Evidian

Authentification de l'accès Web pour les Apps

Trusted partner for your **Digital Journey**

Authentification dans les applications Web

Pourquoi l'authentification et le contrôle d'accès ne devraient pas être gérés par les applications Web ?

Durant le développement d'une application Web, toute erreur de conception dans la gestion de l'authentification peut permettre de contourner le mécanisme d'authentification.

Récemment, une application Web bien connue fut exposée à une faille de sécurité. Elle était sécurisée par une authentification forte à deux facteurs. Malheureusement, il était possible de la contourner grâce à une API Web utilisée par la version mobile de l'application.

De nouveaux « designs patterns » permettent de développer sans distinction des applications Web et des applications mobiles, en utilisant des API qui exposent les mêmes fonctionnalités pour les deux mondes. Dans les deux cas, l'authentification et le contrôle d'accès doivent être correctement adaptés et gérés. Il doit être impossible de contourner les sécurités mises en place.

Développement Web

Pendant le développement de toute application Web, de nouveaux besoins apparaissent et un cycle d'évolution permanent est nécessaire pour créer et livrer de nouvelles fonctionnalités. Les développeurs expérimentés savent désormais sécuriser leur code. Les tests de sécurité des fonctionnalités demeurent un sujet capital, même si des méthodes agiles sont employées. De nouveaux mécanismes d'authentification peuvent être implémentés à tout moment, pour s'assurer que le bon utilisateur peut accéder aux fonctionnalités appropriées. Cependant, une nouvelle fonctionnalité peut à tout moment introduire une faille dans la sécurité de l'application. De nouvelles menaces apparaissent et la contrainte imposée par le délai de commercialisation ne crée pas un environnement favorable pour les traiter.

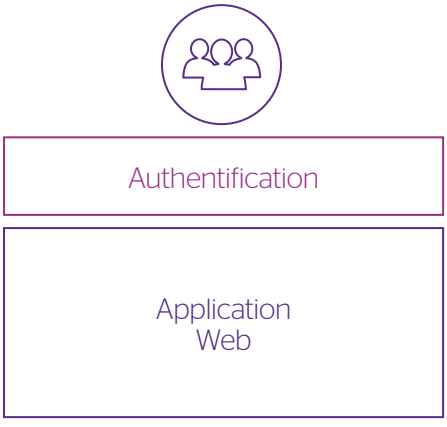
Il est nécessaire de garder trace de toute modification de sécurité, et d'implémenter le bon comportement au sein de l'application.

Les développeurs d'application doivent donc se souvenir des fonctionnalités à développer, et de la façon dont l'utilisateur authentifié y accède. Ils doivent comprendre l'impact des nouvelles méthodes d'authentification sur les fonctionnalités, développées et à développer. Ils doivent gérer deux contraintes majeures : développe rapidement de nouveaux services et garantir la sécurité de la totalité de l'application.

Seuls les développeurs très compétents peuvent mener leurs projets à terme en respectant toutes ces contraintes. Cependant, même eux peuvent introduire des faiblesses.

...mais vos applications vont évoluer :

- L'authentification n'est qu'une première étape, l'autorisation doit aussi être gérée.
- Tôt ou tard, vous aurez besoin d'intégrer de nouveaux moyens d'authentification.
- Vos employés devront-ils s'authentifier via les moyens propres au domaine Windows ?
- Vos utilisateurs externes auront-ils besoin d'un moyen d'authentification robuste tout en bénéficiant de la meilleur experience utilisateur ?
- Votre application est-elle compatible avec les utilisateurs fédérés ou l'authentification sociale ?
- De quels moyens d'auto administration vos utilisateurs pourront-ils disposer ?
- Comment pouvez vous garantir la capacité d'intégrer de nouveaux moyens d'authentification ?



Autorisation et authentification dynamiques

Gestion des autorisations dynamiques

Afin de centraliser la gestion des fonctionnalités auxquelles chaque utilisateur est autorisé à accéder, il faut externaliser le calcul et la gestion des autorisations. Par conséquent, le développement de l'application se concentre sur la création de nouvelles fonctionnalités sans avoir besoin de les relier à la gestion des droits des utilisateurs. L'avantage de cette approche, c'est qu'elle « libère » le développeur des contraintes liées à l'autorisation.

L'autorisation dynamique est confiée à un composant externe dédié, le serveur d'autorisation. XACML est le langage qui permet de l'interfacer, mais toute API fournissant le même service peut être employée. Cette approche favorise une architecture conceptuelle simple, en utilisant des blocs de construction dédiés à chaque fonction. Les applications interrogent simplement le point d'autorisation central pour savoir si l'accès est accordé, suivant l'identification et l'environnement de l'utilisateur (lieu, heure, etc.).

Néanmoins, pour interagir directement avec le mécanisme d'autorisation dynamique, une application doit connaître et comprendre les moyens d'authentification des utilisateurs. Son authentification est-elle forte ? Quel est son niveau ? Dans ce contexte, la gestion et l'ajout de nouveaux moyens d'authentification ou la discrimination des utilisateurs suivant leurs moyens d'authentification se fait en général au niveau de l'application. Toutefois avec cette méthode vous courez le risque que des fonctionnalités, nouvelles ou anciennes, ne prennent pas en compte tous les scénarios possibles. Pour réduire ce risque, la gestion de l'authentification doit être déléguée à un composant dédié : le gestionnaire d'authentification dynamique.

Gestion des authentifications dynamiques

Au sein d'une même application Web, ou dans une collection d'applications Web et d'APIs Web, il peut y avoir différents niveaux de confiance nécessitant une authentification simple ou forte. La façon dont les utilisateurs sont authentifiés est associée avec un niveau d'authentification allant du plus faible au plus fort. Par exemple, un service bancaire peut nécessiter une authentification simple, par identifiant et mot de passe, pour des opérations élémentaires, mais recourir à une authentification à deux facteurs (2FA) lorsque les clients accèdent à leurs comptes bancaires et leurs transactions.

L'authentification à plusieurs niveaux est une façon d'autoriser l'utilisateur à effectuer les opérations, dynamiquement, en fonction du contexte dans lequel il travaille en lui demandant de prouver son identité en utilisant le(s) moyen(s) d'authentification requis. Dans ce contexte des authentifications suivantes pourront avoir des significations et niveaux différents: authentification sociale, identifiant et mot de passe, mot de passe à usage unique (OTP) ou Kerberos.

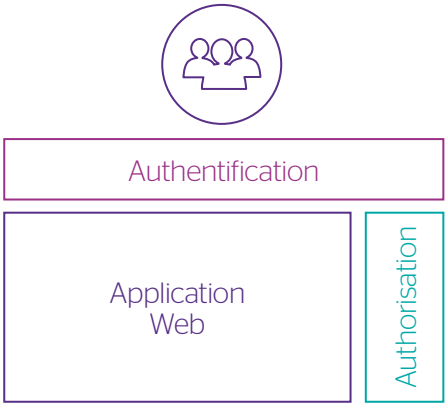
Suivant l'utilisateur, son environnement de connexion, ses moyens d'authentification et les services ou applications Web auxquels il voudrait accéder, le gestionnaire d'authentification dynamique lui proposer une liste de méthodes d'authentification appropriées. Les utilisateurs n'auront qu'à choisir une méthode d'authentification, suivant les moyens dont ils disposent. Associé à cette méthode, le niveau déterminera aussi les ressources et les méthodes HTTP autorisées.

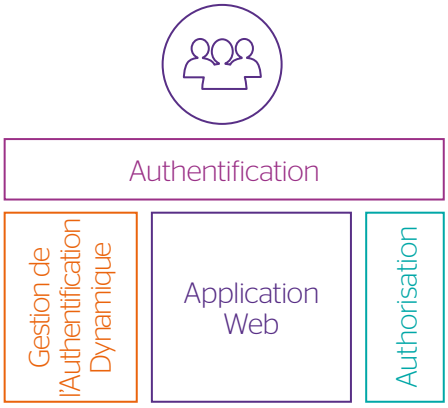
Niveau d'authentification

Par exemple, un utilisateur venant d'internet peut avoir à choisir entre une authentification sociale ou une authentification à deux facteurs (2FA). Le même utilisateur venant de l'intranet peut n'avoir d'autre choix que d'utiliser Kerberos. Les utilisateurs employant une méthode à deux facteurs peuvent avoir les mêmes droits que ceux employant Kerberos, alors que ceux utilisant une authentification sociale n'ont qu'un accès limité aux ressources. Dans cet exemple, les méthodes Kerberos et 2FA sont du même niveau, tandis que l'authentification sociale est d'un niveau inférieur.

Les applications Web doivent connaître le niveau d'authentification de l'utilisateur pour chaque ressource consultée, et donner cette information au composant de gestion d'autorisation dynamique.

Le développement et l'intégration de la sécurité s'en trouvent simplifiés et rendus indépendants. Néanmoins, comment intégrer ces deux composants dédiés à l'authentification et à l'autorisation dans les applications existantes ? Comment « libérer » les applications déjà développées et comment intégrer ces deux composants de façon harmonieuse ? L'introduction d'un gestionnaire d'accès Web (Web Access Manager) devient la meilleure façon de séparer les composants précédents de l'application Web à développer, ou déjà développée.





Gestion des accès Web (WAM)

Gestion d'accès Web (WAM)

Un gestionnaire d'accès Web (Web Access Manager) intègre la gestion de l'authentification dynamique et la gestion des autorisations dynamiques tout en protégeant et en masquant les ressources protégées d'une application Web.

En agissant comme un reverse-proxy, WAM est capable d'effectuer les phases d'authentification et d'autorisation avant de donner accès aux ressources des applications Web. Les applications existantes peuvent ainsi être intégrées en utilisant un mécanisme d'authentification unique sans modification dans ces applications.

Après une authentification primaire, WAM injecte l'identifiant et le mot de passe secondaire dans le mécanisme d'authentification existant dans l'application. Une coopération simplifiée peut être déployée pour les applications en développement où WAM ne transmet que l'identification des utilisateurs. Ainsi, il « libère » l'application de la gestion des mécanismes d'authentification, du niveau d'authentification et de tout scénario de ré-authentification. WAM identifie les ressources sollicitées et détermine lui-même quel niveau d'authentification est nécessaire et en attribue les modalités d'accès.

WAM à l'état de l'art

La méthode d'authentification et la façon dont les utilisateurs sont identifiés seront améliorées et modifiées suivant l'évolution des besoins, des exigences ou des usages. WAM pourra-être configuré sans impact sur les applications qu'il protège.

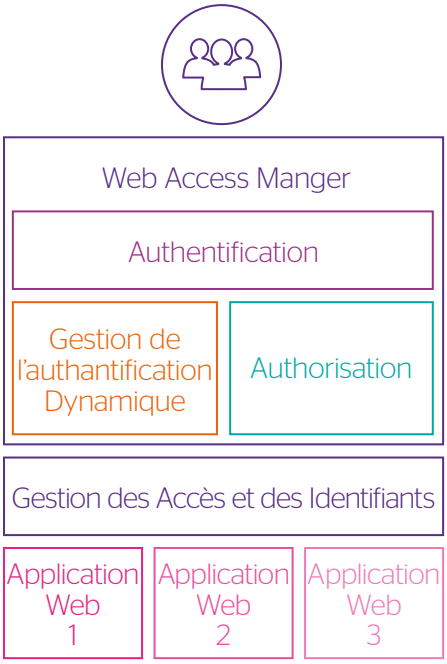
Ces applications protégées peuvent suivre leur propre cycle de développement. De nouvelles fonctionnalités peuvent être ajoutées ou modifiées sans avoir à modifier l'authentification ou l'identification des utilisateurs. De nouvelles populations d'utilisateurs sont ajoutées, de nouveaux référentiels utilisateurs sont configurés dans WAM pour gérer de nouveaux partenaires, utilisateurs externes ou utilisateurs sociaux (Facebook, Google, Twitter, ...).

Le niveau de sécurité de l'accès Web ne dépendra que d'un seul composant : le WAM. Les applications Web sont protégées sans contournement de la sécurité, même si de nouveaux services sont développés et déployés.

Fonctionnalités requises

Un gestionnaire des accès Web doit disposer des fonctionnalités suivantes :

- Gestion de plusieurs référentiels utilisateurs.
- Administration centralisée de la gestion des autorisations et de leurs règles.
- Règles d'accès, suivant les ressources, la localisation, le groupe ainsi que les attributs de l'utilisateur.
- Règles d'accès aux applications acceptant plusieurs niveaux d'authentification.
- Authentification forte grâce à un OTP logiciel ou matériel, avec ou sans réseau cellulaire.
- Vaste panel de méthodes d'authentification, dont celles à plusieurs facteurs ou chainées.
- Protocoles de fédération d'identité, pour les applications cloud externes ou internes.
- Intégration simplifiée dans l'infrastructure existante, Microsoft AD, serveurs RADIUS ou tout équipement de réseau (répartition de charge, reverse-proxys, pare-feu, WAF, ...).
- Portail de service avec auto-enregistrement des utilisateurs et self-service de déblocage (réinitialisation de mot de passe sécurisée, ...).
- Catalogue des comptes et des services protégés pour chaque utilisateur.
- Gestion et sécurisation des flux vers les applications protégées.
- Personnalisation et intégration complète selon la charte graphique souhaitée ou celle des applications existantes.



Web Access Manager constitue le point d'accès unique pour protéger toutes vos applications Web.

Web Access Manager vous apporte la sécurité, sans module supplémentaire, ni côté navigateur, ni côté serveur. Il supporte toutes les applications et n'est pas dépendant du navigateur Web. Il fournit une architecture pour un contrôle d'accès Web flexible. Les protocoles de fédération sont aussi utilisés pour fournir un SSO aux Apps dans le cloud, ou pour utiliser des identités externes provenant de fournisseurs d'identité en SaaS.

Web Access Manager permet aux entreprises de bâtir une plate-forme d'e business sécurisée, avec un seul point d'autorisation et de gestion de l'accès sécurisé à leurs ressources Web en filtrant l'accès des utilisateurs suivant leurs profils.

Web Access Manager en action

Un scénario WAM typique se déroule comme suit :

- Un utilisateur tente d'accéder à une ressource, c.-à-d. une URL protégée.
- WAM détermine le niveau d'authentification nécessaire pour obtenir l'accès.
- Si l'utilisateur n'est pas authentifié, est inconnu au moment de l'accès ou si son niveau d'authentification est trop bas pour cet accès, WAM lance la procédure d'authentification.
- Les méthodes d'authentification proposées dépendront de l'environnement de connexion, de l'utilisateur et de la ressource sollicitée.
- Une fois l'utilisateur identifié et authentifié, WAM analyse les règles d'autorisation. De plus, WAM peut s'appuyer sur des mécanismes externes de postauthentification et de post-autorisation pour déterminer si l'utilisateur peut accéder à la ressource.
- La post-autorisation et la postauthentification peuvent être redirigées vers un point de décision dédié, afin d'outrepasser les règles calculées par Web Access Manager.
- WAM requiert les ressources du serveur protégé, et peut effectuer un Single Sign- On (SSO) en injectant les données d'authentification ou les informations d'identité de l'utilisateur dans la fenêtre de connexion ou dans l'entête HTTP.
- Avant d'envoyer la ressource à l'utilisateur, Web Access Manager peut injecter des données, retirer des informations, réécrire les URLs ou injecter du code HTML/CSS/ javascript, afin de créer une vue externe d'une ressource interne. En effet, WAM est capable d'inspecter la ressource en profondeur, de faire l'historique du trafic et de générer des événements d'audit des interactions de l'utilisateur.

L'introduction d'une nouvelle méthode d'authentification, à deux facteurs ou sociale, est désormais facile et complètement indépendante des applications protégées. Il est recommandé d'accompagner l'utilisation d'un Web Access Manager par :

- La mise en place d'un d'un composant de gestion des identités et des accès afin d'identifier les populations d'utilisateurs et d'allouer les comptes dans les applications protégées en vue d'un SSO.
- Une identification claire de toutes les URL visibles de l'application à protéger, et leur niveau de confiance associé.

Gestion des identités et des accès

Le gestionnaire d'identité et d'accès est connecté au Web Access Manager. Il fournit les fonctionnalités nécessaires pour que l'utilisateur accède aux applications protégées, en allouant les comptes secondaires dans les applications existantes ou en réglant les attributs dans le profil de l'utilisateur, pour le mécanisme d'autorisation dynamique.

Web Access Manager expose les parties publiques du gestionnaire d'identité, en particulier les interfaces Web dédiées aux utilisateurs qui sont protégées par le WAM. Ces interfaces sont utilisées pour activer les moyens d'authentification et gérer les requêtes d'accès à certains services et applications.

Tandis que le gestionnaire d'identité et d'accès provisionne les comptes secondaires dans les applications existantes, il les alloue aussi dans le Web Access Manager. L'utilisateur bénéficie de fonctionnalités SSO et seule l'authentification primaire doit être connue. De la même façon, quand un utilisateur ne dispose plus de l'accès à des applications, le Web Access Manager en est immédiatement informé, éliminant les risques de compte rémanent indésirable.

Impact sur le processus et les responsabilités du développement

Avec l'adoption d'un composant de Web Access Management, le besoin de gérer les moyens et les mécanismes d'authentification disparaît pour les développeurs. Les applications reçoivent l'identification des utilisateurs et leur niveau d'authentification en vigueur.

Basée sur cette identification, la logique de l'application demeure identique. Le niveau d'authentification peut ne pas être employé à des fins de sécurité, mais seulement pour afficher les liens corrects ou des informations quant à l'utilisation du mécanisme d'authentification, la réinitialisation du mot de passe ou la personnalisation des pages affichées.

Les équipes de développement restent concentrées sur les fonctionnalités, la feuille de route et la sécurité du code, et ne se préoccupent plus de l'authentification des utilisateurs. Ils doivent identifier et énumérer toutes les URLs des serveurs Web utilisés pour produire les fonctionnalités. Les risques de chaque série d'URLs doivent être évalués, pour leur associer un niveau d'authentification. Puis, l'équipe de gestion de l'accès Web crée les services en employant les séries d'URLs, et décide des règles et du niveau d'authentification de chacun de ces services. Les URLs non recensées ne peuvent tout simplement pas être atteintes.

Les responsabilités liées au développement Web et à l'authentification sont rendues distinctes. L'une peut évoluer sans aucune influence sur l'autre.

La gestion des droits des utilisateurs, des authentifications et des accès ne concerne plus du tout les applications. La gestion de l'identité et de l'accès devient la tour de contrôle de la gestion des utilisateurs. L'introduction de nouvelles méthodes d'authentification, la publication de nouveaux services, le déploiement de nouveaux moyens d'authentification, ou l'identification via un tiers tel qu'un réseau social mettant en oeuvre des protocoles standards d'échanges (SAML, Oauth, OpenID, ...), relèvent des gestionnaires des identités et des accès.

Conclusion

L'indépendance des composants et la répartition des responsabilités d'équipe assurent qu'il n'existe aucun moyen de contourner les authentifications fortes. Cela simplifie le processus de développement et allège une partie des contraintes de sécurité pour les équipes de développement.

À tout moment, la robustesse de l'authentification peut être accrue sans influencer les fonctionnalités déjà développées.

La gestion de l'authentification dynamique et la gestion de l'autorisation dynamique assurent l'adaptabilité et l'évolution du projet de Web Access Management pour de nouveaux besoins et scénarios.

Et pour les Apps mobiles ?

Web Access Manager protège vos serveurs d'applications mobiles

Applications mobiles Web

Les applications Web peuvent être utilisées telles quelles sur un appareil mobile. De plus en plus d'applications Web respectent le « responsive-design » et sont immédiatement utilisables sur ces appareils, sans modification.

Toutes les interfaces Web de WAM sont compatibles avec les principes du « responsive-design ». Une application Web protégée par WAM peut donc être utilisée sur mobile. Les utilisateurs peuvent employer n'importe quelle méthode d'authentification, accéder à leur profil de compte, réinitialiser leur mot de passe, voir la liste des services accessibles, etc... Toutefois, l'utilisation d'applications Web sur un navigateur mobile présente un gros inconvénient : l'expérience utilisateur est moins satisfaisante qu'avec une application native.

Applications mobiles natives

Pour développer une application dédiée aux mobiles, il existe deux approches :

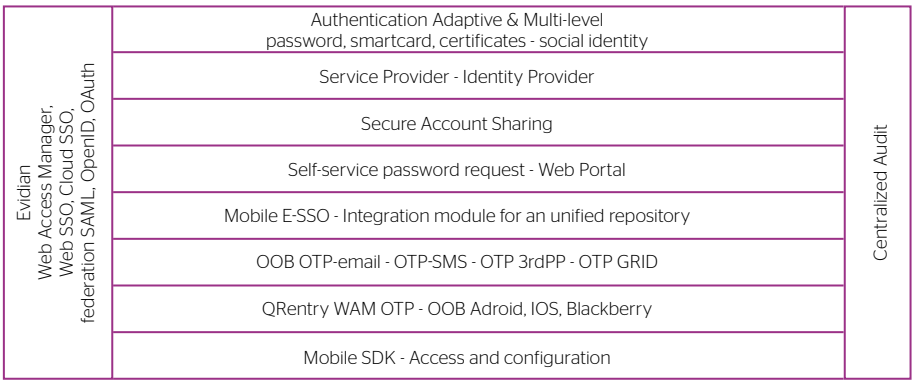
- Développer une application native en utilisant le langage de développement natif et le SDK de la plate-forme. Ce type d'application sera à même d'exploiter tous les points forts des appareils, mais vous devrez refaire le développement pour chaque système supporté.
- Une application native-hybride utilisant à la fois le langage natif pour le système sous-jacent, et HTML5/Javascript pour l'affichage et l'interface. Ce type d'applications est plus puissant que les applications Web pures, mais l'expérience utilisateur peut là encore, être limitée en comparaison à des applications entièrement natives, même si des frameworks permettent d'intégrer facilement les fonctionnalités de chaque plate-forme.
- Dans les deux cas, le mécanisme d'authentification doit être développé en gardant la sécurité à l'esprit. Les développeurs peuvent introduire des faiblesses en tentant de contourner des fonctionnalités de sécurité déjà en place pour protéger les API publiées. Cela entraîne des failles de sécurité pour toute l'application Web.

Il est donc important d'utiliser les mêmes contrôles d'accès et la même sécurité dans les applications mobiles natives et les applications Web qui emploient les mêmes APIs en back-end.

SDK mobile de WAM

L'objectif principal d'un SDK mobile natif est d'aider les développeurs à intégrer de nouvelles fonctionnalités sans redéfinir tous les mécanismes.

Le SDK mobile de WAM apporte toutes les fonctionnalités nécessaires pour interagir entre une application mobile native pure ou native hybride, et les APIs Web protégées. Ces APIs sont les mêmes que celles utilisées par les applications Web, comme décrites précédemment. Les APIs sont principalement des appels REST vers un serveur Web back-end, ou des appels de Web-services ou d'URL dédiées.



Buts du SDK de WAM :

- Fournir un mécanisme d'authentification offrant le même niveau de confiance que celui de la version Web d'une application
- Gérer l'authentification, à plusieurs niveaux ainsi que son expiration afin d'offrir une expérience d'authentification fluide à l'utilisateur
- Fournir les informations de configuration nécessaires à l'application mobile, permettant de découvrir le périmètre du WAM et les URLs des services protégés à utiliser
- Fournir des mécanismes de communications, faciles à développer, entre l'application mobile et le WAM qui protège le back-end.

...et l'accès au back-end des APIs

Les méthodes d'authentification sont toujours définies dans Web Access Manager.

Le SDK de WAM aide à créer facilement des applications qui communiquent avec les serveurs back-end protégés par Web Access Manager. Tout changement dans la façon dont les utilisateurs doivent s'authentifier sera répercuté dans l'interface de l'application mobile. Si l'application mobile tente d'accéder à une URL interdite, ou si une URL restreinte exige un plus haut niveau d'authentification, alors l'application mobile ne sera pas autorisée à récupérer l'URL, ou elle lancera une procédure d'authentification de plus haut niveau.

Web Access Manager constitue le seul point d'accès, pour protéger vos applications Web, et le back-end de vos APIs pour vos applications mobiles.

Le SDK de Web Access Manager fournit des fonctionnalités de sécurité indispensables, allant d'une authentification forte à la sécurisation des communications entre l'application mobile et l'API protégée.

Tout changement dans vos règles de gestion des accès sera répercuté dans le contrôle d'accès mobile et Web de vos applications. Les contournements de sécurité ne seront plus permis.

Gestion des identités et des accès

Avec I&AM et WAM, vous disposez de produits et de fonctionnalités complémentaires

Identity & Access Manager

Identity & Access Manager permet de gérer le cycle de vie complet des identités et des accès. Il dispose d'un moteur de politique de sécurité pour définir la matrice des droits d'accès des utilisateurs, et un portail permettant aux utilisateurs finaux de gérer leurs informations d'identité et leurs demandes de droits. Il permet une administration déléguée, comprend un ensemble de processus(workflows) d'approbation prêt à l'emploi, ainsi qu'un catalogue de connecteurs de provisionnement pour les applications, y compris les applications cloud (SaaS).

L'éditeur de workflows permet aux utilisateurs de créer de nouveaux processus de travail ou de modifier la configuration d'un processus de travail délivré par Identity & Access Manager. Identity & Access Manager peut provisionner nativement les produits SSO comme Web Access Manager.

Evidian Identity & Access Manager modélisent facilement vos politiques de sécurité avec des rôles qui sont compris par les responsables opérationnels. Des règles automatiques et dynamiques suivent les changements des caractéristiques des utilisateurs, et assignent ou retirent les droits d'accès en conséquence. Un espace de simulation (sandbox) permet d'éviter les risques engendrés par de telles opérations. Une intégration efficace avec le contrôle d'accès Evidian signifie que tout changement de droits d'accès est immédiatement répercuté dans le SSO, et les informations d'accès de l'utilisateur complètent alors les informations utilisables par le module de reporting.

I&AM simplifie la gestion des droits d'accès aux ressources et accélère leur mise à disposition ou modification. Les utilisateurs sont rapidement opérationnels et respectent naturellement la politique de sécurité et les contraintes de régulation.

...combiné avec Web Access Manager

Web Access Manager fournit toutes les fonctionnalités nécessaires pour gérer les authentifications, les autorisations, l'interface de l'utilisateur final et un jeu de modules self-service dédiés à l'activation et la réactivation de l'authentification, à la perte de mot de passe.

WAM fournit des fonctionnalités comme le suivi et le pistage de toutes les requêtes d'utilisateur, l'administration de plusieurs annuaires et l'auto-enregistrement, les rôles d'utilisateur à administration multiple, l'auto-allocation ou le partage des comptes secondaires.

Web Access Manager est un produit indépendant et complet dédié à la gestion des accès Web, incorporant les flux d'interaction des utilisateurs. Cependant, quand la population devient plus complexe, avec des utilisateurs internes, des employés, des utilisateurs externes, des clients ou des partenaires, alors un produit de gouvernance d'accès puissant, comme Identity & Access Manager, est indispensable pour modéliser facilement votre politique de sécurité grâce à des rôles.

Web Access Manager et Identity & Access Manager travaillent ensemble pour vous offrir un jeu de processus de gouvernance des identités et des droits, prêts à l'emploi et personnalisables, pour répondre aux besoins de votre entreprise.

I&AM déterminera la politique d'accès de Web Access Manager, et il fournira un audit centralisé ainsi qu'un reporting intégré de tous les accès.

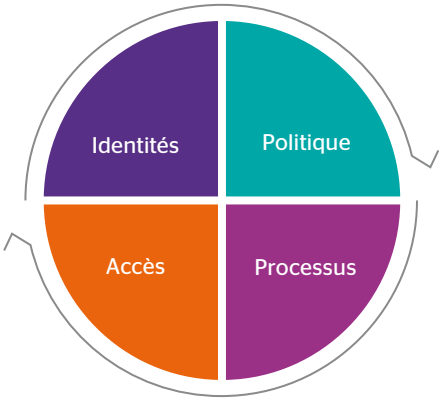
...fournit une gestion complète des accès et des identités :

- A vos applications Web
- A vos applications cloud (SaaS)
- A vos applications mobiles natives utilisant le SDK de Web Access Manager.

Identités Politique Processus Accès I&AM définit et maintient un référentiel central et unique, non-intrusive et compatible avec les sources d'identités existantes.

Le moteur de provisionnement fait appliquer les règles de contrôle des accès sur les systèmes et applications cibles, localement ou dans le Cloud.

Gestion des mots de passe & des méthodes d'authentification



Moteur de politique de sécurité puissant et capable de prendre en compte des besoins complexes et évolutifs.

Administration centrale ou distribuée.

Auto-administration des utilisateurs.

Les processus de gestion des identités et des droits des utilisateurs, des responsables opérationnels et des responsables de la sécurité, assurent la gouvernance de l'accès.

Prérequis au RGPD*

La Gestion des Identités et des Accès est un élément parmi l'ensemble des mesures techniques permettant de mitiger les risques liés à le protection des données. En plus de ses fonctionnalités de contrôle des accès, d'authentification forte et de gouvernance des identités, la Suite Evidian prend en compte les obligations du droit à la personne dans ses solutions. Des fonctionnalités de notification, le libre-service et des rapports dédiées permettent l'exercice des droits utilisateurs et des processus conforme au RGPD.

* Règlement Général sur la Protection des Données

À propos d'Evidian

Evidian est la suite logicielle de gestion des identités et des accès (IAM), d'Eviden.

Evidian IAM est le leader européen des logiciels de gestion des identités et des accès, avec une présence en pleine croissance en dehors du continent européen et notamment aux Etats-Unis et au Japon.

Plus de 5.000.000 d'utilisateurs dans plus de 900 organisations dans le monde entier se connectent tous les jours à leur entreprise et gèrent leurs droits d'accès avec les solutions de gestion des identités et des accès d'Evidian.

CT_J1326_180626_RY_WP_AUTHENTIFIC