Connecting To Okta Using OAuth 2.0

About Okta

Okta is an emerging platform that manages online identities for enterprises. Integration with AtScale will provide you with:

- A single source of truth for user's information
- Instantaneous user disablement from the Okta interface
- Entitlement grants from within Okta

When To Leverage AtScale's Okta Integration

AtScale recommends using Okta when your organization leverages their platform for all of their applications in place of an LDAP server. While there are different mechanisms to integrate with Okta, you can leverage their LDAP server. AtScale provides integration with the OAuth 2.0 mechanism for middleware.

Overview

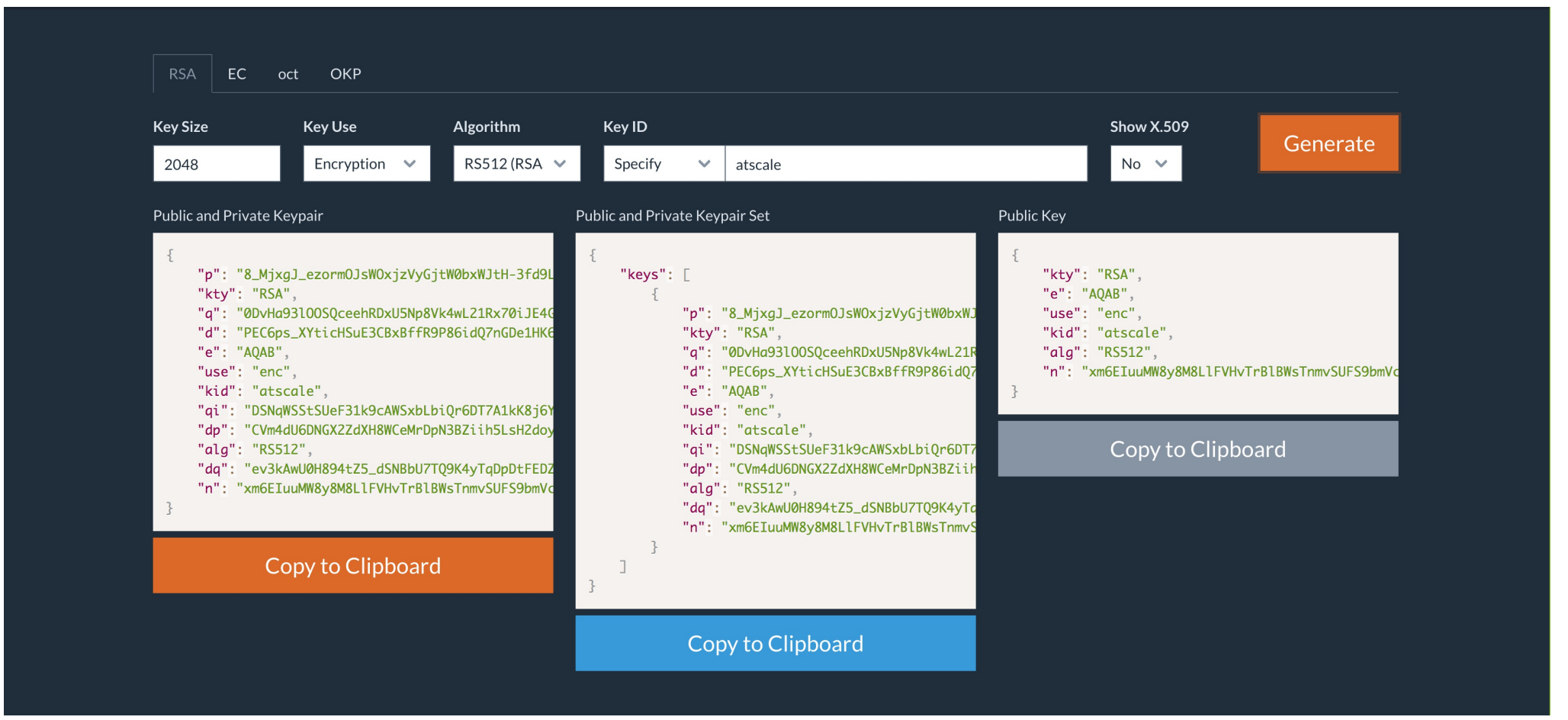
1. Create a public and private key (JWKS Token) that Okta will use to communicate.
2. Create an Administrator Okta API token that will be used to create an AtScale service.
3. Create the OAuth Service Application by combining the JWKS token (public key) and the Administrator Okta Application API token. Send the public key to AtScale.
4. Grant permissions to the AtScale service within Okta.
5. Revoke the Okta API token.
6. Configure AtScale with the private key pair and the Okta service key
7. Assign users to the AtScale service within Okta.

Procedure

1. Create a public and private key using the JWKS token generator here. AtScale will use the Public Key generated from this step.

- **Key size**: 2048
- **Key use**: signature
- **Algorithm**: RSA256
- **Key ID (Optional)**: This can be any value.
- **Show X509**: No

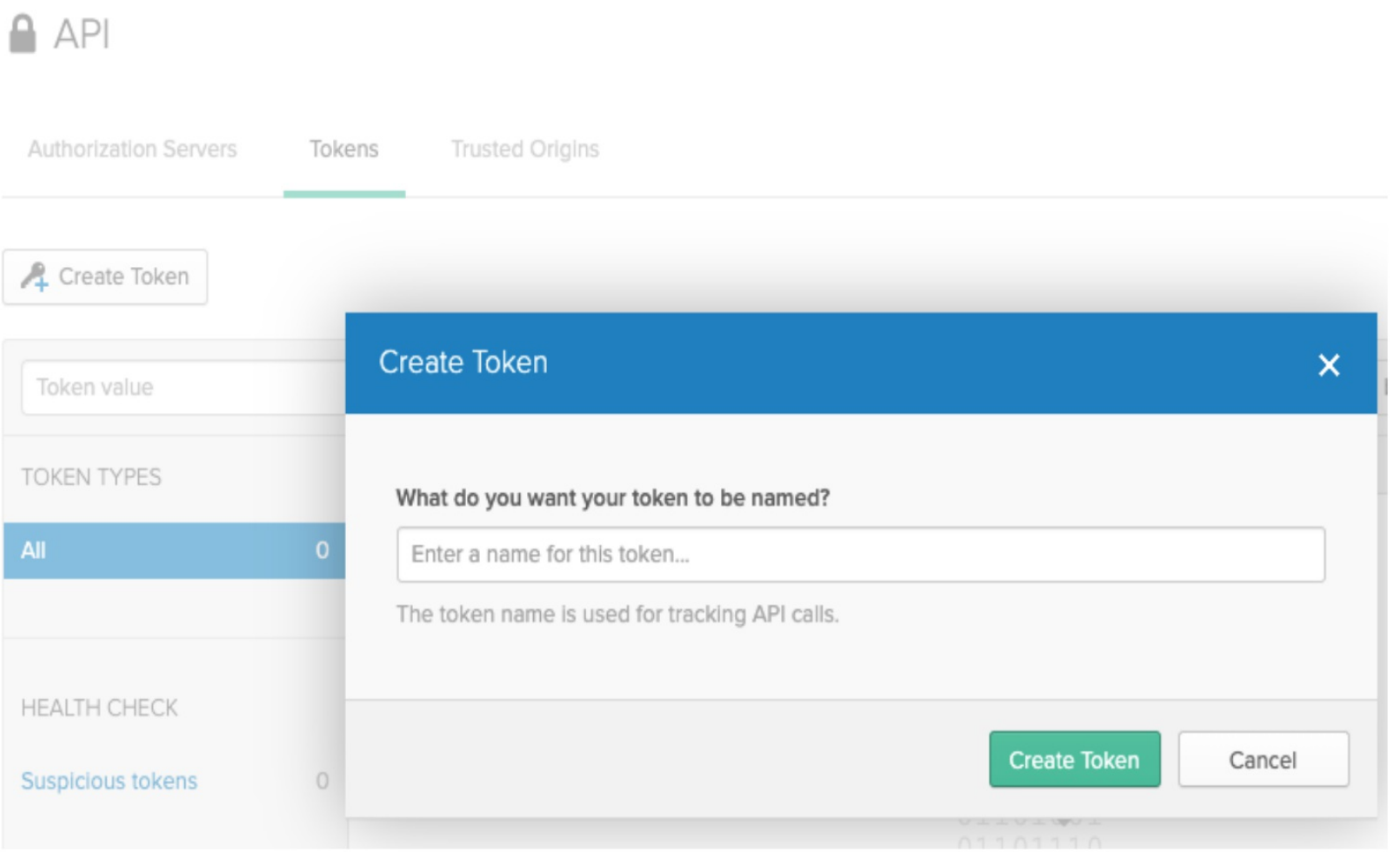
Figure 1. An example JWKS token. Save both the Public and Private Keypair (left-hand image), and the Public Key (right-most image).



2. Create an Administrator Okta Application API token.

1. Sign in to Okta as an Administrator.
2. Select **API > Tokens** and then enter a unique name for your token. This API token will be revoked later in this procedure.
3. Save the resulting SSWS token for later use.

Figure 2. The **Create Token** dialog from within an Okta application.



3. Create an **OAuth Service App** by combining the JWKS token and the Administrator Okta Application API token. You will send a CURL request to the Okta API requesting to use the JWKS token. The response will contain a new **client ID**. Additional information about this step may be obtained in the Okta documentation here.

    1. Modify the following parameters in the CURL request (see Figure 3):
    2. Replace the `${api_token}` with the Okta Application API token created in **Step 2**. Ensure to leave the SSWS prefix.
    3. Replace the example public key with the public key created in Step 1 (See Figure 1).
    4. Replace `https://${yourOktaDomain}/oauth2/v1/clients` with your Okta subdomain. Be sure to point to the subdomain and not to the admin URL.

Figure 3. an example CURL request.

```
curl -X POST \
-H 'Accept: application/json' \
-H "Authorization: SSWS ${api_token}" \
-H 'Content-Type: application/json' \
-d ' {
    "client_name": "AtScale Client",
    "response_types": [
        "token"
    ],
    "grant_types": [
        "client_credentials"
    ],
    "token_endpoint_auth_method": "private_key_jwt",
    "application_type": "service",
    "jwks": {
          "keys": [
                {
                "kty": "RSA",
                "e": "AQAB",
                "use": "sig",
                "kid": "O4O",
                "alg": "RS256",
                "n": "u<example>w"
                }
                ]
        }
}' "https://${yourOktaDomain}/oauth2/v1/clients"
```

Figure 4. An example response from a modified CURL request:

Attention: Take note of the **client_id**. This string is required to Configure the AtScale Directory setup in later steps.

```
"jwks": {
      "keys": [
         {
            "kty": "RSA",
            "alg": "RS256",
            "kid": "AtScale",
            "use": "sig",
            "e": "AQAB",
            "n": "<example>"
         }
      ]
   },
   "client_id": "<SAVE THIS ID>",
   "client_id_issued_at": 1602184770,
   "client_name": "AtScale Client",
   "client_uri": null,
   "logo_uri": null,
   "redirect_uris": [],
   "response_types": [
      "token"
   ],
   "grant_types": [
      "client_credentials"
   ],
   "token_endpoint_auth_method": "private_key_jwt",
   "application_type": "service"
```

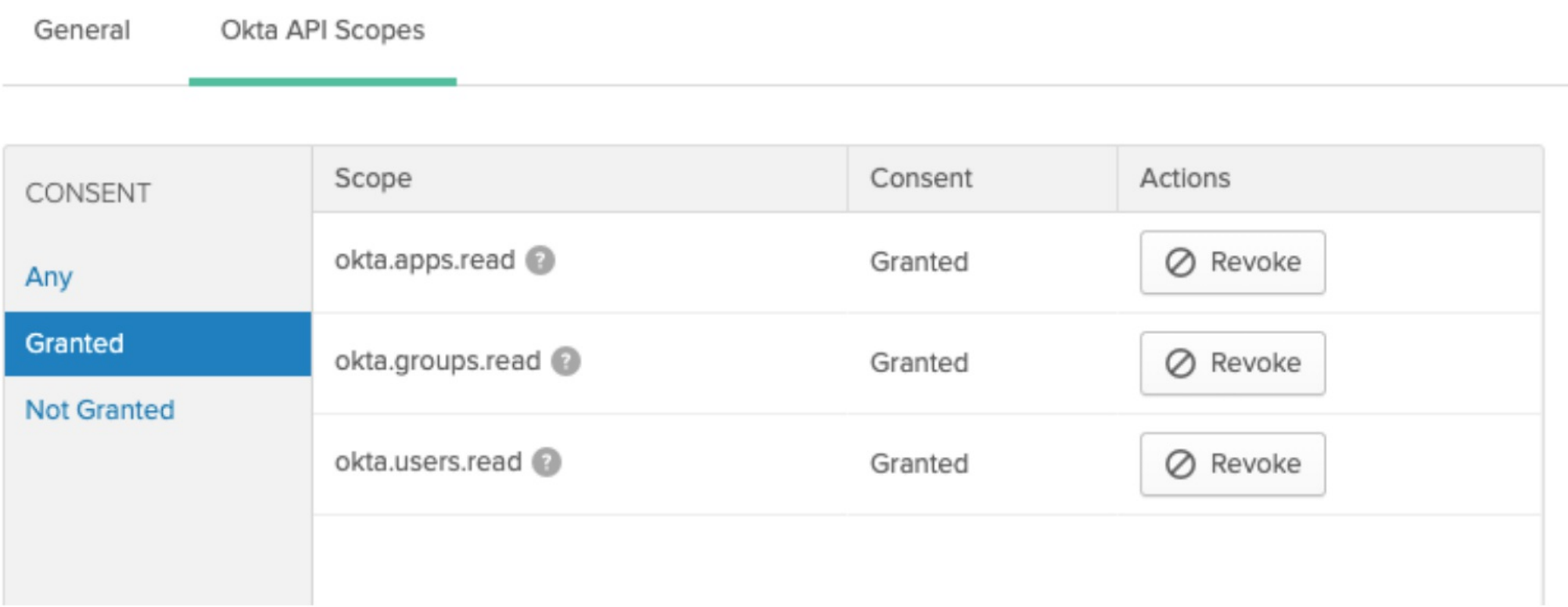
4. Grant permissions to AtScale from within the Okta Application

1. From within Okta, select **Applications**. You should have a new application named **AtScale Client** created from the previous step.
2. Select the **Okta API Scopes** tab. Grant the following Okta grants to your application.

    - okta.apps.read
    - okta.groups.read
    - okta.users.read

3. Once complete, select the **Granted** tab within Okta to ensure that the correct scopes were added.

Figure 5. The **Granted** tab displaying the requisite Okta scopes.



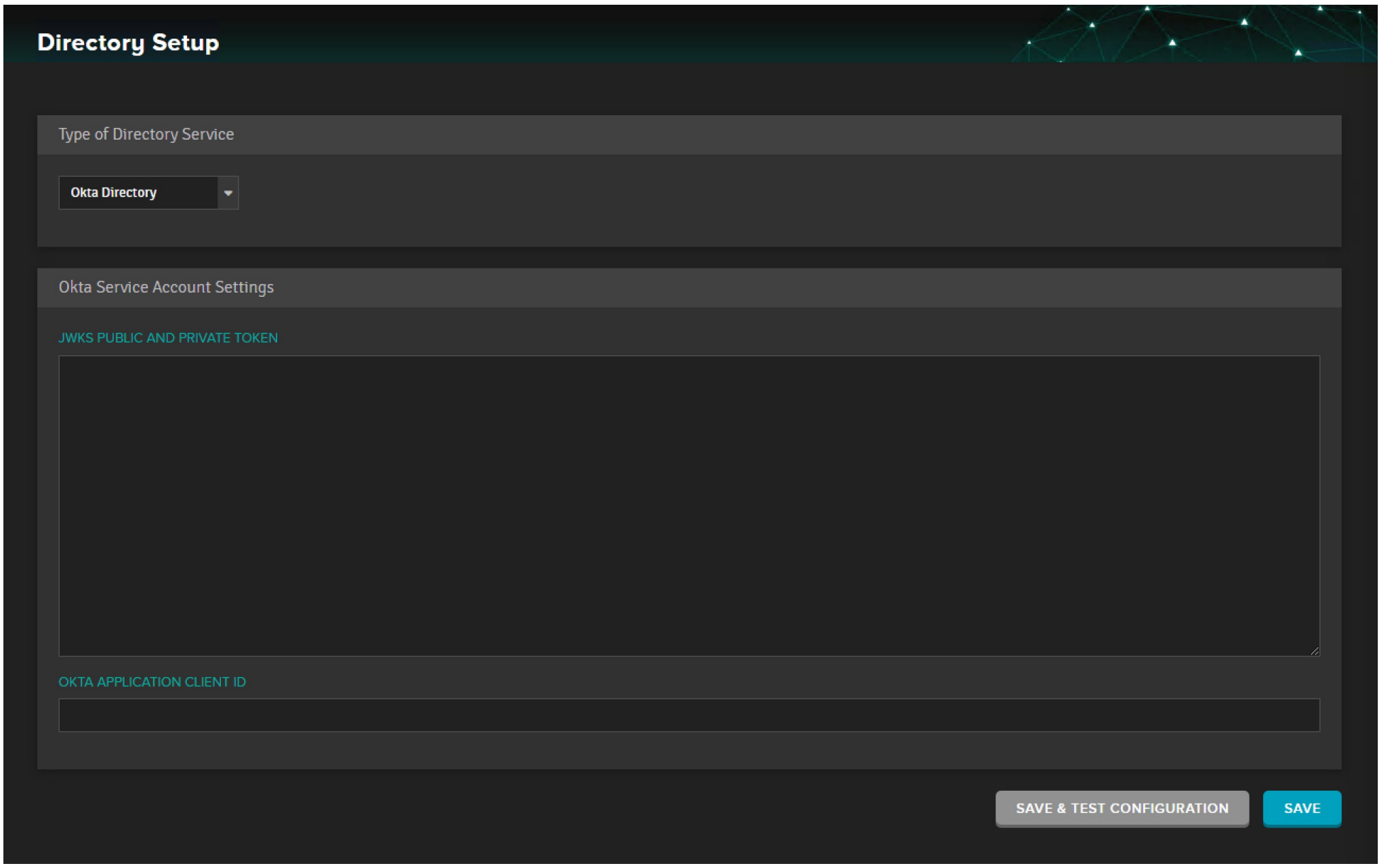
5. Revoke the API token generated in **Step 2**.

    1. In Okta Select **API > Tokens** and then select the trash button to revoke the API token.

6. Configure AtScale with the private key pair and the Okta service key.

1. In AtScale, select Security > Directory: Setup. Select the drop-down beneath **Type of Directory Service** and select **Okta Directory**.
2. Input the **Okta Application Client ID** (The **client_id** from Step 3, Figure 4) into the corresponding field in AtScale.
3. Insert the **JWKS Public and Private Key Tokens** generated from the step 1 (The left-most field from Figure 1) into the corresponding fields in AtScale.
4. Select **Save and Test Configuration** to test that the configuration was valid. Select **Save** to save the valid AtScale/Okta configuration.

Figure 5. The Okta Directory setup page within AtScale.



7. Assign users to the AtScale service within Okta.