Intégration sélective EntraID / PREVIST via Azure Functions

Table des matières

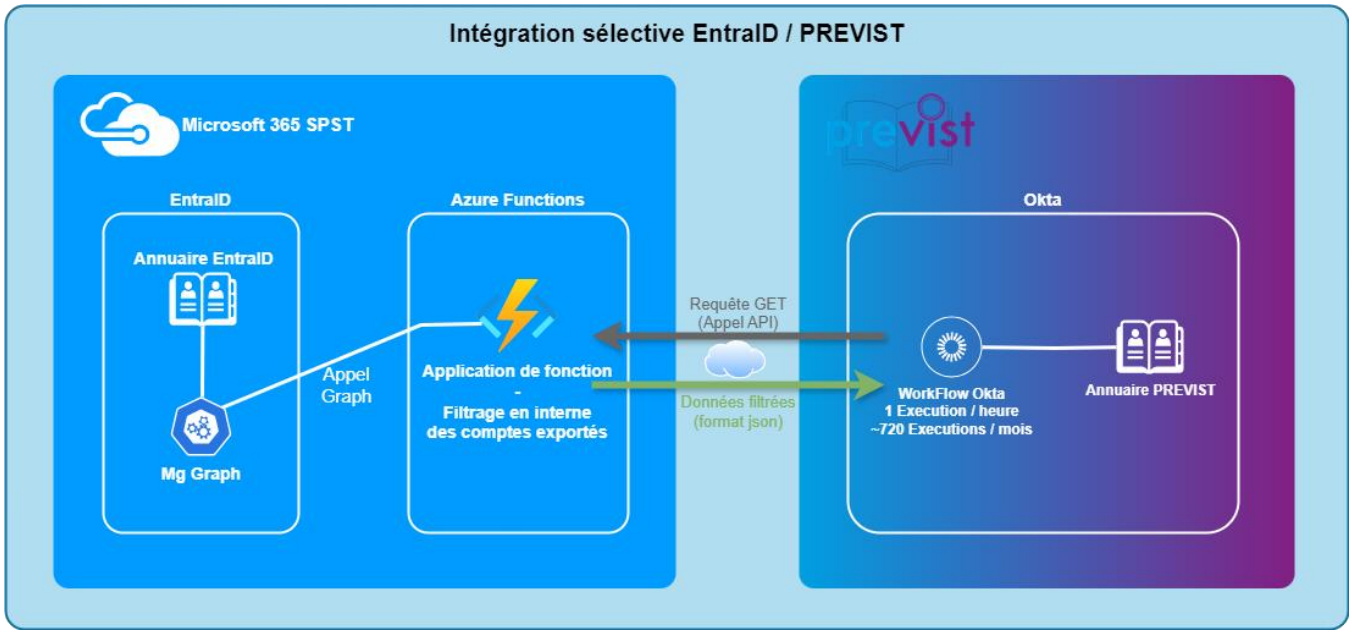
1. Objectif & principe de sécurité « moindre privilège »

Risque initial :
Pour l'intégration des comptes EntraID dans PREVIST (IAM Okta), Microsoft Graph ne permet pas de limiter finement la portée des autorisations de lecture (« User.Read.All » et « Group.Read.All » donnent accès à tout le tenant). *L'établissement doit rester maitre des données présentées.*

Solution :
Une fonction HTTPS (publiée sur Azure Functions) lit MgGraph en interne de façon contrôlée (groupe cible) et expose seulement les comptes désirés à Okta par le biais d'une variable (groupe de sécurité) paramétrable par l'établissement, avec authentification par clé et filtrage IP côté Application de Fonction.
Différentes variables sont paramétrables en interface graphique par l'établissement (secret, groupe à partager, paramètres à exposer).

Vue d'ensemble :
L'Application de Fonction appelle Microsoft Graph en interne sur un groupe racine défini, explore les sous-groupes, concatène les utilisateurs, et retourne un payload JSON strict.
PREVIST vient consommer l'Endpoint HTTPS (auth niveau function, clé secrète paramétrable requise) et intègre les utilisateurs via des Workflows. Possibilité de limiter l'accès à la fonction par filtrage IP.

Bénéfices :
L'établissement conserve la main (paramétrage, activation/blocage, scope fonctionnel) sans exposer l'ensemble du tenant à PREVIST.
L'application de fonction Azure exclut le maintien d'un environnement serveur.



2. Données exposées à Okta

L'application exposera le groupe racine paramétré par l'établissement en variable d'environnement. Il affichera en sortie les comptes présents dans ce groupe (N0), rangés par sous-groupe (N1).
Les comptes présents dans le N0 accèderont à PREVIST (SSO Okta), les comptes présents dans le sous-groupe N1 accèderont aux applications avec les accès liés à ce groupe.
Nb : Pour les accès Curebot, retrouver la matrice des groupes dans l'annexe Annexe_Matrice_Metiers_Curebot.pdf.
Nb2 : Les groupes modernes « Microsoft 365 » ne peuvent être imbriqués, il est nécessaire d'utiliser des groupes de sécurité.

Les comptes présents dans les sous-groupes (N2/3/4/etc…) seront affichés en sortie dans le sous-groupe N1. Les noms de groupes N1/2/3/4/etc… ne seront pas affichés. Une sécurité est en place pour éviter les boucles infinies.

Les paramètres exportés par défaut (non paramétrables) sont : id, userPrincipalName, surname, givenName.
Il s'agit des paramètres nécessaires au cycle de vie du compte dans l'IAM.

- ID : permet de prendre en compte en cas de renommage du compte.
- UPN : login PREVIST (nécessaire pour le SSO).
- Nom et Prénom : pour la complétion des templates.

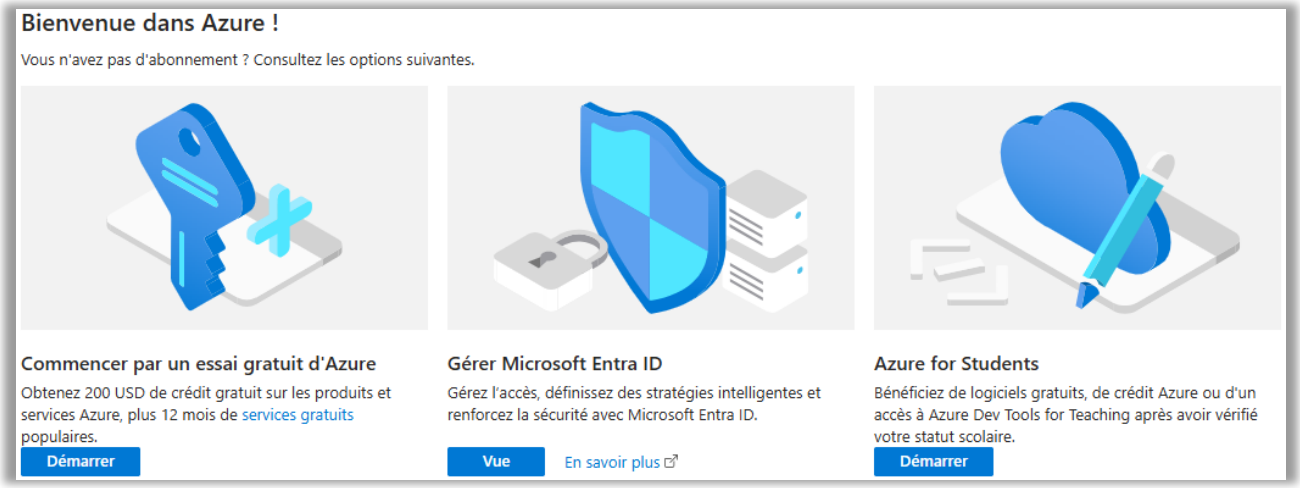
D'autres attributs à exporter peuvent être paramétrés par l'établissement dans une variable d'environnement au besoin (attributs MgGraph).

Exemple de sortie :


```
{
    "success": true,
    "groupName": "GRP_PREVIST",
    "groupId": "********-****-****-****-************",
    "count": 3,
    "users": [
        {"@odata.type": "#microsoft.graph.user","id": "********-****-****-****-************","userPrincipalName":
"utilisateur.test6@previst.fr","surname": "TEST6","givenName": "Utilisateur"}
    ],
    "groups": [
    {
        "@odata.type": "#microsoft.graph.group",
        "displayName": "GRP_PREVIST_Curebot_Medecin",
        "userPrincipalName": null,
        "users": [
            {"@odata.type": "#microsoft.graph.user","id": "********-****-****-****-************","userPrincipalName":
"utilisateur.test4@previst.fr","surname": "TEST4","givenName": "Utilisateur"},
            {"@odata.type": "#microsoft.graph.user","id": "********-****-****-****-************","userPrincipalName":
"utilisateur.test1@previst.fr","surname": "TEST1","givenName": "Utilisateur"}
        ]
    },
    {
        "@odata.type": "#microsoft.graph.group",
        "displayName": "GRP_PREVIST_Curebot_Infirmier",
        "userPrincipalName": null,
        "users": [
            {"@odata.type": "#microsoft.graph.user","id": "********-****-****-****-************","userPrincipalName":
"utilisateur.test5@previst.fr","surname": "TEST5","givenName": "Utilisateur"},
            {"@odata.type": "#microsoft.graph.user","id": "********-****-****-****-************","userPrincipalName":
"utilisateur.test2@previst.fr","surname": "TEST2","givenName": "Utilisateur"}
        ]
    },
    {
        "@odata.type": "#microsoft.graph.group",
        "displayName": "GRP_PREVIST_Curebot_Qualite",
        "userPrincipalName": null,
        "users": [
            {"@odata.type": "#microsoft.graph.user","id": "********-****-****-****-************","userPrincipalName":
"utilisateur.test10@previst.fr","surname": "TEST10","givenName": "Utilisateur"},
            {"@odata.type": "#microsoft.graph.user","id": "********-****-****-****-************","userPrincipalName":
"utilisateur.test2@previst.fr","surname": "TEST2","givenName": "Utilisateur"}
        ]
    }
    ]
}
```

3. Pré-Requis

➔ Abonnement Azure
Pour vérifier s'il existe un abonnement en cours, se rendre sur portal.azure.com, si l'encart ci-dessous s'affiche, alors ***il n'existe pas d'abonnement en court*** lié au compte utilisé et il ne sera pas possible de créer d'application sur le portail Azure :



➔ Carte en cours de validité
Nb : **Pour créer un abonnement Azure, il est nécessaire de renseigner une carte en cours de validité pour confirmer son identité** : cette restriction s'applique à l'essai gratuit, au paiement à la demande, et également dans le cadre de l'offre des 2000$ de crédits Azure offerts pour les organisations à but non lucratif.
Alternativement, il est possible de contacter son revendeur Microsoft pour créer un abonnement Azure dans Open en renseignant un numéro de licence. Cette licence n'est pas proposée chez tous les revendeurs.

➔ Accès admin au portail Azure pour gérer les abonnements. Différents abonnements peuvent exister au sein d'un même tenant. L'abonnement est disponible pour le compte l'ayant créé et peut être assigné à d'autres comptes par le créateur ou un compte admin.

➔ Accès admin au tenant Microsoft EntraID pour créer l'Application Microsoft Graph et consentir les permissions d'application.

➔ Système d'hébergement
Windows *Consommation* : support continu. (recommandé)
Linux *Consommation* : retrait le 30 septembre 2028 → migrer vers *Consommation Flexible* (Windows *Consommation* non impacté).

4. Abonnement Azure

La mise en place d'une Application de Fonction nécessite différentes ressources pour fonctionner.
La fonction utilisera un *Plan Consommation* : 1 M d'exécutions/mois gratuites et 400 000 GB-s gratuits/mois. Une exécution correspond à un appel de la fonction. PREVIST l'appellera une fois par heure, soit environ 720 exécutions/mois. Il est possible d'utiliser le plan *Consommation Flexible*, qui a des quotas différents mais garde 1 M d'exécutions gratuites.

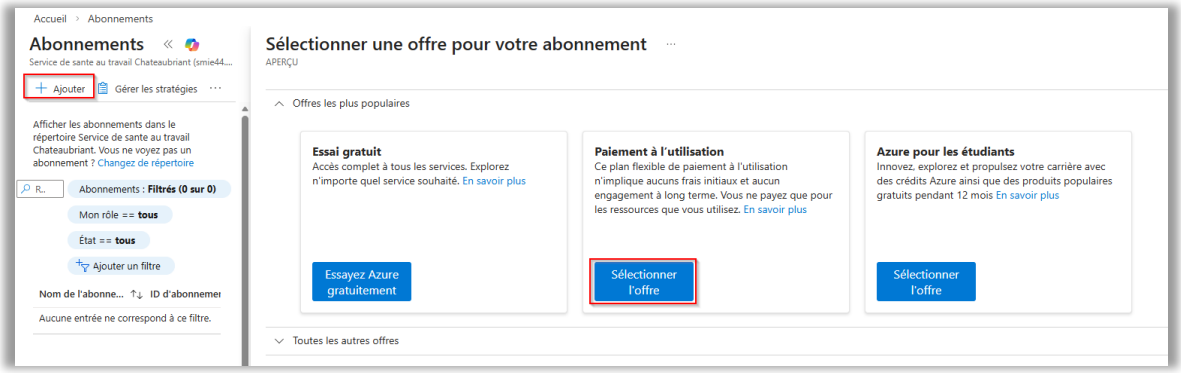
Pour fonctionner, l'application nécessitera une ressource Storage et une ressource Application Insights (créées automatiquement).

- **La ressource *Storage* nécessaire pour le fonctionnement de l'application sera facturée (<1€/mois).**
- **Certaines options de monitoring disponibles dans la ressource *Application Insight* seront facturées (prix variable en fonction des paramètres).**
- **La création d'un *Key Vault* optionnel pour le renouvellement automatique du secret sera facturée (<1€/rotation).**

Nb : Il est possible de recevoir 2000$ de crédits Azure annuels pour les organisations à but non lucratif.
Vérifier la page de tarification la plus à jour.

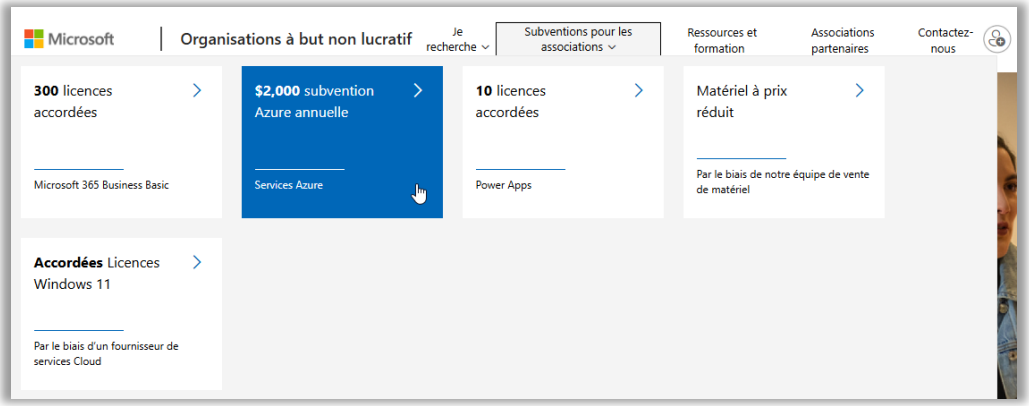
Se rendre sur le Portail Azure et sélectionner le menu : Abonnements

1- Cliquer sur Ajouter
2- Sélectionner l'option Paiement à l'utilisation.
Pour renseigner un numéro de licence Azure dans Open (disponible auprès de certains revendeurs Microsoft), choisir « Toutes les autres offres » et sélectionner l'option Azure dans Open.



3- Renseigner les informations sur le propriétaire de l'abonnement et prendre connaissance des conditions d'utilisation de l'abonnement. Plusieurs abonnements peuvent exister sur un même portail Azure.
Seuls le compte créateur de l'abonnement et le compte administrateur principal du Tenant sera en mesure de créer des ressources dans l'abonnement. Il sera nécessaire de donner les accès à l'abonnement aux autres comptes nécessitant l'accès aux ressources PREVIST.
4- Renseigner une carte en cours de validité.
Pour Azure dans Open, entrer le numéro de licence fourni.

5- Tenant éligible non-profit uniquement : Pour bénéficier de 2000$ annuels de crédits Azure, se rendre sur nonprofit.microsoft.com et se connecter avec le compte propriétaire du tenant. Choisir le menu Subventions pour les associations – Services Azure.



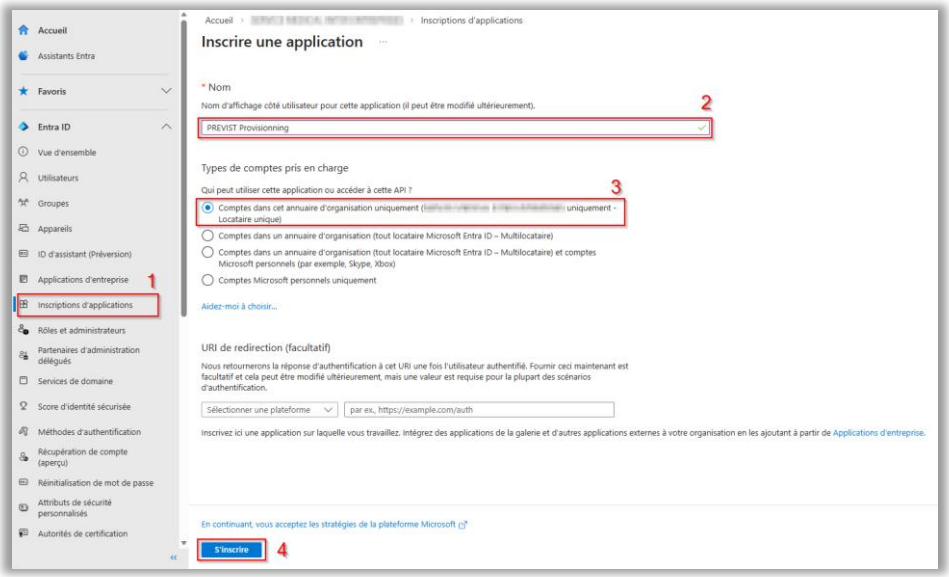
6- La demande sera traitée sous 48h. Une confirmation par mail (ou un refus le cas échéant) sera envoyée sur la boite mail du propriétaire du tenant. L'activation des crédits devra se faire depuis un lien présent dans ce mail sous 90 jours.

7- Le propriétaire du tenant recevra une alerte de renouvellement un mois avant l'expiration des crédits. Le renouvellement peut être anticipé et ne sera pris en compte qu'à la date anniversaire.

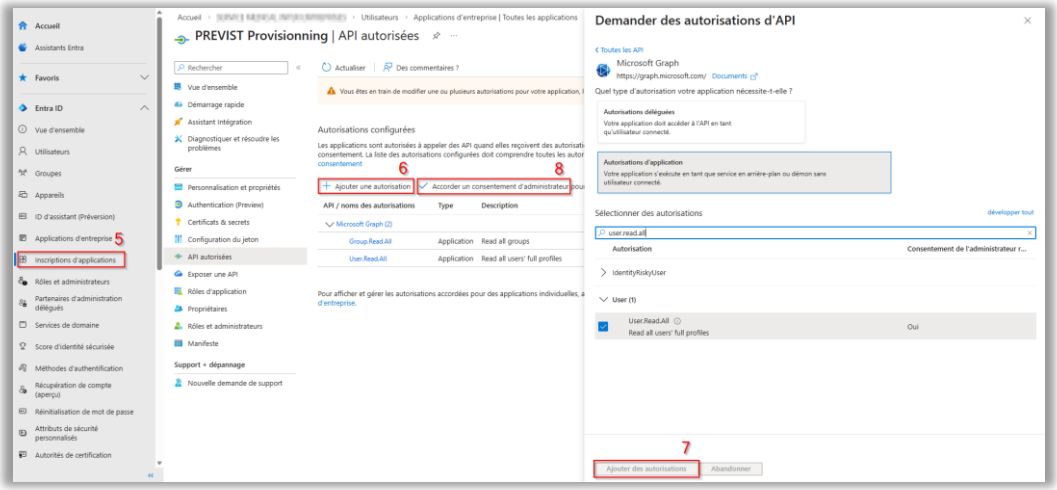
5. Application Microsoft Graph

Créer une application dans le Centre d'administration Microsoft Entra :

1- Inscriptions d'applications – Nouvelle Inscription
2- Nom : PREVIST Provisioning (recommandé)
3- Types de comptes pris en charge : Comptes dans cet annuaire d'organisation uniquement
4- S'inscrire

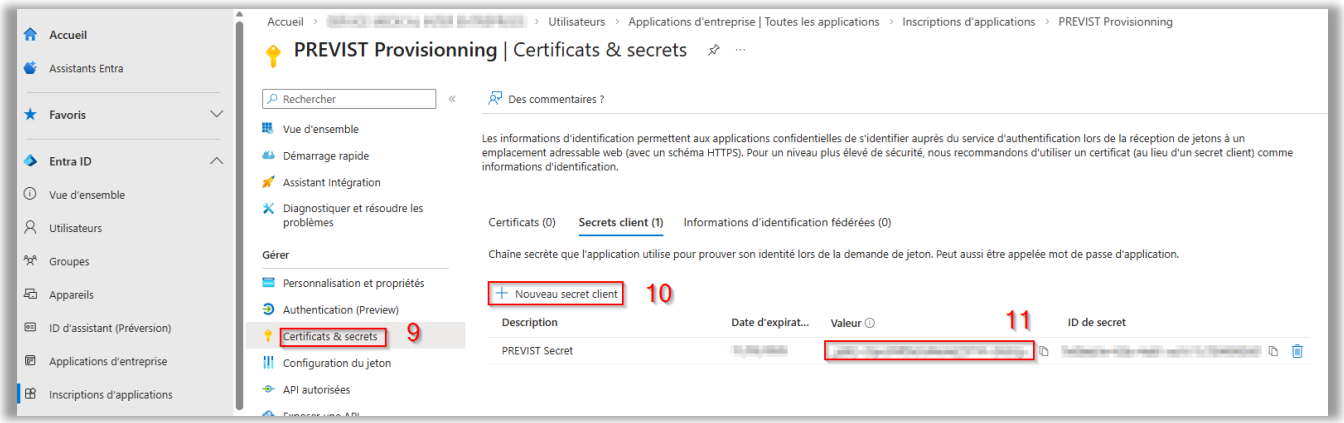


5- Menu *API autorisées*
6- Ajouter une autorisation – Microsoft Graph – Autorisation d'application.
7- Autorisations d'application à accorder :
   - *User.Read.All* (Application)
   - *Group.Read.All* (Application)
8- Accorder un consentement d'administrateur pour *<Tenant>*.



Ces permissions ouvrent la lecture des utilisateurs et des groupes au niveau annuaire pour la fonction.
Nb : la valeur « User.Read » (Déléguée) accordée par défaut n'est pas nécessaire.

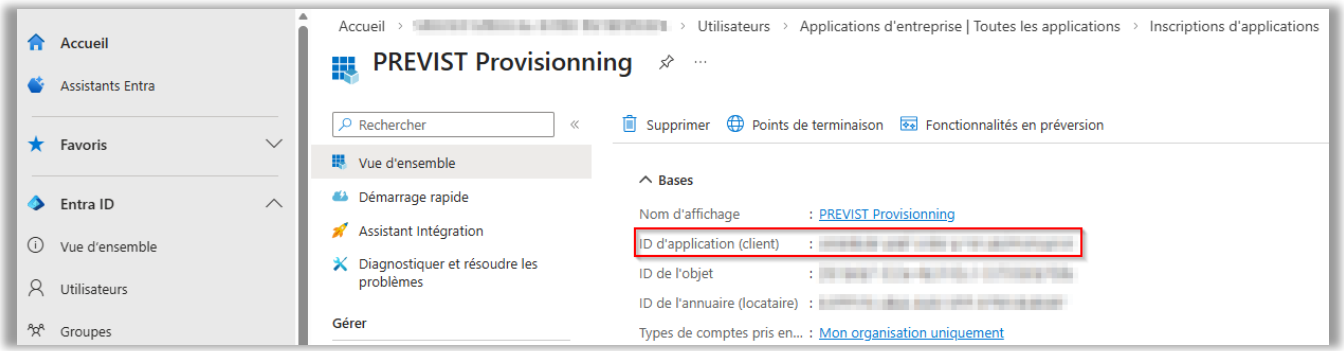
9- Menu *Certificats & secrets* - Secret client
10- Créer un Nouveau secret client – PREVIST Secret
Durée maximale désormais de 2 ans (Microsoft a supprimé les secrets « N'expire jamais »).
Cette clé ne sera utilisée qu'en interne. Elle sera donc à renouveler tous les deux ans sans intervention de PREVIST.
11- **Noter la Valeur : elle ne pourra plus être consultée par la suite.**



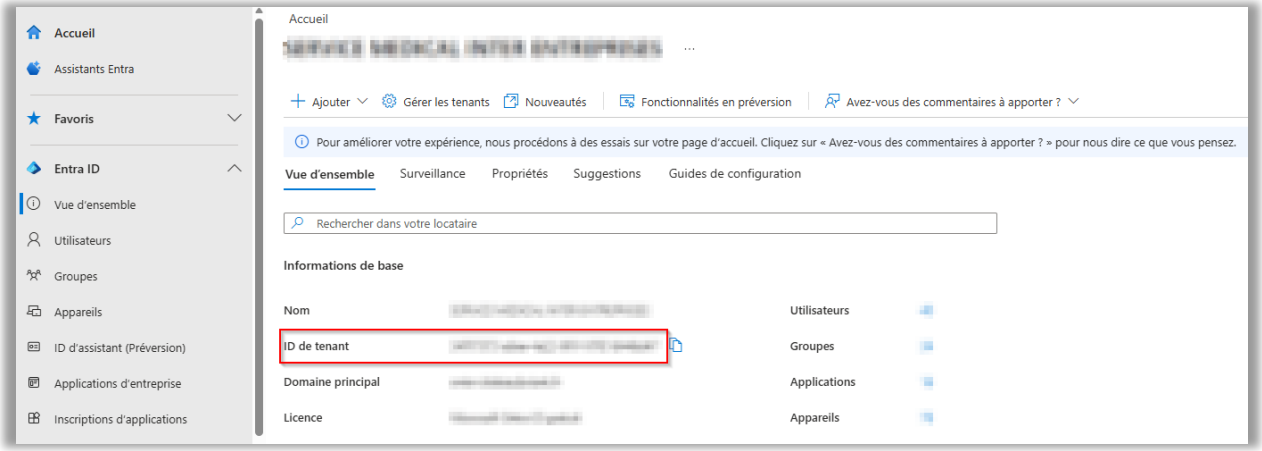
Nb : Il sera possible d'automatiser la rotation du secret (cf. §12).

Pour les chapitres suivants, retenir les valeurs suivantes qui seront à renseigner dans l'application web :

- **Valeur secret client** : cf point 11 ci-dessus
- **ID d'application (client)** : menu Vue d'ensemble de l'application
- ***Pour l'automatisation du renouvellement du secret** (§12, optionnel), retenir également l'ID de l'objet*



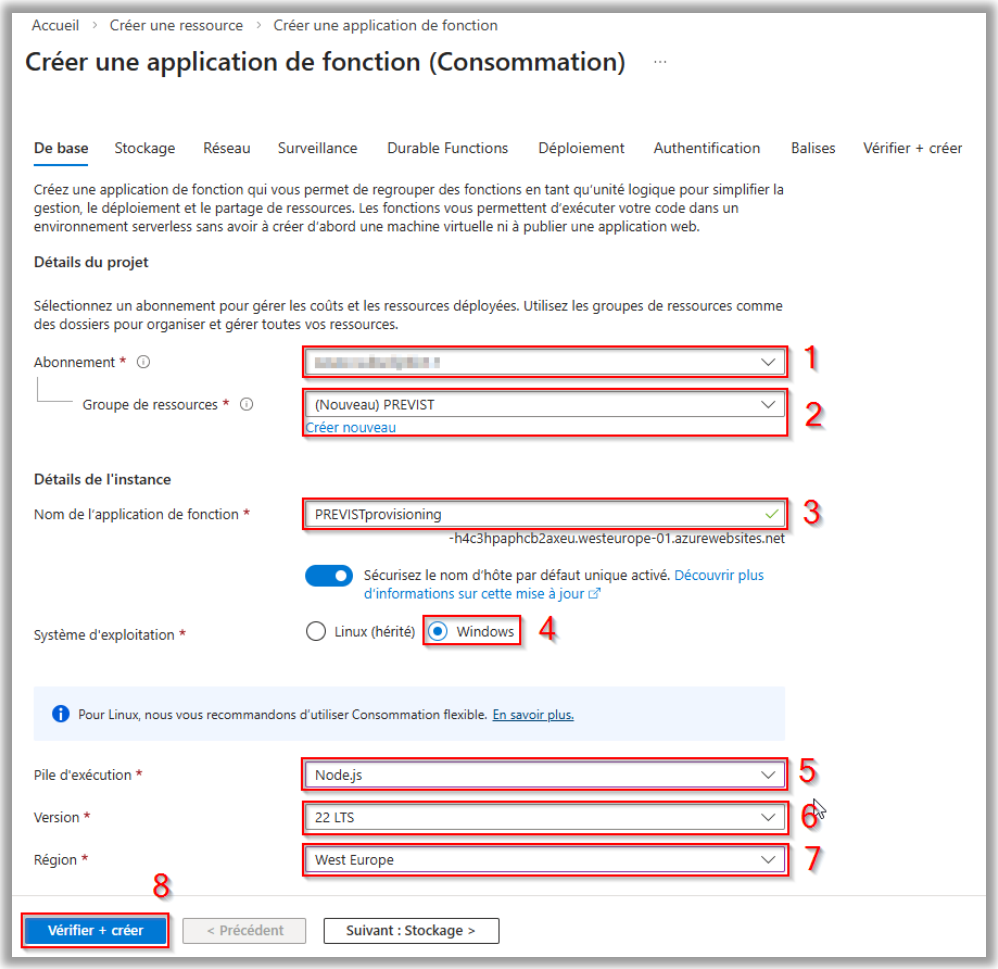
- **ID de Tenant** : menu Vue d'ensemble d'EntraID



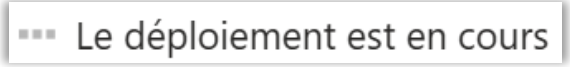
6. Création de l'Application de Fonction (Windows)

Créer une ressource sur le Portail Azure : Application de fonction
Sélectionner une option d'hébergement : Consommation (recommandé)

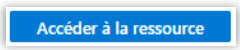
1- Abonnement : Choisir une suscription Azure en cours.
2- Groupe de ressources : Créer nouveau (recommandé : PREVIST)
3- Nom de l'application de fonction : previstProvisioning<SPST> (recommandé)
4- Système d'exploitation : Windows (recommandé)
5- Pile d'exécution : Node.js
6- Version : 22 LTS
7- Région : West Europe (recommandé)
8- Vérifier + créer - Créer



Nb : Patienter quelques instant le temps que l'application de fonction ainsi que ses dépendances se génèrent.



7. Paramètres & variables d'environnement

Depuis l'étape précédente, ouvrir 

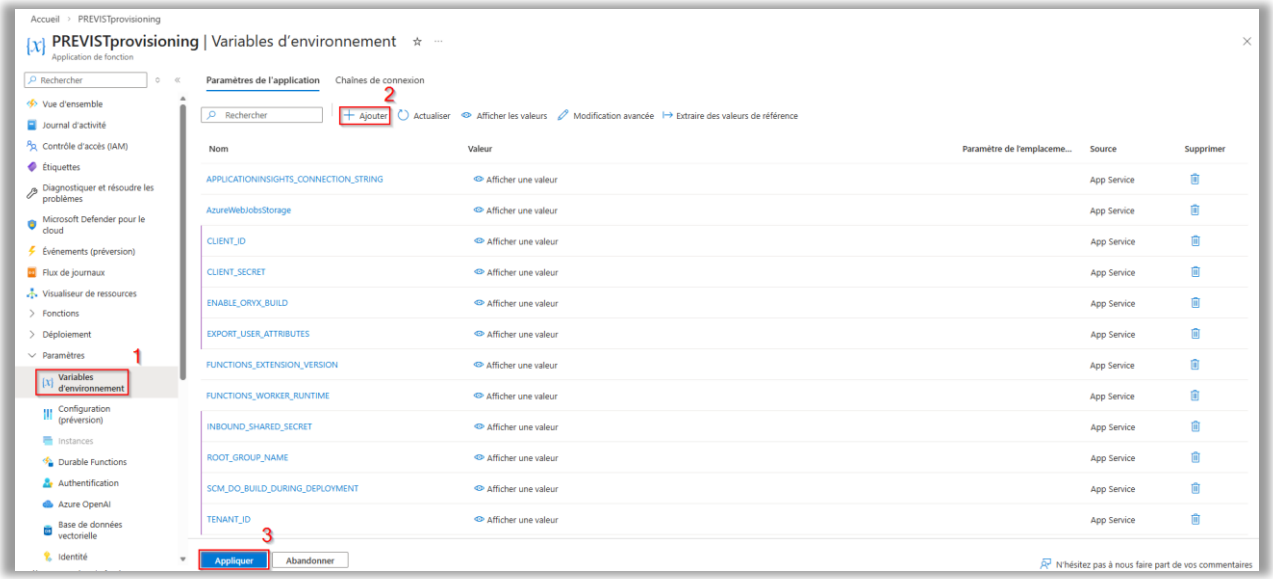
ou depuis le Portail Azure, ouvrir l'application 

précédemment créée.
Ajouter les variables suivantes dans Paramètres → Variables d'environnement.

Paramètres de l'application Microsoft Graph (retenus en *§5)* :
TENANT_ID = <ID de tenant>
CLIENT_ID = <ID d'application (client)>
CLIENT_SECRET = <Valeur secret>

Build (Oryx) pour déploiement Zip :
SCM_DO_BUILD_DURING_DEPLOYMENT = true
ENABLE_ORYX_BUILD = true
Remarque : Oryx est le builder App Service; le couple SCM_DO_BUILD_DURING_DEPLOYMENT + zip-deploy déclenche le build (npm install) côté Kudu. Selon historique de déploiement, Oryx peut s'invoquer même si désactivé ; nettoyer Kudu si comportement inattendu.

Paramétrage fonctionnel :
INBOUND_SHARED_SECRET : Clé de chiffrement « Secret » utilisée en header lors de l'appel de la fonction.
Cette valeur personnalisée sera à communiquer à PREVIST. (Caractères autorisés : « a-z », « A-Z », « 0-9 », « - »)
ROOT_GROUP_NAME : Nom du groupe racine (displayName) dans EntraID (recommandé GRP_PREVIST)
La fonction résout ce groupe par displayName et n'affiche que ses membres directs (utilisateurs) + tous les utilisateurs (directs + imbriqués) pour chaque sous-groupe direct. (cf *§2)*
EXPORT_USER_ATTRIBUTES : liste CSV d'attributs utilisateurs Graph à exporter en plus de id, userPrincipalName, surname, givenName (ex. mail,department,jobTitle).
Cette variable peut rester vide, mais doit être créée.



Appliquer les modifications et Confirmer.
Les variables présentes par défaut doivent être conservées en l'état.

8. Import de la Fonction Node.js (zip)

Importer le projet « previst-provisioning.zip » contenant la fonction node.js.

1- Déploiement - Centre de déploiement
2- Source : Publier des fichiers
3- Fichiers de publication : Cliquer sur Parcourir et sélectionner le projet previst-provisioning.zip fourni
4- Enregistrer
5- Journaux : Cet onglet permet de vérifier l'état du déploiement. Patienter que la colonne « Etat » passe en « Opération réussie (Actif) »

Optionnel : Vérifier les dépendances côté Kudu Console/SSH :

```
npm ls @azure/functions
npm ls @microsoft/microsoft-graph-client
npm ls @azure/identity
```

Ces packages sont utilisés par le handler HTTP et par le client Graph authentifié via Client Credentials.

NB : Des outils de diagnostics avancés sont disponibles dans le menu Outils de développement tels qu'une console (sans élévation), les outils Kudu+, et un éditeur de fichiers.

9. Integration PREVIST (et test appel HTTP trigger)

Sécurité de l'Endpoint (auth = function):
Le handler GetPrevistUsers est défini en HTTP GET, authLevel: 'function', route previst-users.
L'appel requiert deux clés (fonction ou host) via « x-functions-key » ou « ?code=* », et via « X-Previst-Secret » ou « &secret=* ».

Il est possible de tester l'appel de la fonction directement dans un navigateur par l'URL :
https://<app>.azurewebsites.net/api/previst-users?code=<FUNCTION_KEY>&secret=<INBOUND_SHARED_SECRET>

A cette étape, merci de fournir à PREVIST les éléments suivants :

- **<app>** & **<FUNCTION_KEY>** : Menu « Vue d'ensemble – GetPrevistUsers - Obtenir une URL de fonction – default (Clé de fonction) »
- **<INBOUND_SHARED_SECRET>** : Menu « Variables environment »

Ces éléments (*à titre informatif*) permettront d'intégrer les comptes filtrés dans PREVIST à l'aide de Workflows (côté PREVIST). Ces Workflows sont paramétrés dans l'IAM Okta.

Endpoint :
GET https://<app>.azurewebsites.net/api/previst-users

Headers:
x-functions-key: <FUNCTION_KEY> (ou ?code=<FUNCTION_KEY> en query)
X-Previst-Secret: <INBOUND_SHARED_SECRET> (ou &secret=<INBOUND_SHARED_SECRET> en query)

Exemple cURL:

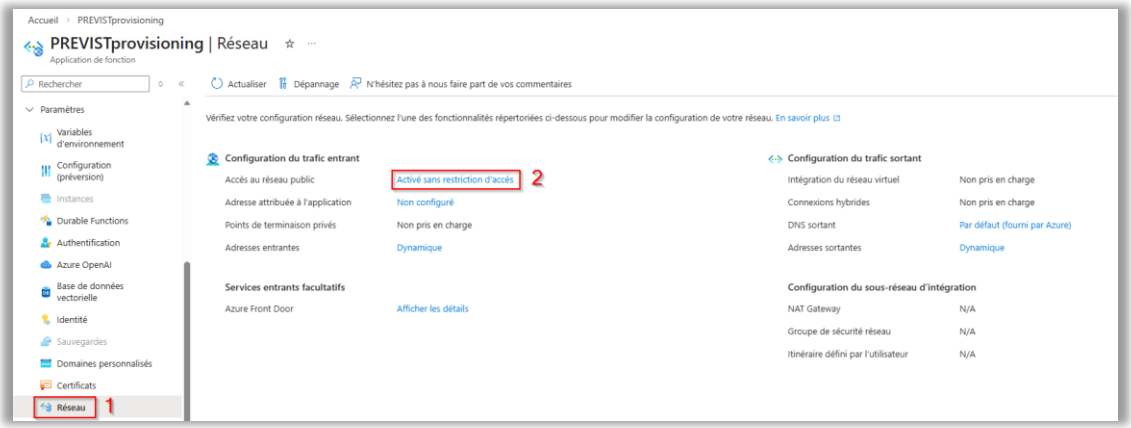
```
curl -s \  -H "x-functions-key: <FUNCTION_KEY>" \  -H " X-Previst-Secret: <INBOUND_SHARED_SECRET>“ \
"https://<app>.azurewebsites.net/api/previst-users"
```

Test query:
https://<app>.azurewebsites.net/api/previst-users?code=<FUNCTION_KEY>&secret=<INBOUND_SHARED_SECRET>

10. (Optionnel) Filtrage IP : Restriction d'accès

Pour mettre en place un filtre par IP afin de sécuriser les appels sur l'application de fonction (en plus nom d'hôte sécurisé, du header function, et du secret) :

1- Menu Paramètres – Réseau
2- Accès au réseau public : cliquer sur Activé sans restriction d'accès

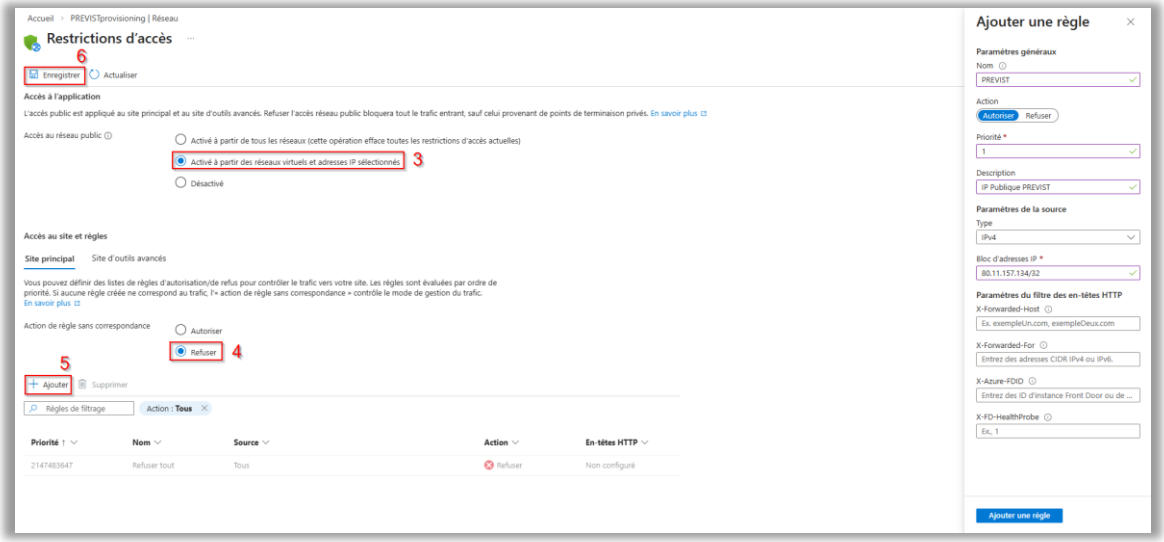


3- Accès au réseau public : Activé à partir des réseaux virtuels et adresses IP sélectionnés
4- Action de règle sans correspondance : Refuser, permettra de refuser la connexion de la fonction depuis toutes les IPs, sauf celles autorisées manuellement.

5- Ajouter la liste d'IPs suivantes :

Nom	Action	Priorité	Description	Bloc d'adresses IP
Okta_n	Autoriser	1	IPs Publiques Okta	Le pool d'IPs publiques d'Okta est disponible sur la page *Allow access to Okta IP addresses*. PREVIST est hébergé dans la cellule « Production Ireland : emea_cell_2 » composée de 180 adresses IP. L'outil de restriction ne permet d'ajouter des IP que par groupe de 8, il sera donc nécessaire de créer plusieurs pools pour autoriser la totalité des IPs d'Okta.
Admins PREVIST (optionnel)	*Autoriser*	*1*	*IP Publique Admins PREVIST*	80.11.157.134/32, 5.39.48.130/32 (*pour tests d'appels par les admins PREVIST*)
<SPST> (optionnel)	*Autoriser*	*1*	*IPs Publique du SPST*	Pool d'IPs publiques du SPST *(pour tests d'appels de la fonction en interne)*

6- Enregistrer, Accepter la mise à jour et Continuer



11. (Optionnel) Monitoring

Il est possible de créer des alertes de supervision depuis le menu Supervision – Alertes dans l'application de fonction.

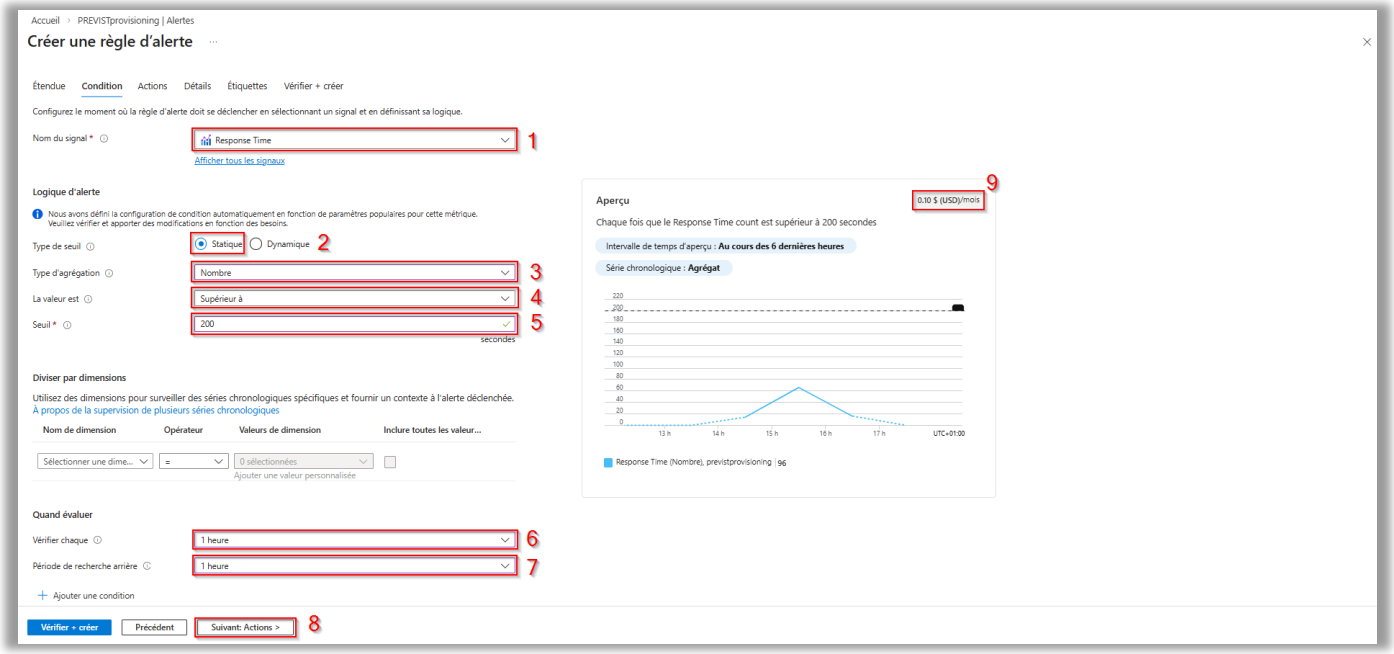
Des options plus poussées sont disponibles dans la ressource Application Insights PREVISTprovisioning générée automatiquement sur le Portail Azure.
Certaines de ces ressources de supervision sont facturées. Les tarifs sont ajustés et affichés en temps réels lors de la préparation des alertes. Si une offre de crédits non-profit est liée à l'abonnement en cours, ces alertes en seront déduites.

Métriques :
Le menu Supervision – Métriques permet de consulter les différentes statistiques sur l'utilisation de la fonction tel que le nombre d'exécution sur une période donnée (FunctionExecutionCount) ou les tentatives de connexion depuis une IP non autorisée (http 403).

Alertes personnalisées :
Pour monitorer le temps de réponse de la fonction (timeout par défaut non paramétrable de 240 secondes) et recevoir un mail en cas de dépassement d'un seuil paramétré à 200 secondes.
Depuis le menu Supervisions – Alertes – Créer.

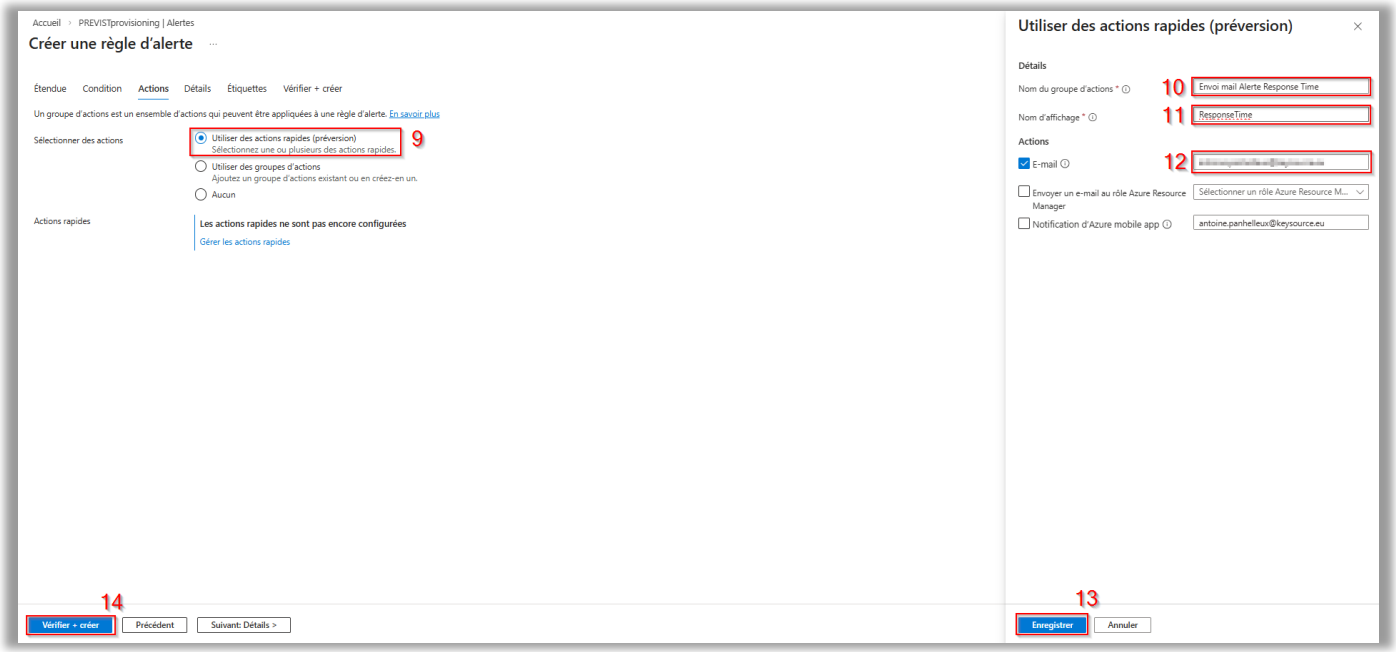
1- Nom du signal : Afficher tous les signaux - Response Time
2- Type de seuil : Statique
3- Type d'agrégation : Nombre
4- La valeur est : Supérieur à
5- Seuil : 200
6- Vérifier chaque : 1 heure
7- Période de recherche arrière : 1 heure
8- Suivant : Actions >

Nb : Un graph affichera en temps réel les résultats de cette alerte, ainsi que le tarif (10cts/mois) de la mise en place (9).



Onglets Actions :

9- Sélectionner des actions : Utiliser des actions rapides (préversion)
10- Nom du groupe d'actions : Envoi mail Alerte Response Time
11- Nom d'affichage : ResponseTime
12- E-mail : l'adresse mail recevant l'alerte
13- Enregistrer
14- Vérifier + créer



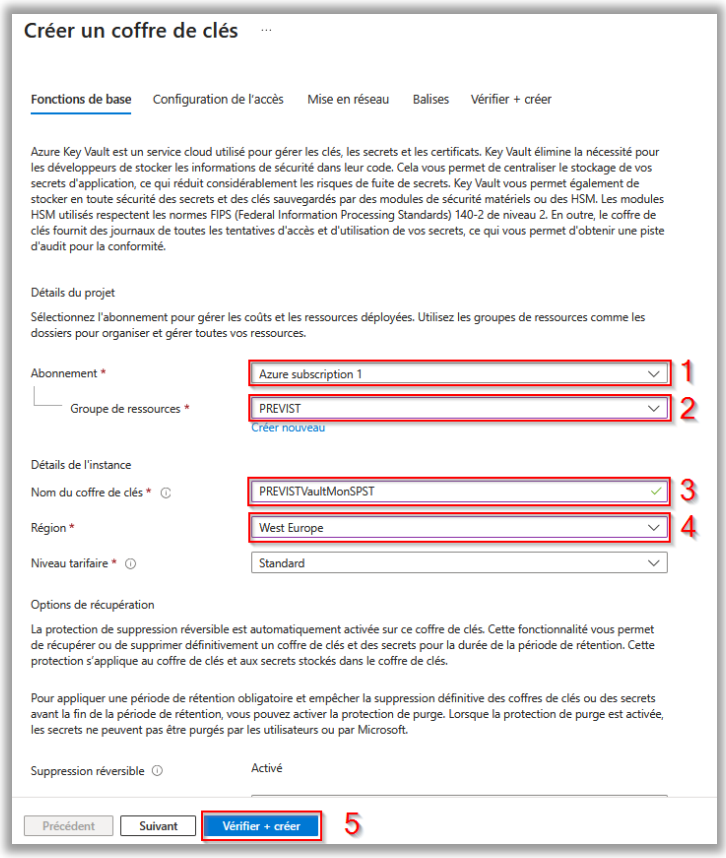
12. (Optionnel) Gestion & renouvellement du secret Graph

Le Secret de l'application Graph créé en §5 sera stocké dans un Key Vault sur le portail Azur. Ce secret sera appelé en lieu et place de la variable d'environnement <CLIENT_SECRET> paramétrée en §7.
Un compte Azure Automation renouvellera le secret dans un Runbook.

a- Création d'une ressource Key Vault Azure : (stockage du secret)

Créer une ressource sur le Portail Azure : Key Vault (ou Coffre de clés)

1- Abonnement : Choisir une suscription Azure en cours.
2- Groupe de ressources : PREVIST (recommandé)
3- Nom du coffre de clés : PREVISTVault<SPST> (recommandé)
4- Région : West Europe (recommandé)
5- Vérifier + créer – Créer



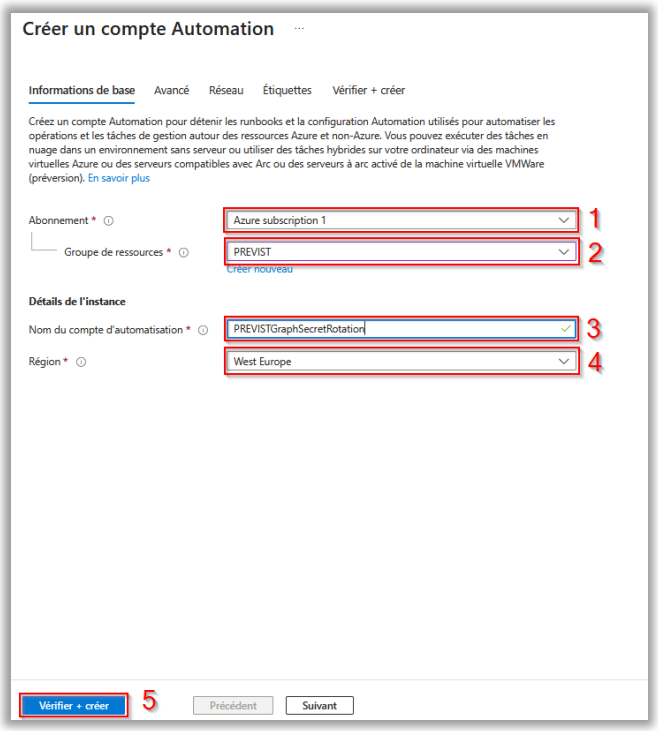
NB : Le compte utilisé pour créer le coffre sera automatiquement paramétré comme Propriétaire. Le rôle propriétaire ne donne pas l'accès aux ressources du Vault, mais la possibilité d'éditer ces accès. Il est donc nécessaire de s'accorder le rôle Agent des secrets Key Vault (cf « Accorder les permissions » ci-dessous).

b- Création d'un Compte Automation : (renouvellement du secret)

Ce compte aura accès à l'application MgGraph et au Key Vault et se chargera de renouveler le Secret et de le mettra à jour dans le Key Vault.

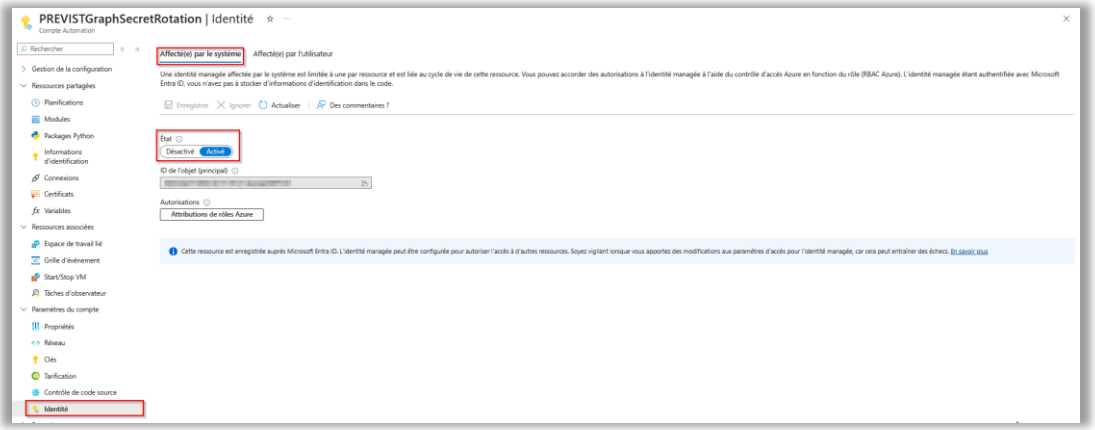
Créer une ressource sur le Portail Azure : Compte Automation 

1- Abonnement : Choisir une suscription Azure en cours.
2- Groupe de ressources : PREVIST
3- Nom du compte d'automatisation : PREVISTGraphSecretRotation (recommandé)
4- Région : West Europe
5- Vérifier + créer - Créer



c- Accorder les permissions :

➔ Depuis le Portail Azure, ouvrir le Compte Automation PREVISTGraphSecretRotation .
Dans le menu Paramètres du compte – Identité, onglet Affecté(e) par le système, vérifier que l'état soit activé.

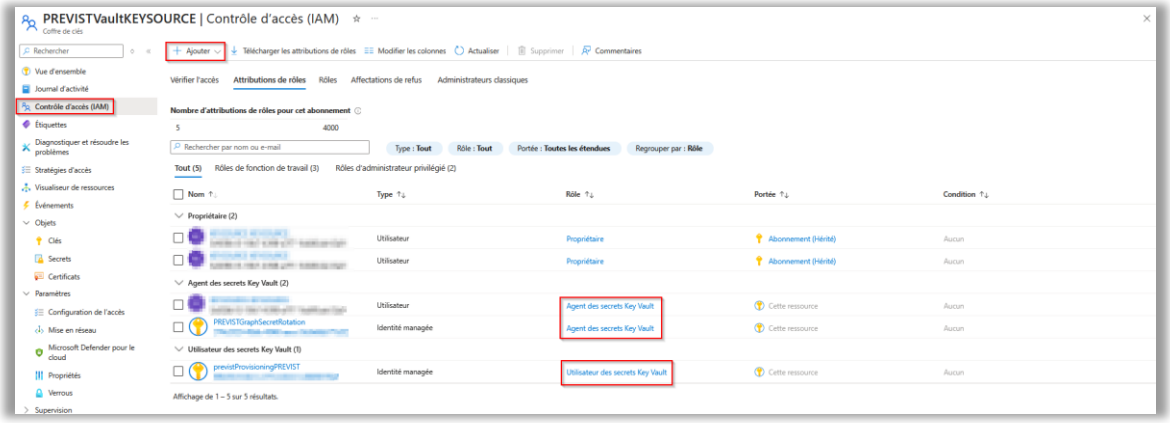


➔ Depuis le Portail Azure, ouvrir l'Application de Fonction PREVISTProvisioning<SPST>.
Dans le menu Paramètres – Identité, onglet Affecté(e) par le système, vérifier que l'état soit activé.

Nb : ce paramétrage permettra de donner les accès au secret enregistré dans le vault.

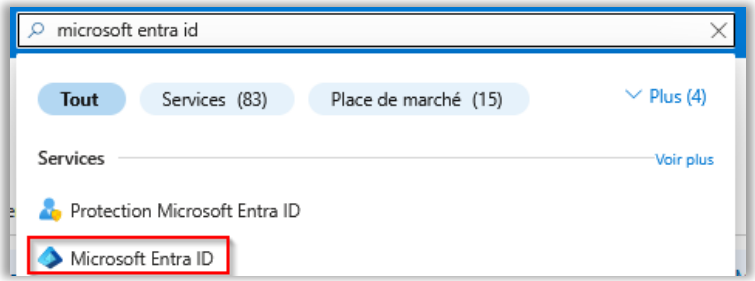
➔ Depuis le Portail Azure, ouvrir le Key Vault *PREVISTVault<SPST>* précédemment créée.
Dans le menu Contrôle d'accès (IAM) - Ajouter une attribution de rôle :

- Attribuer le rôle Agent des secrets Key Vault au compte actuellement utilisé ainsi qu'au compte Automation PREVISTGraphSecretRotation précédemment créé (optionnel : aux autres comptes pouvant nécessiter l'accès à ce secret).
- Attribuer le rôle Utilisateur des secrets Key Vault à l'Application de Fonction PREVISTProvisioning<SPST>

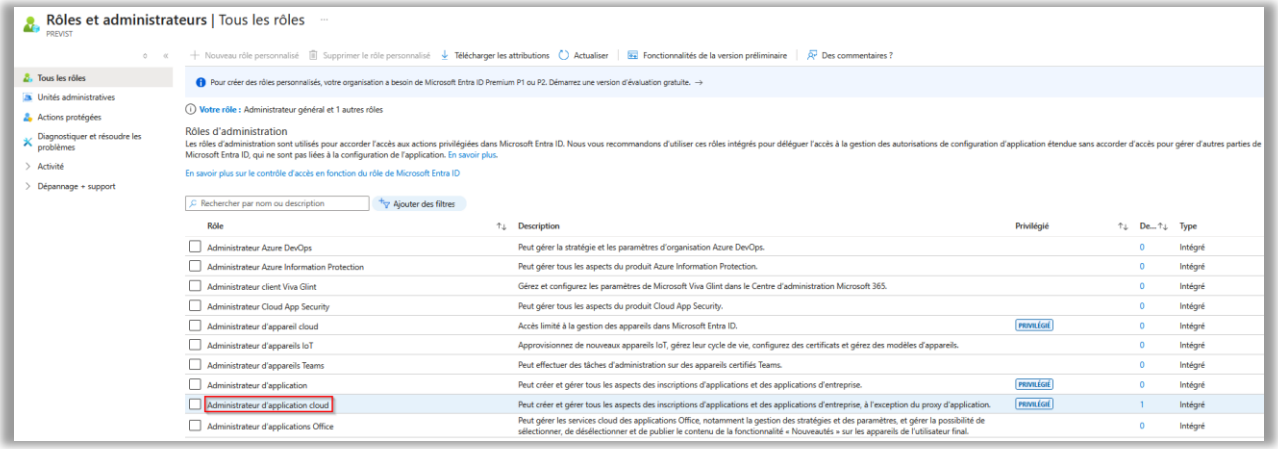


Nb : Ces rôles permettront l'écriture ou la lecture des secrets enregistrés dans le Vault.

➔ Depuis le Portail Azure, rechercher « Microsoft Entra ID » et l'ouvrir.



Dans le menu de gauche, choisir Gérer – Rôles et administrateurs. Ouvrir le rôle Administrateur d'application cloud et y ajouter le compte Automation PREVISTGraphSecretRotation précédemment créé.

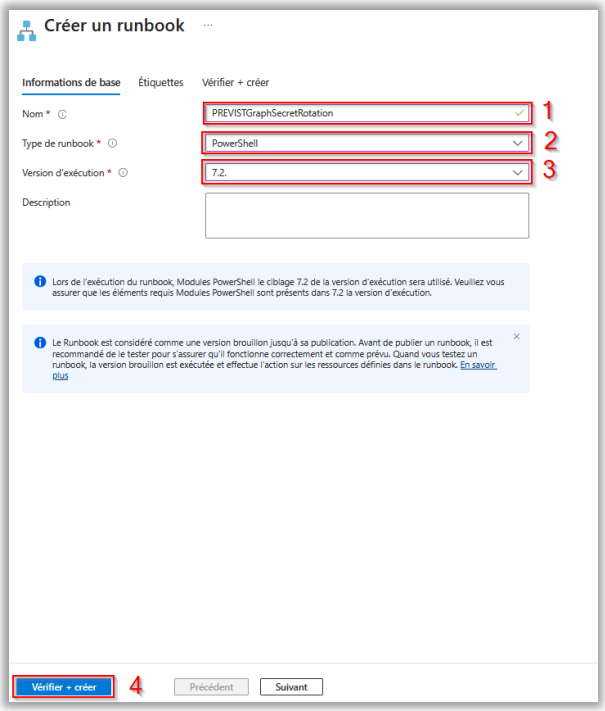


Nb : Cette option permettra au compte Automation de créer un secret dans l'application Graph.

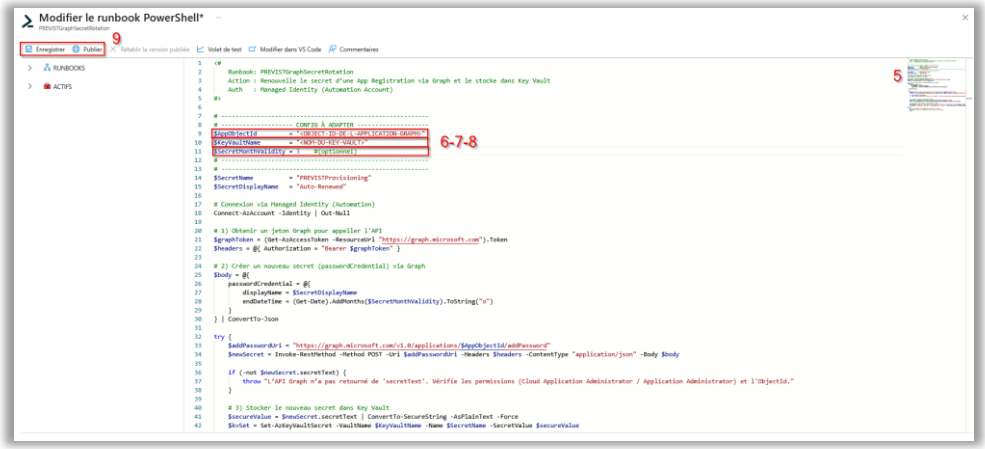
d- Création du Runbook

Depuis le Portail Azure, ouvrir le Compte Automation PREVISTGraphSecretRotation précédemment créé.
Dans le menu Automatisation des processus – Runbooks – Créer un runbook

1- Nom : PREVISTGraphSecretRunbook (recommandé)
2- Type de runbook : Powershell
3- Version d'exécution : 7.2
4- Vérifier + créer – Créer



5- Copier le contenu du script PREVISTGraphSecretRotation.ps1
6- Renseigner l'ID d'objet de l'application Graph retenu en §5
7- Renseigner le nom du Key Vault créé précédemment PREVISTVault<SPST>
8- (Optionnel) Modifier la période de validité du secret (en mois)
9- Enregistrer + Publier

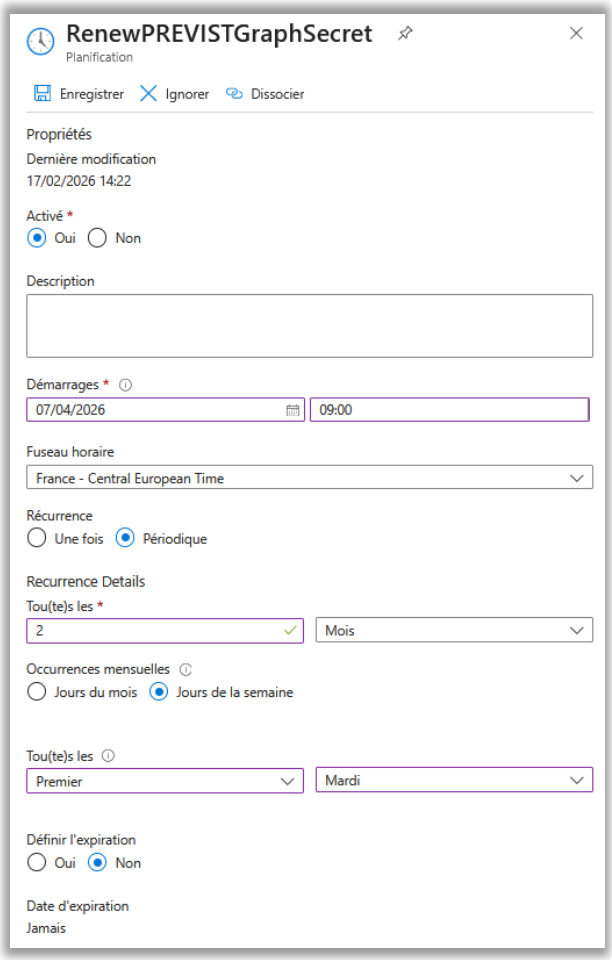


Nb : Dans le scenario par défaut, le secret expirera au bout de 3 mois.

e- Planification du Runbook

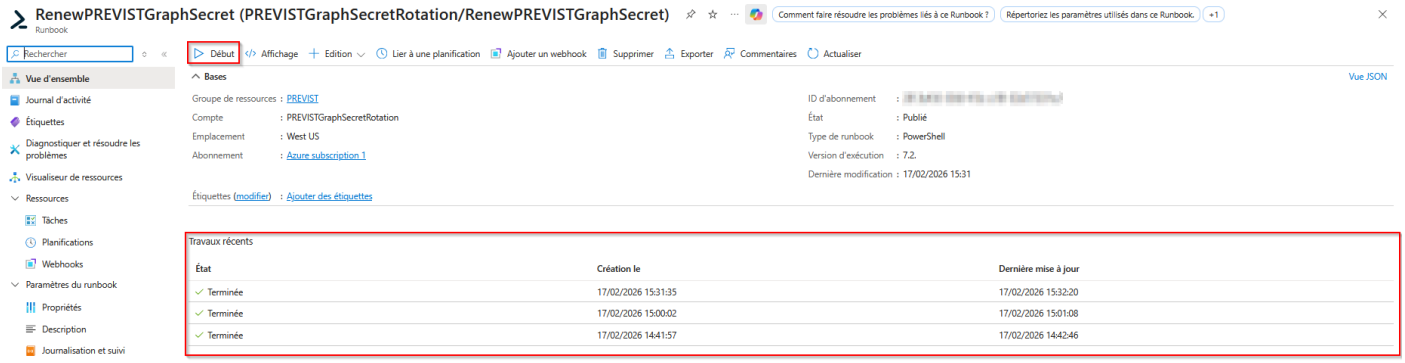
Dans le Runbook en cours, menu Ressources – Planifications – Ajouter une planification

Dans le scenario proposé, le secret expirant au bout de 3 mois, il est recommandé de planifier le renouvellement tous les 2 mois.
Dans cet exemple, le secret sera renouvelé à partir du 07 avril 2026, tous les 2 mois, le premier mardi du mois à 9h.



f- Test du Runbook

Dans le menu Vue d'ensemble du Runbook, cliquer sur Début
Les logs de sortie des exécutions manuelles et planifiées sont disponibles sur cette page.



À la suite du test, vérifier :

- Les logs de sortie de l'exécution
- La création du secret Graph « Auto-Renew » dans EntraID (cf §4)
- La création du secret dans le vault (cf point 2 ci-dessous)

g- Connecter la Fonction d'application au Key Vault :

Depuis le Portail Azure, ouvrir le Key Vault PREVISTVault<SPST> précédemment créée.

1- Dans le menu Vue d'ensemble, collecter l'URL du coffre / Vault.
2- Dans le menu Objets – Secrets, retrouver le secret généré automatiquement par le Runbook.

3- Dans l'application de fonction, remplacer la valeur de la variable <CLIENT_SECRET> (cf §7) par : @Microsoft.KeyVault(SecretUri=https://<URL_DU_VAULT>.vault.azure.net/secrets/PREVISTProvisioning/)

Le secret est désormais appelé depuis le Vault lors de l'exécution de la fonction. Un nouveau secret sera généré tous les deux mois, et mis à jour automatiquement dans le Vault.

13. Annexes

- ➔ *Annexe_Matrice_Metiers_Curebot.pdf : Liste des groupes de sécurité recommandés pour l'attribution des rôles métiers sur l'application finale Curebot.*
- ➔ *previst-provisioning.zip : Script Node.js prêt à uploader dans l'application de fonction. (cf §8)*
- ➔ *WebApp_previst-provisionning.zip : Version Web App du script Node.js pour installation sur VM on-premise.*
- ➔ *PREVISTGraphSecretRotation.ps1 : Script permettant la rotation du secret d'appel de l'application MgGraph. (cf §12)*
- ➔ *Notice générale PREVIST.pdf*
- ➔ *Notice Informatique PREVIST-V2.pdf*
- ➔ *Notice RGPD PREVIST_v2.pdf*