CONNECT YOUR SOLUTION

Okta

Connect your Okta organisation to REVIEWIT: a dedicated service app, authenticated by a key pair, limited to read scopes.

VERSION
1.1 · November 2026

DURATION
Thirty minutes, in four steps

DISTRIBUTION
Confidential · REVIEWIT customers

WHAT YOU NEED

01 **An Okta super administrator**: only they can grant Okta API scopes and assign administrator roles to an application.

02 **A REVIEWIT account** with the subscription administrator role, to open the Subscription Manager.

03 **The time stated**, in the order of this guide: first the credential at Okta, then the REVIEWIT wizard, in four steps.

THE WIZARD, IN FOUR STEPS

STEP 1	STEP 2	STEP 3	STEP 4
Solution	**Prepare**	**Credential**	**Verification**
Pick Okta from your subscription's catalog.	Tick what REVIEWIT must collect; the wizard derives the rights to grant.	Enter what Okta handed you; REVIEWIT verifies it before saving.	One verdict per resource: what is covered, what remains to be granted.

IN THIS GUIDE

1 How REVIEWIT connects to Okta

2 What REVIEWIT will read, and the rights it takes

3 Create the credential at Okta

4 What you will enter in the wizard

5 The connection wizard, in four steps

6 What REVIEWIT receives, and what it never receives

7 After the connection

8 Troubleshooting and checklist

01 How REVIEWIT connects to Okta

REVIEWIT reviews **access**: who can reach what, and whether it is still justified. To do so it reads, **read-only**, the inventory of your resources and their rights at Okta, from your hosting region. It never writes on your side, never reads any content, and keeps the credential you hand it in a vault that only the collector opens.

ON YOUR SIDE
Okta
You create a technical credential, limited to the read rights the wizard lists.

→

YOUR REGION
REVIEWIT collector
Verifies the credential with a real call, then reads the inventory at the frequency of your collection plans.

→

YOUR CONSOLES
Workspace Manager
Campaigns, reviews and reports: never a direct call to Okta.

GUIDING PRINCIPLE

REVIEWIT believes only what it has observed by calling: a right granted but not verified stays "unverified" until the next health pass. What a connection cannot collect is stated **resource by resource**; a resource that cannot be collected can still be imported as a report.

SUPPORT

All Okta organisations, on the `okta.com`, `oktapreview.com` and `okta-emea.com` domains. The service app belongs to no one: it survives staff departures. Group owners depend on an Okta feature: where it is missing, the pass says so and groups are reported without an owner. Reading the scopes granted to applications depends on rights that Okta does not delegate read-only; when it is refused, the pass says so and still reports the consents given by people.

02 What REVIEWIT will read, and the rights it takes

The wizard computes the exact list of rights from the resources you tick at the **Prepare** step, and hands it to you ready to copy ("Copy the plan"). The table below gives the rights of the most commonly retained resources; your plan may hold fewer.

RIGHT OR GESTURE	RESOURCES IT OPENS
`okta.users.read`	Users, closed accounts, never activated accounts; basis for application assignments, administrator roles and OAuth consents
`okta.groups.read`	Groups and their owners, group members, group rules, group assignments to applications; administrator roles held by groups
`okta.apps.read`	Applications, assignments of people and groups; administrator roles held by service apps
`okta.roles.read`	Administrator role assignments — of people, groups and service apps —, custom administrator roles
`okta.apiTokens.read`	API tokens and their holder
`okta.appGrants.read`	OAuth consents

ADMINISTRATION GESTURES

RIGHT OR GESTURE	RESOURCES IT OPENS
"Read-only Administrator" role	All resources: without an administrator role, Okta issues the token but refuses every read
Custom role "View roles, resources, and admin assignments"	Administrator role assignments, custom roles

03 Create the credential at Okta

A super administrator creates, in the Okta Admin Console, **a service app dedicated to REVIEWIT**. It authenticates with a key pair: Okta keeps the public key, REVIEWIT receives the private key. No user signs in; no password or personal token is shared.

01 **Open Applications › Applications › Create App Integration, choose API Services**
Name it, for example "REVIEWIT Collect", and keep an Okta-generated client ID.

02 **General › Client Credentials › Edit: choose Public key / Private key, then save**
By default the app expects a client secret, which Okta refuses for its API scopes. Until this choice is saved, verification fails with `invalid_client`.

03 **Section Public keys › Edit › Add key › Generate new key**
Copy the **private key in PEM format** into a `.pem` file: Okta shows it only once. Finish, then save the section; the key must appear "Active".

04 **Section General Settings › Edit: clear Require Demonstrating Proof of Possession (DPoP) header in token requests**
The box is ticked by default. REVIEWIT does not send that proof: left ticked, verification says so and stops.

05 **Okta API Scopes tab: grant the scopes of the plan**
The six read scopes above, and no other. The REVIEWIT wizard lists them at the Prepare step.

06 **Admin roles › Edit assignments tab: assign Read-only Administrator**
Okta asks for a step-up authentication when saving.

07 **For administrator roles: create a custom role and its resource set, then assign them**
Security › **Administrators** › **Roles** › **Create new role** with the single permission "View roles, resources, and admin assignments"; **Resources** › **Create new resource set** on "Identity and Access Management", all resources. In the assignment, choose the resource set before the role.

08 **Note the app's Client ID and the organisation's domain**
The Client ID starts with `0oa`. The domain is the organisation's, without the console's `-admin` suffix.

READ-ONLY, AND CHECKABLE

The app only receives `…read` scopes: REVIEWIT cannot change anything in Okta. At each token exchange, Okta names the scopes it grants; that observed list is what the wizard shows.

04 What you will enter in the wizard

The **Credential** step only shows the fields Okta declares to REVIEWIT. The connection name is the one you will read in tables and alerts.

WIZARD FIELD	VALUE
Connection name	Free text — for example "Okta · <your organisation>".
Authentication method	**API service app.** The API token only serves the people directory: no collection accepts it.
Okta org domain	The host of the organisation, without `https://` — for example `acme.okta.com`. It is the external identifier of the connection.
Client ID	The service app's, noted at step 8.
Private key (.pem)	The `.pem` file saved at step 3.

A PRIVATE KEY IS A SECRET

Uploaded once, it is stored in the vault and referred to by reference; no screen shows it again. Delete the `.pem` file from your workstation after verification.

05 The connection wizard, in four steps

In the Subscription Manager, open **Data sources** › **Connections**, then **Connect a solution** (or **Connect an instance** on the solution's tile if it is already present). Nothing is saved before Okta answers.

STEP 1 Solution

The catalog lists only what is not connected yet. A "probe available" solution is verified by a real call; a "declarable" one is recorded on your word, at the level you choose.

STEP 2 Prepare

Tick the resources to collect: the list of rights to grant follows. Copy the plan, grant it at Okta, then come back. The wizard never applies that plan itself.

STEP 3 Credential

Name the connection and fill in the fields. **Verify the connection** calls Okta; **Save without verifying** keeps the connection inactive, to verify later.

Okta types its identity (the organisation's domain): it is locked once the connection is created.

STEP 4 Verification

Okta answered: the wizard shows the identity it read and, resource by resource, a verdict. **Save and activate** puts the connection in service; **Save inactive** keeps it for later.

Verification asks Okta for a token, records the granted scopes, then reads a group: a refusal on that read points to a missing administrator role (section 8).

06 What REVIEWIT receives, and what it never receives

RECEIVED FROM OKTA	USE AT REVIEWIT
`Organisation domain`	The identity of the connection.
`Users · groups · members · rules`	Who exists, in which state, and who belongs to what.
`Applications · assignments`	Which person and which group access which application.
`Group owners`	Who answers for each group, when your organisation declares them.
`Administrator roles · custom roles`	Who administers Okta — person, group or service app — and with which permissions.
`API tokens · OAuth consents`	Technical credentials and their holder, the consents given — never the value of a token.

NEVER RECEIVED, NEVER REQUESTED

No user password
The credential is technical, created for REVIEWIT; no person's account is used.

No content
No message, file or e-mail: REVIEWIT reads lists of objects and rights, never what they contain.

No write
The rights requested are read rights; remediation, where it exists, is a separate module, enabled separately.

No secret shown again
The secret is entered once, stored in the vault and referred to by reference; no screen or log shows it again.

07 After the connection

The connection is managed from **Data sources › Connections**. A health pass replays the verification **every 24 hours**; "Verify all connections" runs it on demand. Each tile states how many resources are ready, to configure or unverified.

A right is missing	The resource stays "unverified" and the wizard says which one. Grant it at Okta, then "Check rights".

Renew the secret	Add a **new key** on the service app (General › Public keys › Add key), upload its private key in "Connections › Add a credential", verify, then deactivate the old key in Okta. The Client ID, the scopes and the roles do not change.
A resource that cannot be collected	Import its report from **Data sources** › **Collect** › **Choose a report**: the exact CSV contract downloads from the drawer.
Disconnect	Deactivate the connection; then revoke the credential at Okta. Collection plans that relied on it stop.

08 Troubleshooting

WHAT YOU SEE	CAUSE	WHAT TO DO
`invalid_client` at verification	The app still expects a client secret, or the uploaded key is not the one registered on the app.	Save "Public key / Private key" in the client credentials; check the Client ID; if needed generate a new key.
"The API service app requires proof of possession (DPoP)"	The DPoP box is still ticked.	Clear it in General › General Settings, then verify again.
`E0000006` "You do not have permission"	The scopes are granted but the app holds no administrator role.	Assign "Read-only Administrator" in the Admin roles tab.
Administrator roles stay unreadable	The read-only role does not read roles.	Assign the custom role of step 7, on its resource set.
Notice "application scopes not readable"	Okta refuses that read to a read-only app.	Nothing to do: the consents given by people are reported; the notice stays in the pass report.
Notice "group owners not available"	Your Okta organisation does not have the group owners feature (a group's page has no Owners tab).	Nothing to do on the REVIEWIT side: groups are reported without an owner. Enable the feature in Okta to see them at the next pass.
The domain is refused	It carries `https://`, a path, or the console's `-admin` suffix.	Type the bare host of the organisation, for example `acme.okta.com`.

CHECKLIST BEFORE OPENING THE WIZARD — 6 items

- [ ] I have an Okta super administrator.
- [ ] The service app exists, set to "Public key / Private key", with an **Active** key.
- [ ] The DPoP box is cleared.
- [ ] The six read scopes are granted, and only those.
- [ ] The app holds "Read-only Administrator", and the custom role if I collect administrator roles.
- [ ] I have the `.pem` file, the Client ID and the organisation's domain, and I open the Subscription Manager as a subscription administrator.

This guide is provided to REVIEWIT customers to connect their solutions. Menu paths are those of the vendors' consoles at publication date; they may change. In case of discrepancy, the REVIEWIT wizard prevails: it shows the exact plan computed for your subscription.