Integrating other systems with Venafi products

Every API integration needs to register with Venafi. Registering your Integration allows the client to get an OAuth access token and refresh token with permissions to the REST API endpoints your integration needs. API calls automate your work. By default, all Venafi products, utilities, and other open-source projects are pre-registered. You only need to register integrations from the Venafi Marketplace or your own custom-built integrations.

How does it work?

The API Integration wizard registers details about API calls that the client will make. After registration, the client requests a REST authorization using that same information. The VEDAuth service responds with an access token to allow the client to make API calls. The token is valid until it expires or is revoked. This eliminates the need to continuously get new API keys.

If configured for your integration, you also get a refresh token. When a token expires, the refresh token allows you to get an access token from the same grant. Refresh tokens allow you to get new tokens without re-authenticating to the VEDAuth service. A refresh token is valid only once and it remains valid until your grant expires.

First things first

- Familiarize yourself with OAuth terms and the flow process. For more information, see About API integrations.
- From the developer, get a list of scopes and restrictions based on the integration's needs. Need help? Use this basic chart.

Scope and Developer Example	Privileges and Restrictions					
	Approve	Delete	Discover	Manage	Revoke	[Other]
admin *scope: admin:recyclebin*						recyclet
agent *scope: agent*		✓				
certificate *scope: certificate:approve,manage*	✓	✓	✓	✓	✓	

Scope and Developer Example	Privileges and Restrictions					
	Approve	Delete	Discover	Manage	Revoke	[Other]
codesignclient *scope: codesignclient*						
codesign *scope: codesign:delete,manage*		✓		✓		
configuration *scope: configuration:delete,manage*		✓		✓		
restricted *scope:restricted:delete,manage*		✓		✓		
security *scope: security:delete,manage*		✓		✓		
ssh *scope: ssh:approve,delete,manage*	✓	✓	✓	✓		
statistics (requires Vendor integration) *scope:statistics*						
(**Read access**) Specify a scope. *scope:certificate*						
(**Many scopes**) Use a semi-colon (**;**) between each scope. *scope:ssh;certificate:discover,manage;configuration:manage*						

To register (create) a new integration

1. If you've not done so, set the expiration and refresh time for tokens. See Setting up token authentication.
2. From the Platform menu bar, click **API** > **Integrations**.
3. Do one of the following:

   ALL OF THE SCOPES WE NEED APPEAR IN THE TABLE (ABOVE)

   a. Select **New**.

   b. In **Overview**, complete the form. If you want the client **Name** and **Client ID** to match, leave **Client ID** blank. Click **Next**.

   c. In **Base access**,select the **Scope** and its corresponding **Privileges and restrictions** (if any).

   Read is an implied restriction that is automatically given with any scope. So endpoints that are read-only will automatically have access via a token, even if there are no restrictions given. The settings should match the developer's list of API

calls that the client will use. If a scope or restriction is missing, restart the wizard and use the Import option to specify the needed scopes.

For example: my integration discovers certificates and logs progress. It needs `certificate` scope with `manage` and `discover` restrictions. I also select `log` scope with `manage` and `delete` restrictions.

d. (Optional) To override token refresh settings, click **Use global default settings**, customize the days, hours, or minutes, and then click **Next**.

e. In **User or team access**,search for people or service accounts who will be running your integration. Then click **Add**.

f. Click **Done**.

SPECIAL SCOPES/RESTRICTIONS INCLUDING VENAFI MARKET PLACE INTEGRATIONS

What's next?

- From the Overview tab, copy and give the JSON to the developer. The developer customizes the JSON to request a token via an Authorize method, such as POST Authorize/OAuth.
- Later, if you need to make changes, see Updating integration registrations