Qualys Token-based API Authentication

Technical Brief

Modern API ecosystems demand authentication mechanisms that are secure, scalable, and developer-friendly. The legacy approach of static credentials (usernames and passwords) often presents challenges around credential storage, automated execution, role-based access, and auditability. The token-based authentication solution implemented by Qualys addresses these challenges by moving to a JSON Web Token (JWT-based) model.

This document outlines two distinct approaches for implementing token-based authentication. You can use either the Identity Provider (IdP-based) approach or use the Qualys application user interface to set up token-based authentication. This feature helps organisations to adopt token-based API access with better control, reduced credential exposure, and enhanced flexibility.

In this technical brief, you will find an overview of how token-based authentication is set up in the Qualys environment, including the steps involved and the tangible benefits that result. The content serves as a primer for both the technical implementation and the business value of token-based API authentication.

Key Highlights

- Available for both IdP and non-IdP users.
- Seamless integration of OpenID Connect (OIDC) to enhance API authentication and authorization measures.
- Compatibility with current identity providers and authentication to facilitate a seamless integration experience.
- This authentication is supported by all Qualys APIs, /api/2.0/ and onward versions.
- Eliminates the need for users to provide a username and password. This streamlines Qualys API access by allowing users to use JWT tokens, bypassing the hassle associated with usernames and passwords.

Benefits

- **Enhanced API security**: OIDC uses tokens to establish a user's identity and grant access.
- **Standardized access control**: OIDC provides a standardized way to manage user identities and access control.
- **Centralized Authentication**: By enabling IdP-initiated Single Sign On (SSO), users can authenticate once through your organization's IdP and gain access to all the necessary APIs without needing to log in again. This simplifies the user experience and reduces password fatigue, making access faster and more secure.
- **Compliance and Security**: Helps to meet compliance requirements by ensuring that user authentication processes adhere to established security protocols like SAML and OIDC.

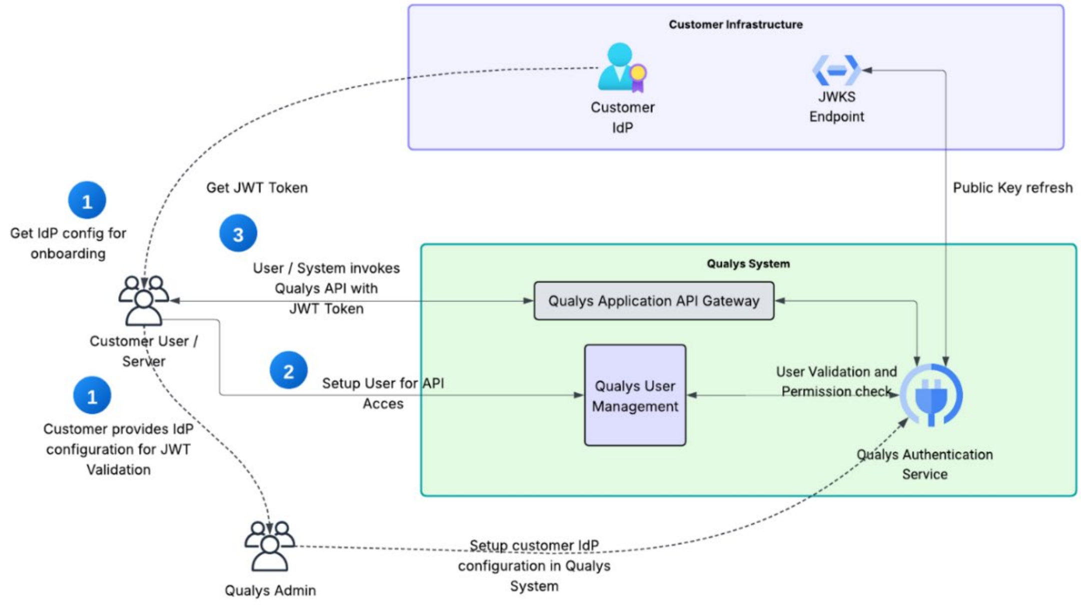
Authentication Method Comparison

The following table presents the key highlights of token-based authentication using IdP and Qualys applications. Check the following table to select an appropriate authentication method for your environment.

Token Generation	Using Customer IDP	Using Qualys Application UI
Access Control	Based on the user's roles/permissions	Granular permissions assigned to tokens
Token Expiry	Configured by the customer	Managed by Qualys (fixed at 4 hours)
User Context Requirement	Required (Mapped 1 user to 1 external ID)	Enabled on a subscription basis. All users with access to the Auth ID Client Management option in the UI can create their own user-level clients. Manager users can also create user-level as well as subscription-level clients. User mapping not required.
SSO Support	An external IDP for authentication is required	No IDP required—authentication is fully managed within Qualys
Client Types	Supported only OIDC Flow.	Supports both user-level (OIDC) and subscription-level (OAuth) clients
Token Scope	Limited to predefined scopes	Customizable scopes and permission selection during client creation

Token-based Authentication using IdP

The following workflow diagram illustrates the end-to-end API authentication process using IdP:


Customer Infrastructure
Customer IdP
JWKS Endpoint
Get JWT Token
Public Key refresh
1
Get IdP config for onboarding
3
User / System invokes Qualys API with JWT Token
Qualys System
Qualys Application API Gateway
Customer User / Server
2
Setup User for API Acces
Qualys User Management
User Validation and Permission check
1
Customer provides IdP configuration for JWT Validation
Qualys Authentication Service
Qualys Admin
Setup customer IdP configuration in Qualys System

Note: You must have an IdP solution implemented to use this approach for Token-based authentication.

Prerequisites for Setting up Token-based Authentication using IdP

To enable OIDC API authentication support, provide the following information to Qualys Support:

- **Certificates/JWKS URL:** You can provide the certificates or JSON Web Key Set (JWKS) URL in one of the following ways:
  Share the Certificates Directly — You can directly share the Key IDs (KIDs) and corresponding public signing certificates to be used. The certificates must be in **X.509 format** (typically **.pem** or **.cer** files). You can have up to 5 certificates/public keys for OIDC configuration.
  OR
  Share the JWKS URL — Confirm if your organization plans to rotate certificates, public keys, or KIDs on a regular basis. If so, provide the JWKS URL.

  This URL hosts an organization's current set of certificates/public keys along with their KIDs. It is usually managed by the IT or Identity team. We will configure this URL in our setup to support OIDC-based authentication.

Once configured, Qualys periodically retrieves the latest keys from the JWKS endpoint, helping maintain up-to-date authentication credentials without requiring manual updates.

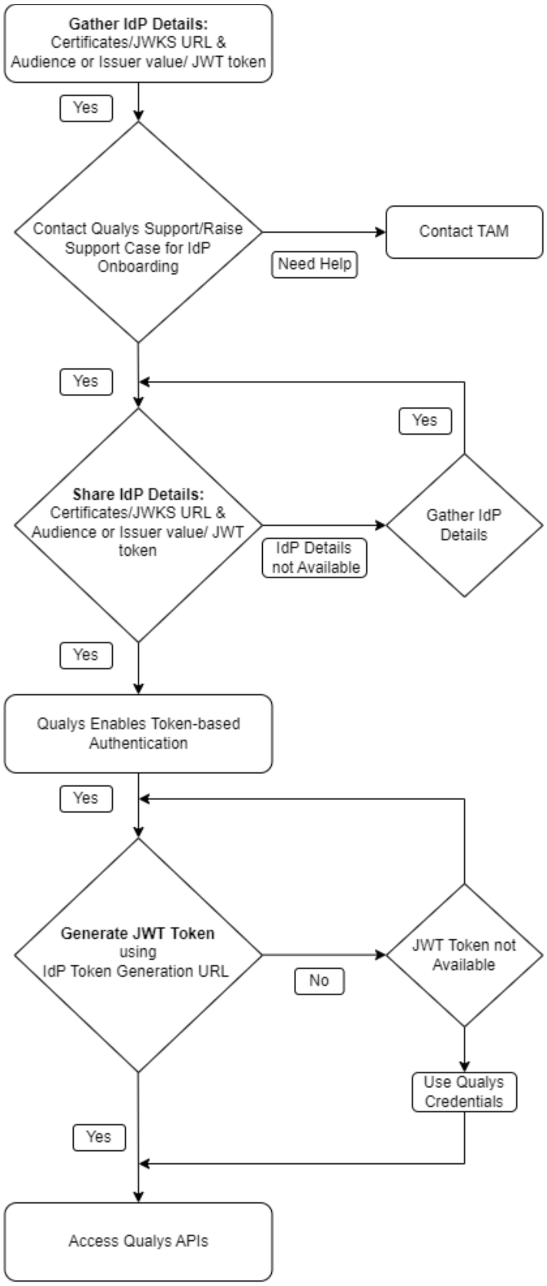
- **Audience and Issuer Values or JWT Token**: The audience and issuer values are important to set up the IdP-initiated token-based API authentication. You can,
  - provide the audience and issuer values directly,
  OR
  - Share a JWT token with us, from which we can extract these values and use them to configure certificates for passwordless authentication.

Steps to Generate IdP Details

- The JWKS URL is an endpoint published by IdP for public key generation. You can find the JWKS URL in the IdP's **Admin** console under the **OIDC Settings** tab.
  JWKS Example: `https://<okta-domain>/oauth2/default/v1/keys`
- Execute the JWKS endpoint to fetch the KIDs. The JSON response of the JWKS endpoint contains the KID.
  JWKS Response:
  ```
  {
    "kid": "abc123",
    "kty": "RSA",
    "alg": "RS256",
    "use": "sig",
    "n": "...",
    "e": "..."}
  ```
- To find the IdP certificates, in the IdPs **Admin** console, navigate to **OIDC Settings**.
- The IdP issuer value can be found in the IdP metadata.
- Audience value is usually the Client ID of your application registered in IdP.

Onboarding Steps

The following block diagram represents the onboarding steps for Token-based Authentication using IdP.


Gather IdP Details:
Certificates/JWKS URL & Audience or Issuer value/ JWT token
Yes
Contact Qualys Support/Raise Support Case for IdP Onboarding
Need Help
Contact TAM
Yes
Yes
Share IdP Details:
Certificates/JWKS URL & Audience or Issuer value/ JWT token
IdP Details not Available
Gather IdP Details
Yes
Qualys Enables Token-based Authentication
Yes
Generate JWT Token
using
IdP Token Generation URL
No
JWT Token not Available
Use Qualys Credentials
Yes
Access Qualys APIs

To start using OIDC API authentication, the following onboarding process must be completed:

1. Contact Qualys Support to request OIDC API authentication activation for your subscription.

2. Qualys Support requests the necessary technical information to enable OIDC. See the **Prerequisites** for details.

3. Once we receive the required technical information, we will enable OIDC API authentication support.

Authentication Workflow

Once the OIDC Authentication is activated for your account, you can leverage passwordless authentication for the Qualys API using an IdP. The following is the basic authentication workflow with OIDC.

1. Use the Customer IDP Token Generation URL to generate the JWT for API access.
2. Use this JWT token in the API requests. Qualys verifies if the correct JWT token is provided or not.
3. Upon successful verification, you are allowed to access the Qualys APIs.

Support for Certificate Rotation

Currently, we support certificate, public key, and KID rotation using JWKS URL. If you opt in for certificate rotation, Qualys periodically (every 30 minutes) retrieves the latest certificate, public key, and KID details. This ensures that authentication is done with the latest credentials.

Planned Enhancement for Certificate Rotation

In the upcoming release, we plan to introduce real-time dynamic JWKS URL rotation. This will help you authenticate using the latest certificates immediately, without waiting for the next scheduled interval for retrieving the authentication credentials.

Use Cases

This section illustrates the use of IdP-based token-based authentication for API authorization.

Generate JWT Token

The following sample illustrates how to generate a JWT token for accessing Qualys APIs.

API Request

```
      curl --location '<<Customer IDP token generation url>>' JWT
Token
      --header 'Content-Type: application/x-www-form-urlencoded'
      --data-urlencode 'grant_type=password' JWT Token
      --data-urlencode 'username=<<Qualys_okta_sandbox_user_id>>'
      --data-urlencode 'password=<<Qualys_okta_sandbox_password>>' \
      --data-urlencode 'client_id=
<<Qualys_okta_sandbox_application_client_id>>' \
      --data-urlencode 'scope=openid profile'
```

Note: The token generation URL may vary based on the IdP application you are using. The sample below illustrates the request and response for Okta.

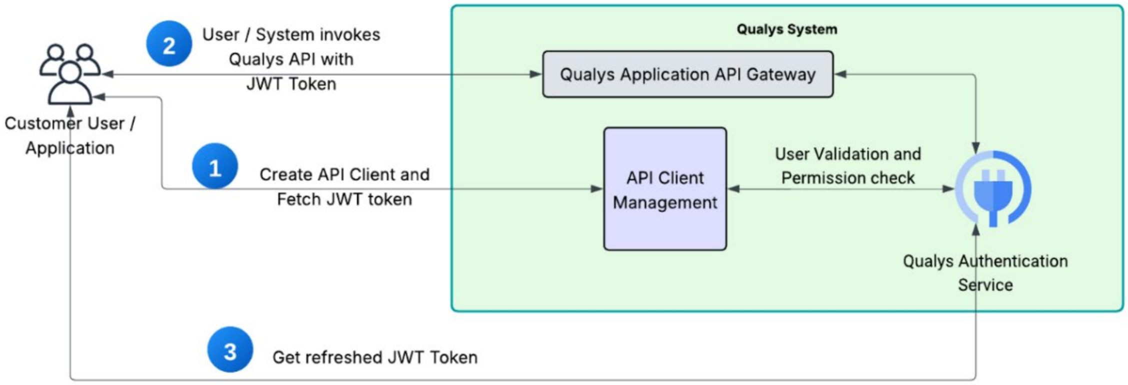
API Response

```
     {
        "token_type": "Bearer",
        "expires_in": 3600,
        "access_token": "<JWT token value>",
        "scope": "profile openid",
        "id_token": "<JWT token value>"
      }
```

Once you get the JWT token, you can use it in API requests for authentication.

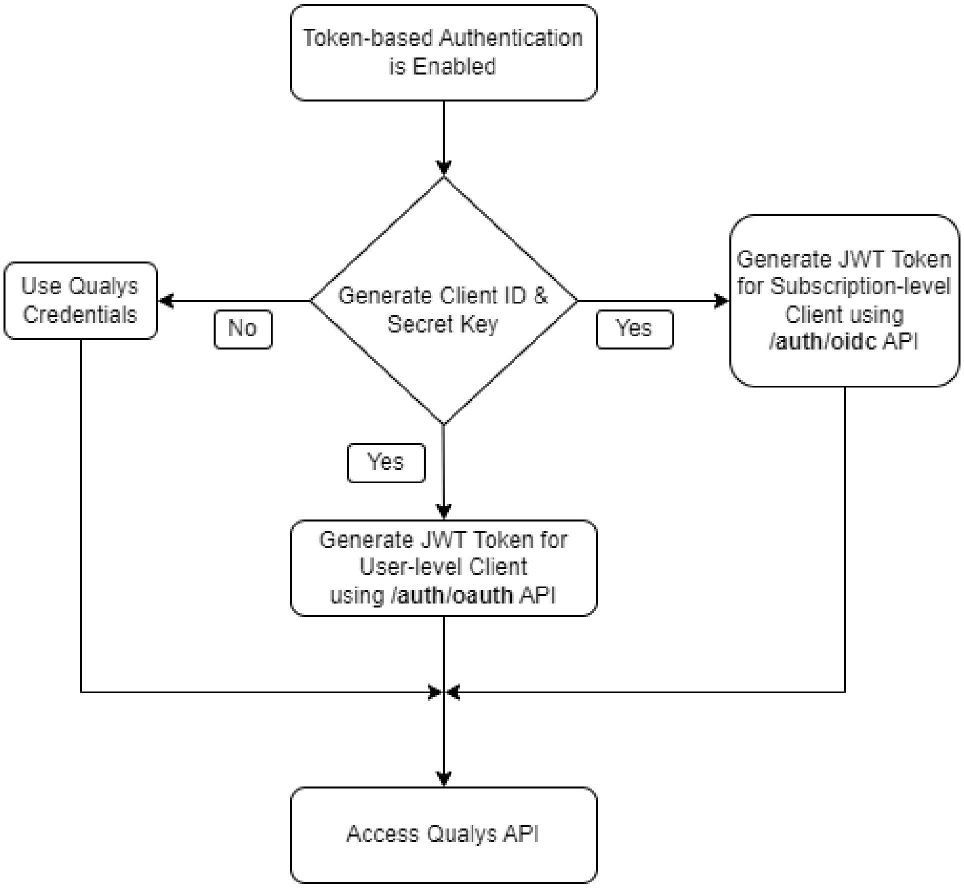
Token-based Authentication with Qualys UI

The following diagram illustrates the end-to-end data flow of token-based authentication with the Qualys application user interface.


Qualys System
2
User / System invokes Qualys API with JWT Token
Qualys Application API Gateway
Customer User / Application
1
Create API Client and Fetch JWT token
API Client Management
User Validation and Permission check
Qualys Authentication Service
3
Get refreshed JWT Token

With this approach, you can set up JWT token-based authentication using only the Qualys application user interface. This approach does not require any IdP.

The following block diagram represents the workflow for using Token-based Authentication with the Qualys application user interface.

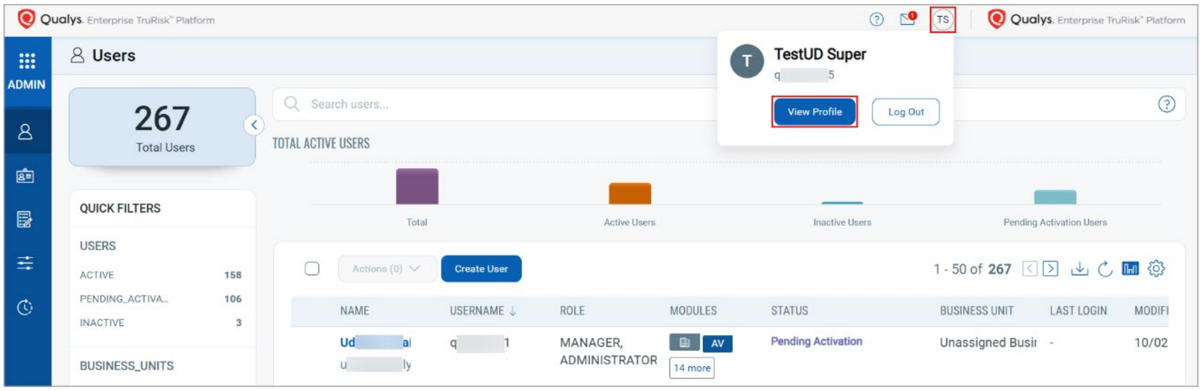

Token-based Authentication is Enabled
Generate Client ID & Secret Key
No
Use Qualys Credentials
Yes
Generate JWT Token for Subscription-level Client using /auth/oidc API
Yes
Generate JWT Token for User-level Client using /auth/oauth API
Access Qualys API

Set up Token-based Authentication from UI

The enhanced token-based authentication supports setting up user-level and subscription-level clients from the application user interface. Perform the following steps to set up the token-based authentication from the Qualys application user interface, such as the **Administration** UI.

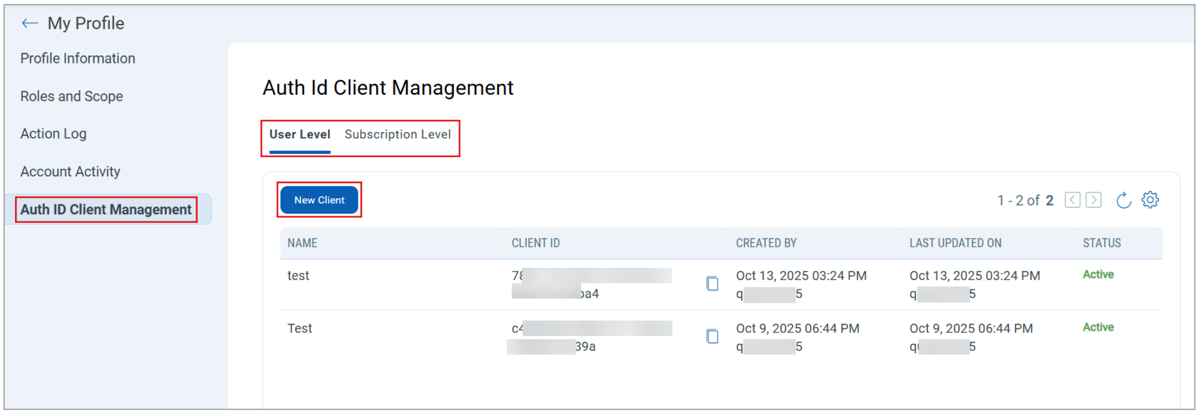
1. In the Qualys application user interface, click the User information icon.

2. In the user details banner, click **View Profile**. The **My Profile** window opens.



NAME	USERNAME ↓	ROLE	MODULES	STATUS	BUSINESS UNIT	LAST LOGIN	MODIFI
Ud[redacted]al u[redacted]ly	q[redacted]1	MANAGER, ADMINISTRATOR	AV 14 more	Pending Activation	Unassigned Busi	-	10/02

3. Click the **Auth ID Client Management** section to start creating a new user-level or subscription-level client.



NAME	CLIENT ID	CREATED BY	LAST UPDATED ON	STATUS
test	7[redacted]a4	Oct 13, 2025 03:24 PM q[redacted]5	Oct 13, 2025 03:24 PM q[redacted]5	Active
Test	c[redacted]39a	Oct 9, 2025 06:44 PM q[redacted]5	Oct 9, 2025 06:44 PM q[redacted]5	Active

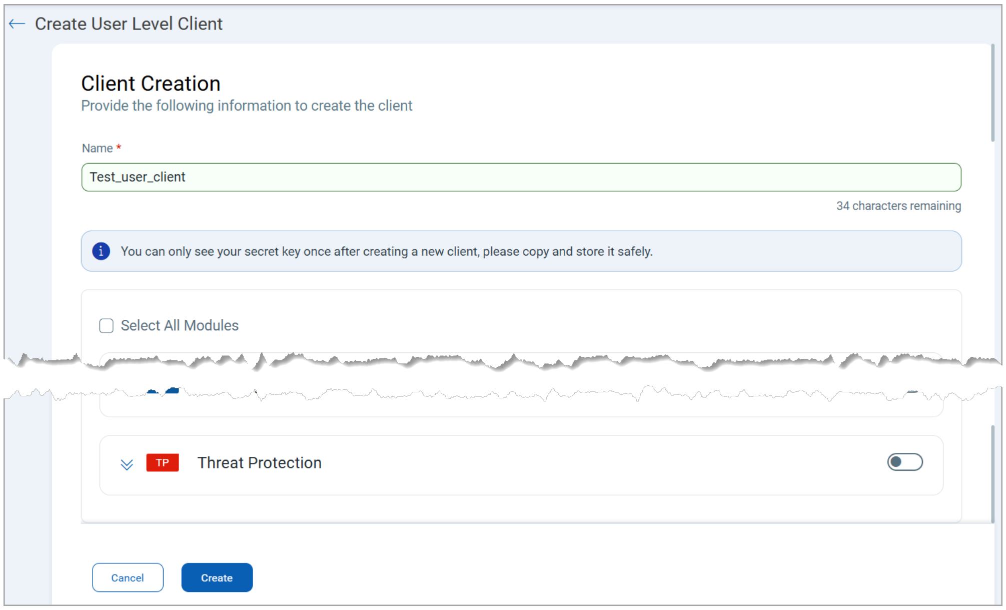
Note: The availability of the **User Level** and **Subscription Level** tabs may vary for different Qualys applications. Some Qualys applications may have both tabs, while others may have only the User Level tab.

Create User-level Client

1. In the **Auth ID Client Management** window, click the User Level tab.

2. Click **New Client**. The **Create User Level Client** window opens.

3. Enter the following details to continue with user-level client creation.



Client Creation

Provide the following information to create the client

Name *

Test_user_client

34 characters remaining

You can only see your secret key once after creating a new client, please copy and store it safely.

□ Select All Modules

Name: Enter a unique client name for the user-level client.

Assign required application permissions for the user-level client. You can give access to all the applications by selecting the **Select All Module** checkbox.

Click **Create** to complete the client creation. The **Client Secret Key** pop-up is displayed on the user interface.

4. Copy the **Secret Key** and store it at a safe place. You will need this key to generate the JWT token for API authentication.

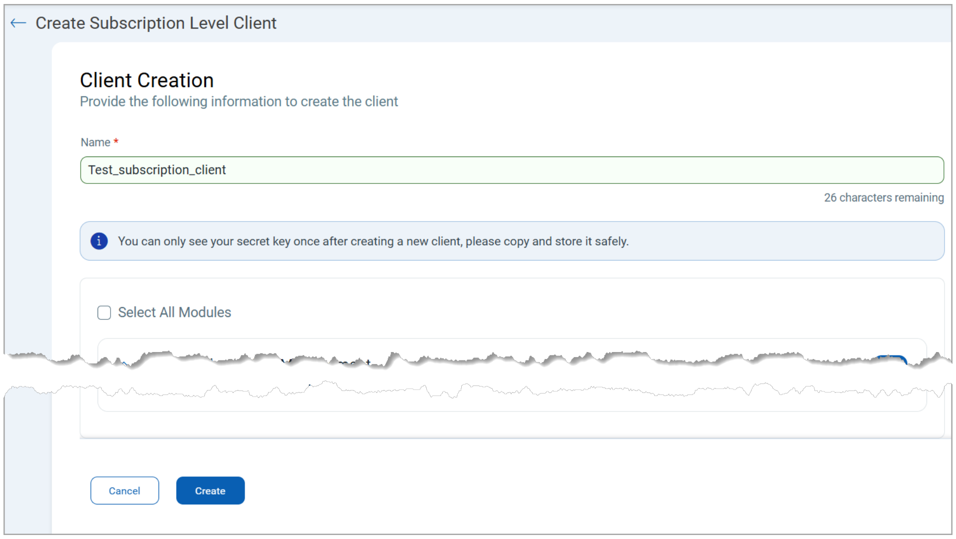
5. Click **Close**. The newly created user-level client is displayed at **Auth ID Client Management** > **User Level** tab. Copy the **Client ID** of the new client to the place where you stored the associated Secret key.

Create Subscription-level Client

1. In the **Auth ID Client Management** window, click the **Subscription Level** tab.

2. Click **New Client**. The **Create Subscription Level Client** window opens.

3. Enter the following details to continue with subscription-level client creation.



← Create Subscription Level Client

Client Creation

Provide the following information to create the client

Name *

Test_subscription_client

26 characters remaining

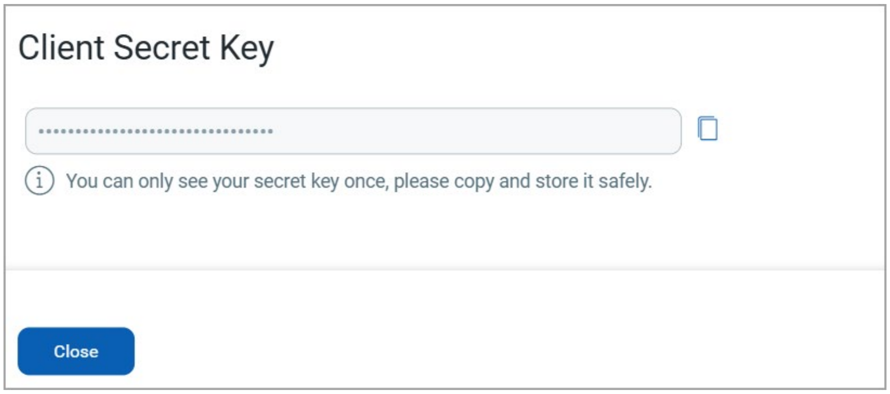
You can only see your secret key once after creating a new client, please copy and store it safely.

□ Select All Modules

Name: Enter a unique client name for the subscription-level client.

Assign required application permissions for the subscription-level client. You can give access to all the applications by selecting the **Select All Modules** checkbox. Click **Create** to complete the client creation. The **Client Secret Key** pop-up is displayed on the user interface.

4. Copy the **Secret Key** and store it at a safe place. You will need this key to generate the JWT token for API authentication.



Client Secret Key

••••••••••••••••••••••••••••••••

You can only see your secret key once, please copy and store it safely.

5. Click **Close**. The newly created subscription-level client is displayed at **Auth ID Client Management** > **Subscription Level** tab. Copy the **Client ID** to the place where you stored the associated Secret key.

Note: You can create multiple Subscription-level clients with different permissions.

Generate Authentication Tokens

Perform the following steps to generate an Authentication Token using the Client ID and Secret Key generated for the user-level and subscription-level clients. The steps to generate the JWT token for both clients are the same.

1. Execute the Qualys authentication API (/auth/oauth or /auth/oauth).
2. Provide the Client ID and Client Secret Key for the user-level or subscription-level client in the API request.
3. The JWT token for the associated user and subscription is generated. Store this JWT token at a safe place. The newly generated JWT token is valid for the next four hours.

Note: You can use the existing Client ID and Client Secret Key to generate new JWT tokens.

Use the following API Endpoints to generate the JWT Token for API Authentication:

Generate Token for Subscription Level Client

POST /auth/oidc

Use this API endpoint to generate an authentication token using Client ID and Client Secret Key for a subscription-level client.

Permissions Required: Need UI and API Access Permissions.

Input Parameters

You must provide the following input parameters in the API Request header to generate an authentication token.

Input Parameters	Mandatory/Optional	Data Type	Description
clientSecret	Mandatory	String	Provide the client secret key generated while creating the user-level or subscription-level client.
clientId	Mandatory	String	Provide the Client ID for the user-level or subscription-level client for whom you want to generate the JWT Token.
encrypted {}	Optional	String	Provide the encryption details for JWT Token. The encrypted JWT Tokens improve the security.

Sample: Generate an Authentication Token for Subscription Level Client

This API illustrates generating an authentication token to access Qualys APIs using the Client ID and Client Secret Key. Provide the Client ID and Client Secret Key in the API request to fetch the authentication token.

API Request

```
  curl -X POST '<qualys_base_url>/auth/oidc'
--header 'clientSecret: wJalrXUtnFEMI/K7MDENG+bPxRfiCYEXAMPLEKEY'
--header 'clientId: 123e4567-e89b-12d3-a456-426614174000'
--data-raw ''
```

API Response

```
eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCJ9.
eyJzdWIiOiJjbGllbnQxMjM0NTYiLCJzYwZXM
iOlsiYXBpLnJlYWQiLCJhcGkud3JpdGUiXSwi
aXNzIjoiaHR0cHM6Ly9kdW1teS1hdXRoLmNvb
SIsImV4cCI6MTcwMDAwMDAwMH0.SdXn3I6yTb-
JNk9LPjR8W9xAtH7dN3Mqf3HdJ5WnRfE
```

Generate Token for User Level Client

POST /auth/oauth

Use this API endpoint to generate an authentication token using Client ID and Client Secret Key for a user-level client.

Permissions Required: Need UI and API Access Permissions.

Input Parameters: Refer to the input parameters described for the subscription-level client.

Sample: Generate an Authentication Token for User-level Client

This API illustrates generating an Authentication Token to access Qualys APIs using the Client ID and Client Secret Key. Provide the Client ID and Client Secret Key in the API request to fetch the JWT token.

API Request

```
curl -X POST '<qualys_base_url>/auth/oidc'
--header 'clientSecret: wJalrXUtnFEMI/K7MDENG+bPxRfiCYEXAMPLEKEY'
--header 'clientId: 123e4567-e89b-12d3-a456-426614174000'
--data-raw ''
```

API Response

```
eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCJ9
eyJzdWIiOiJjbGllbnQxMjM0NTYiLCJzYwZXM
iOlsiYXBpLnJlYWQiLCJhcGkud3JpdGUiXSwi
aXNzIjoiaHR0cHM6Ly9kdW1teS1hdXRoLmNvb
SIsImV4cCI6MTcwMDAwMDAwMH0.SdXn3I6yTb-
JNk9LPjR8W9xAtH7dN3Mqf3HdJ5WnRfE
```

Once the authentication token is generated, you can use it to authenticate your API requests. Include the token in the Authorization header of API requests.

The system validates the authentication token and authorizes the request based on the user's permissions.

Troubleshooting Token-based Authentication

The following table illustrates the troubleshooting steps for Token-based authentication.

Error	Root Cause	Solution
400 – Invalid Signature	Invalid Token signature (Incorrect certificate files/ JWKS sync issue)	• Update JWKS URL or Certificates • Verify KID • Validate IdP configuration
	Invalid audience and issuer values	Update IdP with the correct audience and issuer values
401 – Invalid Token / Unauthorized request	Expired, missing, invalid, or malformed JWT token	• Regenerate JWT token • Fix Token Header • Check the System date and time for token validity
	Invalid token signature	• Update JWKS URL or Certificates • Verify KID • Validate IdP configuration
403 – Permission Denied	Insufficient Scope/permissions	• Update the user scope from IdP • Update client permissions from UI
429 – Too Many Requests	API rate limit reached	• Add retry logic to APIs • Reduce API request frequency

Error	Root Cause	Solution
500 – Service not available	Server misconfigurations	Contact Qualys Support.

Appendix A – Resources for Other IdPs

Refer to the attached PDF documents for more information on setting up token-based authentication with some other IdPs.

Identity Provider (IdP)	Description	Document Source
Microsoft Entra (Azure AD)	Microsoft Entra/Azure Active Directory (AD) Identity Provider (IdP) offers secure, cloud-based identity and access management, enabling single sign-on, multi-factor authentication, and seamless integration with apps and services for modern zero-trust security.	PDF generate_jwt_token_microsoft_entra.pdf
Okta	Okta Identity Provider delivers secure, cloud-based identity and access management, enabling single sign-on, adaptive multi-factor authentication, and seamless integration with applications to simplify user access and strengthen enterprise security.	PDF generate_jwt_token_okta.pdf

Appendix B – Frequently Asked Questions (FAQs)

The following are some of the most frequently asked questions about token-based authentication.

1. What is Qualys Token-based API authentication, and why would I use it?
   Answer: Token-based API authentication lets you authenticate Qualys APIs via your Identity Provider (IdP) and Qualys UI, issuing JWT tokens that you pass as Bearer tokens to Qualys endpoints, reducing reliance on username/password and aligning with modern SSO practices.

2. Is token-based authentication enabled by default in my Qualys subscription?
   Answer: No. token-based API authentication requires onboarding and enablement by Qualys Support for your subscription before you can use it.

3. Once token-based authentication is enabled, how do I use the token with Qualys APIs?
   Answer: After you obtain a JWT from your IdP, include it in API requests as Authorization: Bearer <JWT>; Qualys APIs (e.g., /api/2.0/ and later) will accept the token for authorization.

4. How do I generate a JWT for API Access if my IdP is Okta (example)?
   Answer: Call your IdP's token endpoint (varies by IdP) using form-encoded parameters such as grant_type, client_id, username, password, and scope=openid profile; the response returns access_token and id_token.

5. What is the "Token-based authentication from the Qualys UI," and when should I prefer it?
   Answer: It's a Qualys-managed JWT solution where you create user-level or subscription-level clients in the Qualys UI to obtain JWT tokens—ideal when you don't have (or don't want to integrate) an external IdP, and when you need granular, role-based API access for automation.

6. What's the difference between "user-level" and "subscription-level" clients?
   Answer: User-level clients are tied to a single user (manager or non-manager) and become invalid if that user is deactivated; use these for manual workflows. Subscription-level clients are created by manager users, usable across the subscription, and better for automated API workflows.

7. How do I set up token-based authentication clients in the Qualys UI?
   Answer: Go to Auth ID Client Management in your profile, choose User Level or Subscription Level, create a client, assign application permissions, then securely store the generated Client ID and Client Secret Key for token generation.

8. After creating a client, how do I obtain a JWT and call APIs?
   Answer: Use the Client ID and Client Secret to generate an authentication token (JWT) per the UI-based flow, then pass it as Authorization: Bearer <token> in subsequent Qualys API requests (e.g., Cloud Platform, ETM, CSAM, PM, TotalCloud, Cloud Agent).

9. Can I still use basic authentication if OIDC is enabled?
   Answer: Yes. When OIDC API authentication is implemented, you can use either basic authentication or JWT token authentication while executing Qualys APIs, depending on your security posture and migration timeline.

Last updated: December 15, 2025