us	Create OAuth app	Once ever
Us	Run login flow	Once per Google account
Code	Auto-refreshes token + pulls data	Every run

Everyone	Nothing	Never again
-----------------	---------	-------------