“ Le graal ”: WebAuthn, l’authentification web renforcée pour tous

Colin Bontemps, Jeremie Boutard

« Le Graal » : WebAuthn, l'authentification web renforcée pour tous

Colin Bontemps
Direction des systèmes d'information du CNRS
Tour Gaïa, 358 rue Pierre-Gilles de Gennes
31 670 Labège

Colin Bontemps est un spécialiste de la gestion des identités, des accès numériques et de l'authentification web. Il travaille dans l'équipe « Identité et Accès » de la DSI du CNRS depuis 2018.

Jérémie Boutard
Direction des systèmes d'information du CNRS
Tour Gaïa, 358 rue Pierre-Gilles de Gennes
31 670 Labège

Jérémie Boutard travaille au CNRS depuis 2002, où il a débuté par des activités d'ingénieur en infrastructure et système Linux. Aujourd'hui, il est en charge des sujets de sécurité des systèmes d'information de la DSI du CNRS incluant les problématiques liées à la confiance numérique.

Résumé

Nos utilisateurs sont régulièrement confrontés au supplice du mot de passe : « je ne m'en souviens plus », « il est trop complexe », « je me le suis fait voler ». Mais de nouvelles technologies vont y mettre fin !

Depuis 10 ans, le consortium FIDO Alliance produit des standards libres et ouverts pour réduire la dépendance aux mots de passe.

Cela a abouti aux spécifications d'authentification FIDO2 dont la composante web « WebAuthn » est standardisée par le W3C depuis mars 2019. Celles-ci proposent une alternative ergonomique et sécurisée aux mots de passe et aux multiples facteurs d'authentification existant sur le web (mots de passe, certificats x509, OTP…).

Dans cet article et cette présentation, nous exposerons ce standard et ses avantages par rapport aux autres facteurs les plus courant.

Nous expliquerons ensuite pourquoi le CNRS l'a choisi pour sécuriser les accès sensibles de son système d'information de gestion et de service (SIGS) par des mécanismes à plusieurs facteurs.

Pour continuer, nous présenterons et nous détaillerons un retour d'expérience sur la mise en place d'un second fournisseur d'identité dédié à l'authentification forte et basé sur Keycloak. Aussi, nous dévoilerons quelques développements « maison » déployés pour la gestion de clés de sécurité FIDO2. Nous évoquerons les mécanismes de support, d'enrôlement et de secours.

Enfin, nous partagerons nos perspectives pour proposer aux utilisateurs une alternative aux mots de passe pour les accès les moins sensibles du SIGS du CNRS grâce à WebAuthn.

Mots-clefs

Authentification, U2F, WebAuthn, FIDO, Keycloak, SSO, Web SSO, Yubikey, MFA, SAML, passwordless, securité, Shibboleth

1 WebAuthn, authentification renforcée pour tous

1.1 Mot de passe

Pour ouvrir le sujet de l'authentification renforcée, il semble nécessaire de revenir rapidement sur le pilier de l'authentification avant même que l'informatique existe : le mot de passe. Ce dernier sert depuis la nuit des temps à préserver la confidentialité en ne mettant dans la confidence qu'une poignée de personnes choisies. Il est également utilisé dans divers contextes : militaire, cours d'école, contes et légendes (« Sésame, Ouvre-toi ! »), etc.

Employé dès le début des années 60 en informatique pour restreindre l'accès aux rares ressources disponibles, il a été associé à l'identité de l'utilisateur comme premier facteur d'authentification : « ce que je connais ». Mais dès 1962, le mot de passe a été piraté par un jeune chercheur en informatique : Allan L. Scheer[1], qui a pu retrouver les mots de passe stockés en clair dans le serveur.

Depuis, il est présent dans tous nos systèmes d'information sous différentes formes : du code PIN de carte bleu à la passphrase wifi en passant par le mot de passe de session, si bien qu'une journée mondiale[2] lui est dédiée.

Pour les institutions, c'est devenu une source intarissable de normes de stockage, de hachage et de salage[1] [2] comme de recommandations de construction et de renouvellement en fonction du niveau de sensibilité des informations à protéger[3] que demandent d'appliquer nos Politiques de Sécurité des Systèmes d'Information (PSSI)[4].

Pour l'utilisateur, c'est devenu un partenaire quotidien aussi périlleux qu'envahissant dans sa vie professionnelle comme dans sa vie personnelle. Notre utilisateur est soumis à des menaces intangibles de vol qu'il doit associer à la menace bien réelle de l'oubli.

Pour nous, administrateur de système d'information, c'est devenu le sujet de cristallisation des complaintes, un gouffre de temps en support technique et pédagogique associé à une veille technique permanente.

Pourtant, malgré les efforts conjoints des institutions, des utilisateurs et des administrateurs, le mot de passe est resté sensible à la menace du vol par un tiers, quelle qu'en soit la forme : post-it malheureux, lecture indiscrète à l'insu de l'utilisateur, captation des touches du clavier, stockage en clair, partage inapproprié, coercition, brut force[3], password spraying[4], social engineering[5], phishing[6], etc.

Dès lors, il est considéré comme trop faible pour garantir la confidentialité de nos actifs les plus sensibles et le concept de l'authentification basée sur le principe simple du « ce que je connais » n'est plus suffisant face à l'ingéniosité des hackers qu'ils soient institutionnels, professionnels ou amateurs.

---

1 https://history.computer.org/pioneers/scherr.html

2 https://www.journee-mondiale.com/libre/mot+de+passe.htm

3 L'attaque par « brut force » (force brute) est une méthode utilisée en cryptanalyse pour trouver un mot de passe ou une clé. Il s'agit de tester, une à une, toutes les combinaisons possibles.

4 Le « password spraying » est une attaque où un pirate utilise des mots de passe courants pour tenter d'accéder à plusieurs comptes d'un seul domaine. À l'aide d'une liste de mots de passe courants, tels que 123456, azerty et autres, un pirate peut potentiellement accéder à des centaines de comptes en une seule attaque.

5 L'attaque par « social engineering » est une technique qui a pour but d'extirper des informations à des personnes sans qu'elles ne s'en rendent compte. Contrairement aux autres attaques, elle ne nécessite pas de logiciel. La seule force de persuasion est le fondement de cette attaque

6 Le « phishing » (hameçonnage) est une technique frauduleuse destinée à leurrer l'internaute pour l'inciter à communiquer des données personnelles (comptes d'accès, mots de passe…) et/ou bancaires en se faisant passer pour un tiers de confiance.

1.2 L'authentification renforcée

Commençons par un petit rappel sur l'authentification. L'authentification, qui n'est pas à confondre avec l'identification, sert à prouver qui nous sommes. L'identification n'est là que pour déclarer qui nous sommes. Traditionnellement, l'authentification se fait selon différents principes qui sont :

- « ce que je connais » (facteur connaissance – le mot de passe),
- « ce que je possède » (facteur matériel – certificat, carte à puce, clé, etc.),
- « ce que je suis » (facteur biométrique – même si de plus en plus ce facteur est relégué à l'identification – empreinte palmaire, iris de l'œil, etc.),
- et « ce que je fais » (facteur comportemental – geste, voix, etc.)

L'authentification est dite simple quand l'identification est prouvée part un seul facteur, l'authentification est dite forte quand l'identification est prouvée par au moins 2 facteurs (authentification double facteur : 2FA – authentification à plusieurs facteurs : MFA).

Il peut également venir se rajouter un facteur en complément des précédents, le facteur « où je suis ». Ce dernier est un élément adaptatif contextuel à l'authentification. Il peut s'agir d'un lieu physique, d'un lieu réseau, d'un terminal validé, d'un moment de la journée, d'une fréquence d'identification, d'un usage, etc.

Ce dernier facteur permet d'amener la notion d'authentification renforcée, imposé notamment dans le secteur bancaire par la DPS2[5] entrée en vigueur le 13 janvier 2018, qui permet à l'utilisateur, avec la même identité, de s'authentifier différemment selon des facteurs contextuels.

1.3 Besoins fonctionnels

La DSI du CNRS héberge et traite de nombreuses données sensibles en particulier dans son SI de soutien. Cette sensibilité est à la fois intrinsèque (données de santé, données du potentiel scientifique et technique, données sensibles au sens RGPD, etc.), mais aussi liée à la quantité de données traitées.

D'un point de vue menace, les études de risque menées sur les systèmes d'informations gérés par la DSI mettaient en avant une évolution significative de la menace cyber qui allait peser sur les vulnérabilités des actifs de support.

À cette date, le constat était que la DSI du CNRS offrait les possibilités suivantes :

- l'authentification « legacy » par identifiant et mot de passe : l'application gère par elle-même ses utilisateurs et son système d'authentification,
- l'authentification centralisée par WebSSO[7] (IDP SAML Shibboleth/Janus) qui propose l'authentification par identifiant et mot de passe ou par certificat logiciel x509.
- l'authentification fédérée par WebSSO par la fédération d'authentification RENATER qui propose de déléguer l'authentification à plusieurs fournisseurs d'identité SAML[6].

In fine, toutes ces méthodes d'authentification sont basées sur la sécurité du mot de passe de l'utilisateur qui s'avère être beaucoup trop faible pour garantir la sécurité aux vues de la menace existante et des besoins de sécurité. Nous avons considéré que le mot de passe pouvait être considéré de niveau AAL 1[8] (Authenticator Assurance Level) selon la catégorisation du NIST.

7 L'authentification unique web, ou « Web Single Sign-On » (WebSSO) permet aux utilisateurs de ne procéder qu'à une seule authentification pour accéder à un ensemble de services web demandant une authentification.

Il a donc été décidé d'étudier la réalisation d'une solution pour augmenter le niveau de sécurité de l'authentification pour réduire les risques engendrés par la faiblesse des mots de passe. L'objectif recherché pour réduire le risque se situant sur un AAL 3[9] tout en restant acceptable pour l'établissement comme pour les utilisateurs.

1.4 Lignes directrices de la solution

À partir du moment où la décision de mettre en place une solution d'authentification forte au niveau de l'établissement a été prise, nous devions ensuite définir des lignes directrices qui permettraient l'acceptabilité des utilisateurs, l'approbation de l'établissement et l'intégration technique de la solution. Dans ce sens, nous avons établi les critères d'orientation suivants :

Domaines	Sous-domaines	Sujets
Gouvernance	Principe général	La solution doit permettre d'atteindre une AAL 3 pour permettre de protéger les accès des applications web sensibles.
	Solution utilisateur	La solution privilégie une authentification ne nécessitant pas l'utilisation des mécanismes courants du smartphone (OTP, push authenticator, etc.). La solution doit permettre d'être mobile et simple.
	Intégration	La solution doit pouvoir cohabiter avec l'actuelle solution d'authentification et ne pas multiplier les dispositifs d'authentification. Les niveaux d'AAL 2 et AAL 3 pourront utiliser le même dispositif (voire à terme AAL 1 également).
	Cible applicative	Les applications cibles les plus sensibles du SI sont répertoriées, mais elles ne sont pas limitatives. Le dispositif doit être suffisamment modulaire pour accepter, à coûts maîtrisés, d'autres applications.
	Déploiement et support	La solution technique ne doit pas engendrer de coûts de déploiement et de chaîne logistique pour les équipes centrales ou décentralisées du CNRS. Pour gagner en efficacité opérationnelle, les unités doivent être le plus autonome possible. Le corollaire est que le coût pour les unités de la solution doit être faible et maîtrisé.
	Populations cibles	Dans un premier temps, il n'est pas envisagé de déployer la solution à l'ensemble des utilisateurs. Mais la solution doit être suffisamment modulaire et souple pour permettre une généralisation vers l'ensemble des utilisateurs du CNRS avec une maîtrise des coûts d'infrastructures et des coûts de licence.

8 AAL 1 fournit une certaine assurance que le demandeur contrôle un authentificateur lié au compte de l'utilisateur. AAL 1 nécessite une authentification à un ou plusieurs facteurs à l'aide d'un large éventail de technologies d'authentification disponibles. Une authentification réussie nécessite que le demandeur prouve la possession et le contrôle de l'authentificateur via un protocole d'authentification sécurisé

9 AAL 3 offre une très grande confiance que le demandeur contrôle le ou les authentificateur(s) lié(s) au compte de l'utilisateur. L'authentification AAL 3 est basée sur la preuve de possession d'une clé via un protocole cryptographique. L'authentification AAL 3 doit utiliser un authentificateur matériel et un authentificateur offrant une résistance à l'usurpation d'identité du vérificateur ; le même appareil peut remplir ces deux conditions. Afin de s'authentifier en AAL 3, les demandeurs doivent prouver la possession et le contrôle de deux facteurs d'authentification distincts par le biais de protocoles d'authentification sécurisés. Des techniques cryptographiques approuvées sont requises.

Domaines	Sous-domaines	Sujets
Architecture / Besoins techniques	Modèle de solution	Le modèle général est une solution transverse gérée par la DSI du CNRS.
	Modèle de déploiement	Le mode de déploiement « sur site » (« on premise ») est privilégié.
	Modèle de licensing de la solution	Une solution OpenSource est privilégiée. Si une solution payante est déterminée, le mode de licensing doit être étudié pour être maîtrisé.
	Nombre de composants	La solution doit être relativement simple et monolithique, ne pas impliquer le déploiement de composants transverses d'entreprise (type ETL, moteur de workflows, etc.)
	Contraintes pour les composants centraux	OS : Linux RedHat Base de données : PostgreSQL La solution doit pouvoir fournir un haut niveau de SLA[10]. La solution doit être scalable[11]
	Intégration au système d'IAM du CNRS	La solution doit se baser sur les identités gérées par le système IAM du CNRS. Autant que possible, les processus fonctionnels devront s'appuyer sur https://sesame.cnrs.fr
	Intégration aux outils de l'utilisateur	La solution doit pouvoir être utilisée sur l'équipement de l'utilisateur. La cible du CNRS est vaste : OS : Linux, MacOS, Windows Navigateur web : Firefox, Edge, Chrome, Safari
	Drivers pour les applications	Les applications ne doivent pas être écrites ou réécrites spécifiquement pour la solution. La solution doit être compatible avec le protocole SAML 2.0 utilisé actuellement.
Périmètre fonctionnel	Gestion du token d'authentification	La solution devra permettre la gestion des tokens d'authentification : • self-enrollment • self-deactivation • self-renewal (si nécessaire) • processus d'enrôlement en mode face-à-face • désactivation rapide (par une personne habilitée)

10 Le « service-level agreement » (SLA) ou « accord de niveau de service » est un document qui définit la qualité de service, prestation prescrite entre un fournisseur de service et un client. Autrement dit, il s'agit de clauses basées sur un contrat définissant les objectifs précis attendus et le niveau de service que souhaite obtenir un client de la part du prestataire et fixe les responsabilités.

11 En informatique matérielle et logicielle et en télécommunications, l'extensibilité ou scalabilité désigne la capacité d'un produit à s'adapter à un changement d'ordre de grandeur de la demande (montée en charge), en particulier sa capacité à maintenir ses fonctionnalités et ses performances en cas de forte demande ?

Domaines	Sous-domaines	Sujets
	Autres besoins d'authentification	La solution devra permettre de refuser l'accès à un utilisateur authentifié avec un niveau d'AAL insuffisant par rapport au niveau minimum requis pour une cible applicative Ré-authentification possible selon la cible applicative Il n'est pas demandé à l'utilisateur de se ré-authentifier tant qu'il navigue sur des applications référencées comme demandant un niveau d'AAL supérieur à celui actuellement atteint
	Blocage des comptes utilisateurs	Les comptes utilisateurs doivent pouvoir être bloqués sur authentifications en échecs.
	Fonction de secours	La solution doit permettre à un utilisateur ne disposant plus d'un token d'authentification d'accéder de manière temporaire à des applications nécessitant un niveau d'AAL supérieur à 1. La solution doit permettre de : · limiter les cibles pouvant être atteinte en mode secours · limiter la durée d'accès dans ce mode ; · permettre l'activation du mode à la suite d'un workflow de validation de l'identité de l'utilisateur
	Traçabilité	La solution doit permettre de tracer les évènements de gestion et d'accès des utilisateurs et administrateurs de la solution

1.5 Réponse technique

Des éléments d'orientation formalisés, il en est ressorti qu'il y avait plusieurs sujets dans ce projet. Le premier est lié à l'expérience utilisateur et les impacts de la solution dans les unités. Le deuxième est lié à la mise en place technique au niveau des infrastructures centralisées de la DSI. Le troisième est lié au support de la solution aussi bien au niveau des utilisateurs que des applications et métiers qui utiliseront la solution.

L'expérience utilisateur semblant être déterminante ce projet, nous nous sommes concentrés dans un premier temps sur un premier axe technique basé sur l'usage quotidien que pourraient avoir les utilisateurs et les unités d'une solution d'authentification forte.

Nous avons passé en revue un certain nombre de technologie de token matériel ou logiciel pour obtenir un AAL 3.

Par nos orientations, nous avions déjà exclu toutes les solutions, qui auraient pourtant été simples à mettre en place, basées sur le smartphone de l'utilisateur avec un composant OOB[12], aussi bien

12 Matériel (ou soft) adressable par le vérificateur au travers d'un canal spécifique différent du canal primaire

celles qui se basaient sur un SMS OTP – réputées aujourd'hui non-fiable – que celles qui se basent sur un Authenticator – type OpenOTP[13] -.

Nous avons exclu assez rapidement les solutions d'authentification propriétaire comme les tokens RSA SecureID : elles sont relativement chères à l'achat et au support, la chaîne d'approvisionnement et d'assistance peut être lourde à mettre en œuvre et à gérer lors d'un passage à l'échelle. Ces dernières années un certain nombre d'incident de sécurité[7] laisse à penser que ces solutions ne sont pas forcément les plus fiables.

Nous avons finalement exclu les solutions basées sur les certificats X509 sur clé matériel qui nous auraient permis d'atteindre potentiellement une AAL 4 et un niveau RGS ***[8]. Mais la solution aurait été surqualifiée par rapport à nos besoins et le déploiement des certificats X509 sur clé demande une logistique que nous ne pouvions pas mettre en place.

Enfin, nous nous sommes intéressés au standard U2F de FIDO que le NIST venait de qualifier au niveau AAL 3[9]. Ce standard permet de répondre à la plupart de nos critères :

- le standard est ouvert[14],
- il est orienté web,
- il est intégré aux OS et navigateurs cibles,
- il est ergonomique pour l'utilisateur,
- il est aussi bien adapté aux usages personnels qu'aux usages professionnels,
- l'utilisateur et l'unité peuvent gérer eux-mêmes l'approvisionnement (la plupart des sites web d'e-commerce propose des clés U2F),
- en retour d'expériences, les GAFAM (Google[15], Amazon[16], Microsoft[17], etc.) l'avaient déjà adopté et, plus récemment, les banques[18] permettent aussi l'usage de ce standard,
- nous avons pu assez rapidement réaliser un POC avec Shibboleth[19],
- son principal désavantage est qu'il n'intègre pas de façon native de protocole de secours,
- un autre désavantage est que, étant un protocole ouvert, il existe aujourd'hui une multitude de fabricants de clé U2F. Toutes n'apportent pas le même niveau de sécurité et de respect du standard. La FIDO Alliance a donc dû mettre en place une certification[20].

1.6 De U2F à WebAuthn

1.6.1 À l'origine

Le Consortium FIDO Alliance, actif depuis 2013, réuni des acteurs du secteur privé pour développer une solution d'authentification sécurisée facile à utiliser (authentification en un seul geste), respectant la vie privée et facile à mettre à l'échelle. Leurs travaux ont produit plusieurs

13 https://play.google.com/store/apps/details?id=com.rcdevs.auth&hl=fr&gl=US

14 https://github.com/google/OpenSK

15 https://store.google.com/fr/product/titan_security_key?hl=fr,,

16 https://docs.aws.amazon.com/fr_fr/IAM/latest/UserGuide/id_credentials_mfa_enable_u2f.html

17 https://docs.microsoft.com/fr-fr/azure/active-directory/authentication/howto-authentication-passwordless-security-key

18 https://www.boursorama.com/aide-en-ligne/mon-espace-client/identifiant-et-mot-de-passe/question/en-quoi-consiste-la-connexion-par-cle-de-securite-sur-internet-5165516

19 https://www.yubico.com/fr/works-with-yubikey/catalog/shibboleth/

20 https://fidoalliance.org/certification/fido-certified-products/

protocoles et normes basés sur l'authentification par cryptographie asymétrique. Parmi ceux-ci nous pouvons trouver UAF, U2F puis WebAuthn[21].

Le principe est d'avoir une paire de clés asymétriques unique par utilisateur et par site web auquel il accède. Cette paire de clé est enrôlée directement sur le site web cible appelé « Relying party »[22]. Ce dernier stocke la clé publique associée à chaque utilisateur pour le reconnaître. Ce mécanisme garanti une protection de fait contre l'hameçonnage, le protocole s'appuyant sur TLS pour authentifier le serveur avant d'enclencher l'authentification et la clé privée ne passant jamais sur le réseau.

La clé privée est soit stockée sur un élément sécurisé externe (ex. : clé de sécurité), soit intégrée à la plateforme (ex. : TPM[23]). L'utilisation de cette clé privée peut être protégée par un facteur d'authentification local, en général biométrie ou code pin. Pour faire une analogie dans le monde des systèmes, cela pourrait s'assimiler à une paire de clé SSH unique pour chaque serveur, toutes protégées par le même code pin. Le protocole permet aussi de dériver une paire de clés par site à l'aide d'une unique clé privée présente sur le matériel. Cela permet de fabriquer du matériel d'authentification moins cher, car ne nécessitant pas de stockage. Cela permet aussi l'utilisation virtuellement illimitée d'un unique authentificateur FIDO pour de nombreux services.

1.6.2 Le protocole U2F

C'est le protocole U2F qui nous intéresse de prime abord à la DSI du CNRS. Il vient se positionner en second facteur d'une authentification initiale par mot de passe. Le principe est une clé de sécurité externe présentée via USB, NFC ou Bluetooth LTE. Le protocole U2F est issu en particulier des travaux sur l'authentification des employées de Google en partenariat avec le fabricant de clé de sécurité suédois Yubico[10]. Google publie cette étude en 2016 ou ils comparent l'usage en 2^{e} facteur d'une clé de sécurité avec celle d'un code « one time password » (OTP) envoyé par SMS ou d'un OTP généré sur une application mobile. Ces seconds facteurs viennent en complément du mot de passe. Cette étude montre que la clé de sécurité est plus rapide à utiliser et génère moins de support que les méthodes via OTP.

1.6.3 Arrivée de FIDO2 et WebAuthn

Les travaux initiaux de FIDO sur la biométrie (UAF) et sur le second facteur (U2F) ont depuis été intégrés au sein de la norme FIDO2. FIDO2 comprends deux normes :un volet navigateur WebAuthn et un volet « Client to authenticator » CTAP2[24]. Webauthn est une API JavaScript intégrée au navigateur et standardisée par le W3C en 2019. WebAuthn permet d'initier l'enrôlement et l'authentification des utilisateurs depuis le code JavaScript de l'application cliente.

WebAuthn peut être utilisé pour proposer un second facteur en complément du mot de passe, mais aussi pour remplacer complètement le mot de passe. Dans le cas d'un second facteur, il est possible de se contenter de la possession de l'authentificateur comme preuve unique. Dans le cas d'une authentification sans mot de passe (passwordless), un facteur d'authentification local est en général demandé pour utiliser l'authentificateur. Ce facteur local pourra prendre la forme soit d'une biométrie, soit d'un facteur de connaissance, qui peut dans ce cas être un simple code PIN. L'avantage du facteur d'authentification local est qu'il est par nature moins vulnérable à l'hameçonnage (car il ne passe pas sur le réseau) et permet aussi de conserver les données biométriques en local sur le matériel de l'utilisateur.

21 Liste complète disponible sur https://fidoalliance.org/specifications/

22 Relying Party : au sens FIDO, il s'agit de l'application utilisant les mécanismes d'authentification FIDO. Dans un contexte d'authentification unique, le Relying Party FIDO est souvent le fournisseur d'identité.

23 TPM : Trusted Platform Module, technologie de coffre fort cryptographique intégrée à un appareil.

24 CTAP2 : Client To Authenticator Protocol 2.

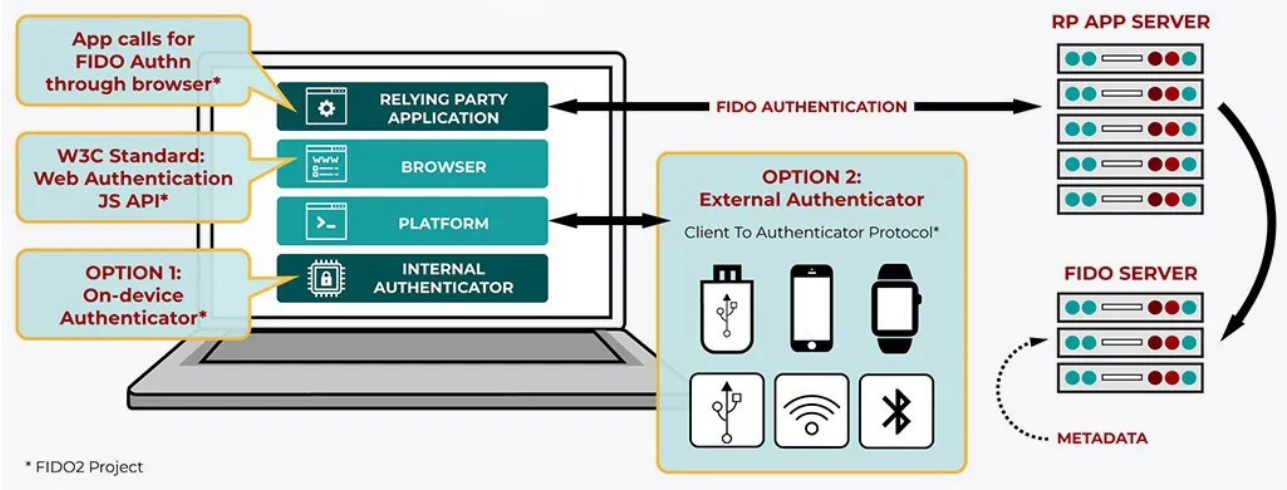

Figure 1: Schéma global de fonctionnement de FIDO2.
Source : https://fidoalliance.org/fido2/

1.6.4 Matériel U2F et FIDO2,

Les authentificateurs externes uniquement certifiés U2F sont compatibles avec WebAuthn, mais ne permettent pas d'utiliser le mode sans mot de passe. Pour cela, il est impératif d'employer un authentificateur certifié FIDO2. Les clés de sécurité FIDO U2F coûtent autour de 10 à 20 € et celles FIDO2 autour de 20 à 30 € par unité. Les clés de sécurité U2F sont limitées à un usage en second facteur. Les clés de sécurité FIDO2 peuvent être utilisées en facteur principal pour une authentification « passwordless » voire « loginless ».

Les authentificateurs FIDO2 se regroupent en deux familles : les authentificateurs « platform » c'est-à-dire intégré à l'appareil, et les authentificateurs externes.

Les **authentificateurs intégrés à l'appareil** utilisent en général la puce « TPM » de l'appareil pour stocker la clé privée. Le déverrouillage utilise une biométrie ou un code pin géré par le système d'exploitation. Sous MacOS Safari permet depuis sa version 14[25] d'utiliser « Touch ID » ou « Face ID »[26] comme un authentificateurs FIDO2 de type « platform ». Sous Windows 10 et 11 il est possible d'utiliser « Windows Hello »[27] avec ses options code pin ou biométriques comme un authentificateur FIDO2 intégré[28]. Côté mobile, WebAuthn est également supporté par certains navigateurs sous Android et iOS. Dans les 2 cas, les mécanismes natifs de déverrouillage du téléphone sont utilisés comme facteur local d'authentification. Il n'y a pas encore d'authentificateur intégré de manière générique sous Linux à la connaissance des auteurs.

Les **authentificateurs externes** peuvent prendre de nombreuses formes. Comme prévu dans la norme CTAP, ils peuvent fonctionner en USB, NFC ou BLE. L'exemple typique sont les clés USB de sécurité du type « Yubico security key »[29]. Il existe de nombreux fournisseurs de clés proposant différents facteurs de forme, mais aussi différents niveaux de sécurité et de certification. Il est aussi possible de réaliser des authentificateurs externes logiciels, ou d'utiliser son téléphone Android en tant qu'authentificateur FIDO[30] externe sur un ordinateur via Bluetooth. Il existe des modèles de

25 Release notes safari 14 :https://developer.apple.com/documentation/safari-release-notes/safari-14-release-notes

26 « Touch ID » et « Face ID » sont des systèmes d'authentification biométriques propriétaire d'Apple.

27 « Windows Hello » est un système d'authentification, pouvant utiliser la biométrie, propriétaire de Microsoft

28 https://docs.microsoft.com/en-us/windows/security/identity-protection/hello-for-business/webauthnapis

29 Exemple de clé de sécurité USB: https://www.yubico.com/fr/product/security-key-nfc-by-yubico/

30 Comment utiliser l'authentificateur FIDO interne d'un téléphone Android : https://support.google.com/accounts/answer/9289445

clés open source et open hardware[31] qui permettent d'en savoir plus sur les entrailles de ces appareils.

Tout appareil compatible FIDO2 doit permettre une vérification de l'utilisateur. La vérification d'utilisateur demandée par FIDO2 pour l'authentification « passwordless » peut être un simple code PIN sur les modèles de base. Sur certains modèles de clés de sécurité haut de gamme, un lecteur d'empreinte digitale est intégré[32]. Une clé de sécurité logicielle sur téléphone Android pourra faire usage du mécanisme de déverrouillage natif (code, schéma, biométrie… selon le modèle) pour vérifier l'utilisateur.

Il existe plusieurs types de certifications pour les authentificateurs FIDO. La principale est la certification issue de la FIDO Alliance elle-même[33]. Elle définie plusieurs niveaux L1, L1+,L2, L3 et L3+. Le niveau L3+ impose en particulier le stockage sur un élément sécurité certifié critères communs[34]. En dehors de la certification de la FIDO Alliance, certains constructeurs font certifier leurs matériels FIPS ou CSPN (cf. : ANSSI[35]).

1.6.5 Compatibilité navigateur U2F ou FIDO2

La mise en place des nouveaux protocoles WebAuthn et CTAP2 a demandé des évolutions dans les navigateurs mais aussi dans les systèmes d'exploitation. L'écosystème évolue encore et le support continue à s'étendre. Les authentificateurs externes via USB et NFC sont maintenant largement supportés et peuvent être utilisés assez largement. Apple qui était un peu à la traîne a fini par rejoindre la FIDO Alliance en février 2020[36]. Microsoft qui était un peu en retard sur U2F a pu rattraper tout son retard en tirant parti de Chromium dans son dernier navigateur Edge. Du côté des authentificateurs intégrés, il manque encore quelques acteurs clés de l'écosystème, tel que le support de Linux et de Firefox sous MacOS[37].

31 Exemple de clés open source, les SoloKeys : https://github.com/solokeys/solo

32 Exemple de clé FIDO biométrique chez Yubico : https://www.yubico.com/fr/product/yubikey-bio/

33 https://fidoalliance.org/certification/authenticator-certification-levels/

34 https://fidoalliance.org/certification/authenticator-certification-levels/authenticator-level-3-plus/

35 https://www.ssi.gouv.fr/entreprise/certification_cspn/yubikey-5-series-version-firmware-5-4-2-2/

36 https://fidoalliance.org/forbes-apple-just-made-a-striking-new-security-move-that-could-impact-all-users/

37 https://webauthn.me/browser-support

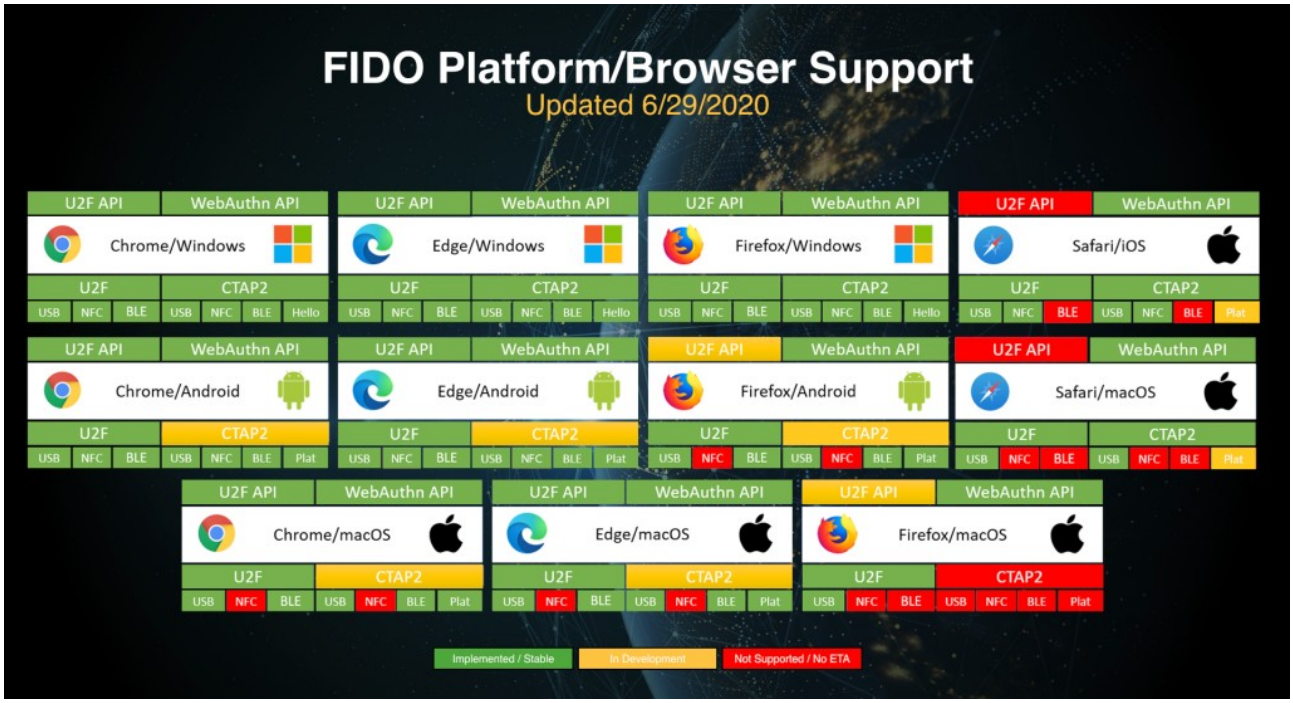

Figure 2: Support des différents protocoles FIDO au 29/6/2020 selon FIDO Alliance
Source : https://fidoalliance.org/fido2/fido2-web-authentication-webauthn/

Les spécifications WebAuthn et CTAP2 continuent d'évoluer et nous pouvons nous attendre à ce que le niveau de support des nouvelles fonctionnalités WebAuthn dans les navigateurs ne soit pas homogène. Chrome et Microsoft restent un cran en avance du fait de leur implication de long terme dans la FIDO Alliance, mais la nature ouverte de ces normes et son implémentation open source dans Chromium laisse l'opportunité aux projets du monde du libre de suivre le train (voire de les devancer).

1.7 FIDO2

1.7.1 Fonctionnement détaillé de FIDO2

La cinématique d'enrôlement permet au site web de recevoir et d'enregistrer la clé publique correspondant à l'authentificateur de l'utilisateur. Chaque enrôlement ou authentification via WebAuthn demande une action de l'utilisateur pour s'assurer de son consentement. L'action peut être un bouton dans l'interface, toucher la clé, insérer la clé, utiliser un lecteur d'empreinte digitale, etc. Cela permet d'éviter que l'authentificateur ne soit utilisé à l'insu de l'utilisateur.

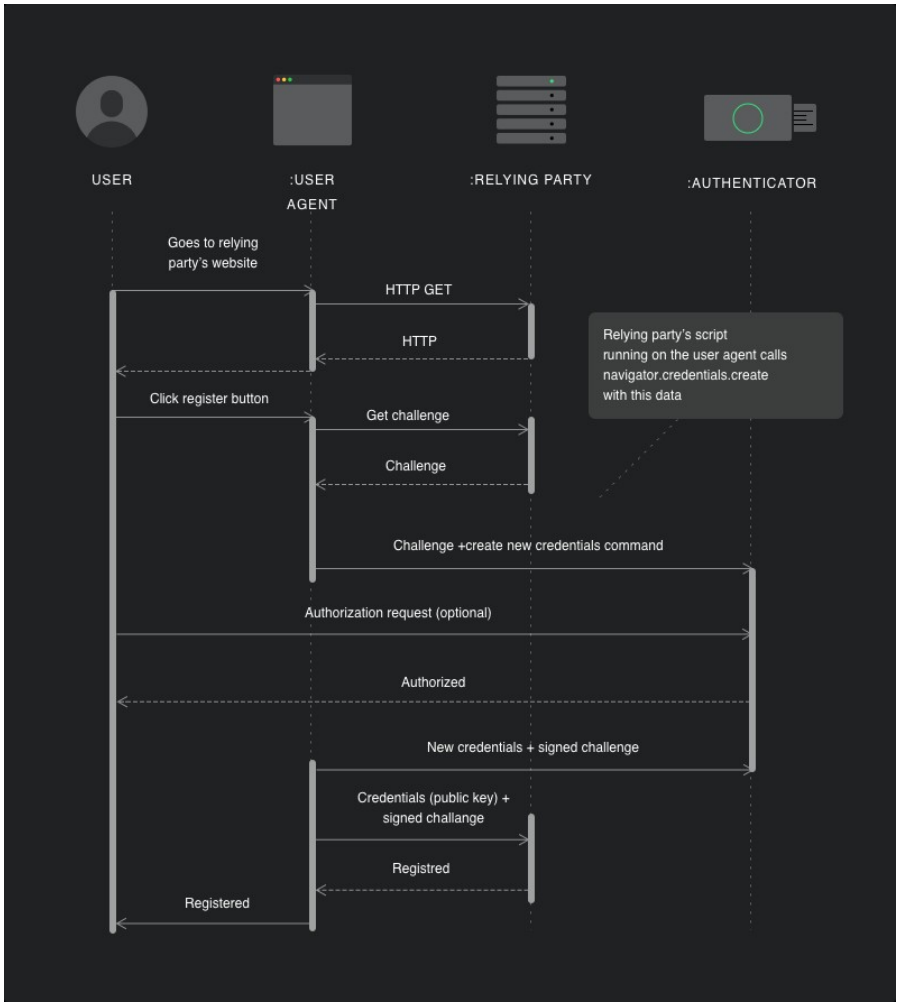

Figure 3: Enrôlement avec FIDO2.
Source : https://webauthn.me/introduction

Les cinématiques d'enrôlement et d'authentification via WebAuthn sont décrites dans les figures 3 et 4. Les différents acteurs de ces schémas sont :

- « user » : l'utilisateur de l'application web
- « user agent » : le navigateur ainsi que le code javascript de l'application exécutée sur le navigateur
- « relying party » : la partie serveur ou « backend » de l'application web

- « authenticator » : l’ authentificateur FIDO2 qui peut être externe ou interne.

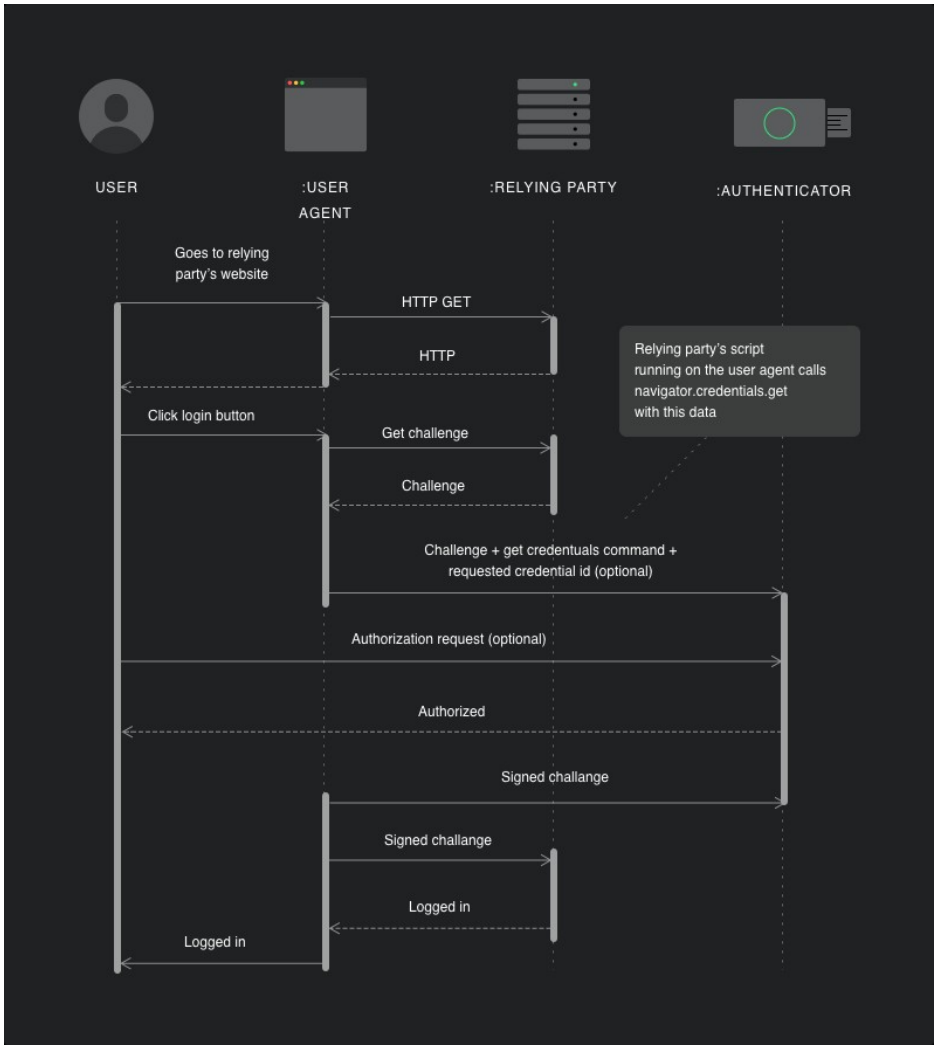

Figure 4: Authentification avec FIDO2.
Source : https://webauthn.me/introduction

1.7.2 FIDO2 en dehors du web

La partie CTAP2 de FIDO2 n’a rien de spécifique au web. Rien n’empêche d’utiliser les authentificateurs FIDO2 ou FIDO U2F dans d’autres contextes. C’est en particulier le cas avec OpenSSH qui permet depuis sa version 8.2 d’utiliser une clé SSH générée à partir d’un authentificateur FIDO2. L’option de clé résidente permet de faciliter la portabilité de la clé SSH en stockant la « key handle » sur l’authentificateur.

Pour Générer une clé SSH sur authentificateur FIDO2 :

```
ssh-keygen -t ecdsa-sk -f ~/.ssh/id_mykey_sk
```

Avec l’option « resident » pour faire une clé résidente.

```
ssh-keygen -t ecdsa-sk -f ~/.ssh/id_mykey_sk -O resident
```

1.7.3 FIDO2 et l'authentification unique dans un organisme

Le besoin d'authentification unique (SSO) pourrait être ré-évalué dès lors qu'il n'y a plus de mot de passe à saisir et que l'authentification peut être réalisée d'un simple geste. En effet si l'authentification est facile et ergonomique pour l'utilisateur, on peut envisager de lui demander de s'authentifier plus régulièrement. Pour autant un service d'authentification unique permet aussi de centraliser la gestion de l'enrôlement et des modes de secours, et d'éviter de devoir gérer ces mécanismes dans chaque application. FIDO2 s'intègre donc naturellement dans les mécanismes d'authentification unique existant pour les organismes publics ou privés. Il peut être proposé en facteur alternatif au mot de passe pour du « passwordless », ou en facteur complémentaire au mot passe, selon les besoins identifiés. À noter, les protocoles FIDO ne sont pas conçus spécifiquement pour implémenter une forme de SSO[38].

Dans le cas d'une authentification unique, le service d'authentification agit alors en tant que « relying party » au sens FIDO. Ensuite il peut agir en tant que fournisseur d'identité pour les applications raccordées, par exemple par l'intermédiaire d'un protocole de fédération d'identité.

1.7.4 FIDO et les protocoles de fédération d'identité

Les protocoles FIDO se positionnent en complément aux protocoles de fédération d'identité et ne visent pas à les remplacer[39]. Une des synergies identifiée entre FIDO et protocole de fédération d'identité est que ces derniers permettent de véhiculer la nature de l'authentification réalisée. Il serait alors possible pour une application d'exiger une authentification FIDO de la part d'un fournisseur d'identité ou de refuser l'accès si le type d'authentification réalisé ne correspond pas à ses attentes.

Les protocoles de fédération tels que SAML ou OpenID Connect permettent de véhiculer le type d'authentification demandé et de l'authentification réalisée dans les messages de ces protocoles. Le nom des paramètres est :

- dans SAML : saml:AuthnContextClassRef[40],
- dans OpenID Connect :acr_values[41].

Pour référence, un exemple détaillé de transmission du type d'authentification avec SAML est décrit dans le paragraphe 3 d'une publication au JRES2019 de Guillaume Rousse et Ludovic Auxepaules [6].

À ce jour et à la connaissance des auteurs, il n'existe pas de valeur standardisée pour FIDO ou WebAuthn dans les valeurs de ces attributs.

Les valeurs de l'attribut SAML AuthnContextClassRef transmet généralement la nature de l'authentification réalisée[42]. La nature seule du moyen d'authentification utilisé n'est pas suffisante pour déduire le niveau de confiance à accorder à l'identité fournie. Ce niveau de confiance dépend d'autres facteurs dont la mode d'enrôlement du moyen d'authentification ainsi que des politiques de sécurité en place. Pour illustrer ce propos, une authentification FIDO, aussi forte soit-elle, avec un enrôlement en self-service accessible publiquement sur internet ne donne aucune garantie sur l'identité de la personne « authentifiée ».

38 https://fidoalliance.org/integrating-fido-and-federation-protocols/

39 https://fidoalliance.org/fido-and-federation-protocols-tech-note/

40 Défini dans la spécification « Authentication Context for the OASIS Security Assertion Markup Language (SAML) V2.0 ».

41 Défini dans la spécification « Core » de OpenID Connect.

42 https://docs.oasis-open.org/security/saml/v2.0/saml-authn-context-2.0-os.pdf

1.7.5 Différences et synergies entre FIDO et une infrastructure de gestion des clés (IGC)

La principale différence entre FIDO et une IGC est que dans FIDO il n'y a pas d'autorité centrale qui émet et signe les paires de clés asymétriques. Chaque paire de clé FIDO est générée pour un utilisateur et pour un service.

Avec un **certificat personnel x509** émit par une IGC :

- L'enrôlement du certificat est centralisé. Avec FIDO il serait nécessaire d'enrôler l'utilisateur pour chaque service authentifié, mais l'utilisation d'une authentification unique (WebSSO) évite ce ré-enrôlement systématique à chaque service.
- Le service qui authentifie n'a besoin que du certificat de l'AC et de la liste de révocation pour valider l'identité de l'utilisateur. Avec FIDO, le service doit stocker la clé publique de chaque utilisateur enrôlé pour assurer l'authentification.
- La clé privée est générée au sein de l'entreprise en particulier si l'IGC est interne. Avec FIDO, la clé mère du TPM ou de la clé de sécurité externe est générée par un tiers, en général le constructeur. Ce fonctionnement peut entraîner des risques supplémentaires par rapport à une IGC ou tous les secrets sont générés en interne.
- La clé privée est facilement exportable et transportable, ce qui représente à la fois un avantage ergonomique mais aussi un risque de sécurité. Avec FIDO la clé privée ne quitte jamais l'authentificateur.
- Le certificat personnel peut servir à d'autres usages, comme signer des e-mails. FIDO se concentre sur l'authentification.

Avec un **authentificateur FIDO** :

- Moins de complexité de mise en œuvre : Avec FIDO pas besoin d'IGC, de renouvellement d'AC, d'AC intermédiaire, pas de clé privée de l'AC hyper sensible à gérer et pas de liste de révocations à gérer.
- L'enrôlement est décentralisé et se passe techniquement directement entre le service et l'utilisateur. En organisation c'est souvent géré par le service de SSO. L'enrôlement est géré directement dans le navigateur via une API JavaScript standard. Pas besoin de client lourd ou de client web java pour installer un certificat sur une clé USB.
- La protection de la clé privée par un facteur local peut être exigée à l'authentification. C'est ce qui permet une authentification à la fois « passwordless » et multi facteur. Avec un certificat on peut laisser sa clé privée déprotégée et le serveur qui authentifie l'utilisateur n'en saura rien.
- Le sens du vent souffle plus dans la direction de FIDO que des certificats personnels. Les GAFAM investissent largement sur FIDO. Les grands cabinets d'analyse recommandent de s'intéresser à FIDO dans les organisations sans pour autant enterrer les certificats[43].

43 https://www.gartner.com/doc/reprints?id=1-2634S388&ct=210519&st=sb

Une authentification Web FIDO et une authentification Web par certificat ne sont pas forcément à mettre en opposition et peuvent aussi agir en synergie. Les facteurs d'authentifications peuvent être complémentaires, et peuvent aussi agir en moyen de secours l'un pour l'autre.

1.7.6 Evolution de FIDO2

Les standards Web Authentication et CTAP2 continuent d'évoluer, même si le socle de base de l'authentification est maintenant mature. Le tableau ci-dessous détaille certaines des principales nouveautés[44].

Web Authentication Level 2, recommandation W3C publiée en avril 2021[45] :

- Enterprise Attestation (EA) : permet d'atténuer le fait qu'un TPM a souvent une clé unique qui permettrait de pister l'utilisateur utilisant le même authentificateur intégré sur plusieurs sites.

CTAP 2.1, à l'état de « Review Draft »[46] chez FIDO Alliance :

- Gestion des authentifiants présents sur un authentificateurs : API standard pour gérer le contenu d'un authentificateur, en particulier les clés résidentes pour le « login less ».
- Gestion de la biométrie : ajouter / enlever des empreintes.
- Gestion du code PIN : en particulier la longueur minimale et demander un changement.
- Vérification de l'utilisateur systématique : permet à l'utilisateur de configurer son authentificateur pour toujours vérifier son identité à l'usage, même si le site ne le demande pas.

Une version Web Authentication Level 3 est en cours de rédaction, au jour de l'écriture encore à l'état de « Working Draf »[47].

2 Projet

Au-delà du projet technique, la mise en place de l'authentification renforcée nécessite l'implication d'un certain nombre d'acteurs pour être mené à terme. Cela est d'autant plus vrai qu'il s'agit d'un projet « transverse » qui ne sert pas directement un objectif fonctionnel métier de l'établissement.

2.1 Sponsor métier

Le sponsor « métier » est finalement l'acteur qui va cadencer la mise en place et l'acceptation du dispositif. Un retard sur le projet « métier » va se traduire par un retard sur le projet « transverse ».

Dans le cas des dispositifs U2F, le métier a accepté le principe lié à la facilité d'usage et aux mesures organisationnelles de la solution. A cela s'ajoute la possibilité pour les acteurs « métier » d'avoir accès aux nouvelles fonctions de nomadisme sur l'application cible grâce à

44 https://techcommunity.microsoft.com/t5/identity-standards-blog/what-s-new-in-passwordless-standards-2021-edition/ba-p/2124136

45 https://www.w3.org/TR/webauthn-2/

46 https://fidoalliance.org/specs/fido-v2.1-rd-20210309/

47 https://www.w3.org/TR/webauthn-3/

l'authentification renforcée (fonctions qui n'étaient pas disponibles sur le contexte précédent en raison de la faiblesse d'un dispositif de simple mot de passe).

2.2 Équipes techniques

L'équipe techniques ou plutôt les acteurs techniques ont aussi un rôle à jouer dans la réussite du projet. Dans le contexte de la DSI du CNRS, il a s'agit de coordonner en interne les responsables sécurité des systèmes d'information (porteur de la fonction) avec les acteurs responsables des identités et authentification (porteur du produit), les développeurs (constructeurs des outillages et des intégrations entre outils), les exploitants techniques (les administrateurs systèmes et réseaux) et le support (des futurs utilisateurs).

En phase d'exploitation, la coordination de la fonction transverse à déployer se répartira également sur les acteurs du terrain que sont les services des systèmes d'information en délégation et les administrateurs de système d'information locaux.

3 Implémentation

3.1 Support et gestion des clés

Un des axes du dispositif était la facilité pratique et économique pour l'utilisateur de se doter du matériel nécessaire pour la double authentification. Avec la norme FIDO2 retenue et son orientation grand public, il est possible de se procurer ce matériel facilement, à moindre coût sans aucune difficulté liée au canal d'achat.

Par contre, en termes de sécurité de l'authentification et de sûreté de fonctionnement, nous avons commencé par limiter l'accès aux dispositifs aux clés certifiées par la FIDO Alliance (cf. : chapitre 1.5 Réponse technique, page 6). Puis par souci de gestion de support aux utilisateurs et de capacité du serveur d'authentification nous avons été contraints de nous limiter à un nombre limité de token matériel (cf. chapitre 3.7.2 Vérification du matériel WebAuthn utilisé page 20).

3.2 Enrôlement

Pour l'enrôlement, organisationnellement, nous avons opté pour un « self-enrollment » de l'utilisateur sur l'initiative, à terme, d'un chargé de sécurité des systèmes d'information, comme cela se réalisait antérieurement pour les certificats CNRS2-Standard. Techniquement, nous sommes appuyés sur l'outil https://sesame.cnrs.fr avec un développement spécifique.

Pour le processus pratique, l'utilisateur, muni de sa clé FIDO2, contacte par un « face-à-face » en présentiel ou à distance son chargé de sécurité des systèmes d'information. Une fois l'identification « face-à-face » effectuée, ce dernier se connecte à https://sesame.cnrs.fr pour initier la demande d'enrôlement. Cette initialisation se présente comme un mail envoyé à l'utilisateur sur son adresse mail pré-enregistré sur l'annuaire de l'établissement. Ce mail invite l'utilisateur à se connecter à une URL sérialisée à usage limité dans le temps (30 minutes). Quand l'utilisateur va sur ce lien, https://mfa.id.cnrs.fr, le site l'invite à lier sa clé à son profil. À la fin du processus, la clé est enregistrée et l'utilisateur pourra, par la suite, s'authentifier en 2FA aux applications qui le nécessitent.

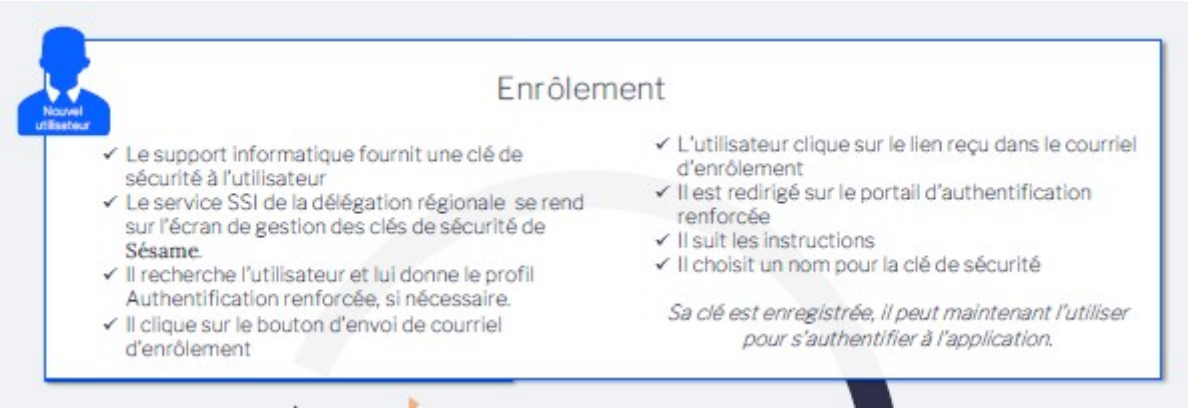

Figure 5 : Extrait de la maquette du processus d'enrôlement

3.3 Mode de secours

Techniquement, le protocole FIDO2 n'implémente pas, par défaut, de mode de secours. Pourtant, ce mécanisme est essentiel pour permettre l'accès au système d'information pour les utilisateurs lorsque que ces derniers ont perdus leur second moyen d'authentification.

La difficulté du mode de secours est que ce dernier ne doit pas affaiblir ou dégrader le niveau d'authentification recherché (AAL 3). Après plusieurs essais et ateliers de procédures techniques ou organisationnelles (mail OPT, SMS, grilles, quorum de confiance, questions secrètes, assistance utilisateurs, etc.), nous avons fini par nous ranger à l'avis des experts de le FIDO Alliance[11]. Le seul moyen de secours qui permet de ne pas dégrader le niveau d'authentification ciblé est finalement basé sur la mise à disposition par l'utilisateur d'un nouveau token d'authentification. La FIDO Alliance préconise soit la possibilité pour l'utilisateur de posséder plusieurs clés d'authentification enregistrées, soit de posséder plusieurs clés dont une enregistrée et d'autres à enregistrer en cas de perte de la première.

Pour se prémunir des risques de vol non-perçu de clé, nous avons opté pour la solution où l'utilisateur n'enregistre qu'une clé d'authentification. En cas de perte de cette clé, il devra enrôler une nouvelle clé (cf. : chapitre 3.2 Enrôlement, page 17) pour pouvoir de nouveau accéder aux applications qui nécessitent un accès AAL 3. Ce nouvel enrôlement supprime l'enregistrement de la clé précédente.

3.4 Révocation

La révocation est l'autre processus essentiel à l'authentification renforcée, nous avons opté pour l'implémentation d'un dispositif de révocation des clés d'authentification (à l'image des processus de révocation des certificats – à la différence qu'il n'y aura pas la nécessité d'une liste de révocation étant donné que le serveur d'authentification ne connaît que les clés enregistrées).

Via le site https://sesame.cnrs.fr, nous avons implémenté la possibilité de révoquer une clé d'authentification, à terme, par l'utilisateur, son CSSI (et adjoints), son RSSI Régional (et adjoints), son RSSI National (et adjoints) ou le support utilisateur de la DSI. Techniquement, la clé est supprimée du référentiel.

Il est à noter qu'en cas de départ de l'utilisateur de l'établissement, automatiquement, son compte n'ouvre plus de droits et qu'il ne pourra plus se connecter aux applications même si la clé est encore enregistrée.

À noter également, qu'une clé supprimée peut de nouveau être enrôlée. En effet, la suppression enlève une référence à une biclé générée sur la clé lors de l'enrôlement. Lors d'un nouvel enrôlement de la même clé physique, une nouvelle biclé logique sera générée sur la clé et enregistrée sur le référentiel (cf. chapitre 1.7 FIDO2 page 12).

3.5 Serveur d'authentification

Plusieurs solutions ont été mises à l'étude pour ajouter de l'authentification forte à la DSI du CNRS. Nous avons comparé en particulier la possibilité d'étendre notre SSO existant basé sur l'IDP Shibboleth avec un mécanisme d'authentification forte ou la possibilité d'un second IDP indépendant dédié à l'authentification forte. La deuxième solution a été retenue, au moins temporairement, comme cela sera expliqué ci-dessous.

Les raisons pour ne pas avoir retenu l'IDP Shibboleth pour ce besoin dans un premier temps sont diverses. Tout d'abord il n'y a pas de support officiel de U2F ou WebAuthn mais uniquement des modules communautaires. Au jour de l'écriture aucun d'entre eux n'est officiellement compatible avec la version 4 de l'IDP Shibboleth[12]. Ensuite, Webauthn demande d'avoir des interfaces d'enrôlement et de gestion des clés, hors Shibboleth ne propose pas d'interface de self service ni de gestion des moyens d'authentification par défaut. Il faut donc mettre en œuvre un portail de gestion en complément. Enfin une des implémentations classique de l'authentification forte par les universités nord américaines est une solution « Software as a Service » de second facteur d'authentification, appelée Duo Security. Cette solution, depuis rachetée par Cisco, propose une mise en œuvre rapide et des interfaces ergonomiques et elle bien intégrée avec Shibboleth IDP. Cependant elle est obligatoirement hébergée chez un grand fournisseur de cloud états-unien, ce qui n'est pas adapté au contexte du CNRS et des EPST en France. Enfin, les délais initiaux attendus du projet étant assez courts, il apparaissait plus pragmatique d'utiliser une solution clé en main.

3.6 Choix techniques

Le choix s'est alors porté sur RedHat SSO (en commençant avec sa version communautaire Keycloak) pour réaliser cette première implémentation WebAuthn. Parmi ses forces, le support complet de WebAuthn quand les autres éditeurs ne supportaient encore que U2F (ils ont depuis rattrapé leur retard pour la plupart), la communauté importante et grandissante y compris dans le monde académique (Ex : CERN[13]), le code ouvert sous licence ASL2[48], le coût maîtrisé pour la version supportée par RedHat, l'interface graphique simple et moderne, etc. Les principales faiblesses que nous avons identifiées dans notre contexte sont le manque d'interopérabilité avec une fédération d'identité (il faut déclarer les fournisseurs de service un par un) et la jeunesse relative des mécanismes d'authentification multi facteur dans le produit et donc en particulier de WebAuthn (la fonctionnalité est encore officiellement en mode « technology preview »[49] dans RedHat SSO).

Ce nouvel IDP d'authentification forte « Janus+ » propose de l'authentification forte systématique pour les applications les plus sensibles du système d'information. La mise en œuvre de l'authentification forte de manière plus globale sur le SSO principal du CNRS (Janus) est encore au stade de projet. Plusieurs scenarii sont à l'étude, dont certains avec une coordination entre les deux fournisseurs d'identité (dans le style d'une intégration CAS & Shibboleth) ou en ne gardant qu'un seul d'entre eux.

48 https://github.com/keycloak/keycloak/blob/main/LICENSE.txt

49 La fonctionnalité WebAuthn n'est donc pas encore pleinement supportée par RedHat.

3.7 Détails techniques

3.7.1 Configuration WebAuthn pour notre usage

La configuration WebAuthn de Keycloak est très complète et propose la majorité des paramétrages disponibles dans la spécification[50]. Le choix est fait de proposer à la fois une configuration WebAuthn orientée second facteur, et une configuration WebAuthn orientée « sans mot de passe » qui peuvent coexister avec des paramétrages différents.

Dans le cas du CNRS le paramétrage est le suivant :

- Algorithme de signature de l'assertion d'authentification : ES256 (ECDSA w/ SHA-256)
- Forme d'attestation : « direct » permet d'avoir la vérification du modèle de clé
- Type d'authentificateur : « cross-platform » donc externe.
- Vérification de l'utilisateur : « découragée » c'est du 2FA, pas du « passwordless ».
- Demande de clé résidente : non, utile seulement pour le « loginless ».

3.7.2 Vérification du matériel WebAuthn utilisé

Une vérification du constructeur et modèle de clé de sécurité est réalisée avec le mécanisme d'attestation natif de FIDO U2F / FIDO2. Le constructeur de la clé est authentifié par la vérification de la signature de l'attestation. Les autorités de certification des différents constructeurs sont importées dans un « truststore » java pour cet usage. Cela permet de restreindre l'usage de clés issues de constructeur connus et éventuellement certifiés.

En complément est spécifiée une liste blanche des AAGUID[51] de clés autorisées pour l'authentification. Comme spécifié par FIDO2, un AAGUIDs doit permettre d'authentifier un lot de clé identiques d'un même constructeur. Grace à ce mécanisme il est possible de qualifier le matériel d'authentification utilisable sur la solution et de limiter le matériel utilisé au matériel qualifié à une liste de modèles de clés autorisés. Ce type de mécanisme est assez peu courant dans les usages grand public mais prend tout son sens dans l'usage AAL 3 mis en place dans notre contexte. Les certificats et liste d'AAGUID peuvent être récupérés sur demande auprès des constructeurs[14]. Actuellement une limite connue de Keycloak restreint la liste d'autorisation des AAGUIDs à 6 valeurs, ce qui est suffisant pour l'usage prévu dans un premier temps.

4 Intégration

4.1 Briques techniques principales

Les différents composants sont déployés sur des machines virtuelles hébergées au centre serveur de l'IN2P3[52], sur les infrastructures VMWare de la DSI.

50 Voir le paragraphe « Options for Credential Creation » dans la spécification WebAuthn.

51 AAGUID : Authenticator Attestation GUID. Il s'agit de l'identifiant d'un lot d'authentificateur du même modèle. Il permet à un relying party FIDO de contrôler les modèles d'authentificateurs utilisés mais aussi refuser des lots de matériels réputés compromis.

52 IN2P3 : Institut national de physique nucléaire et de physique des particules

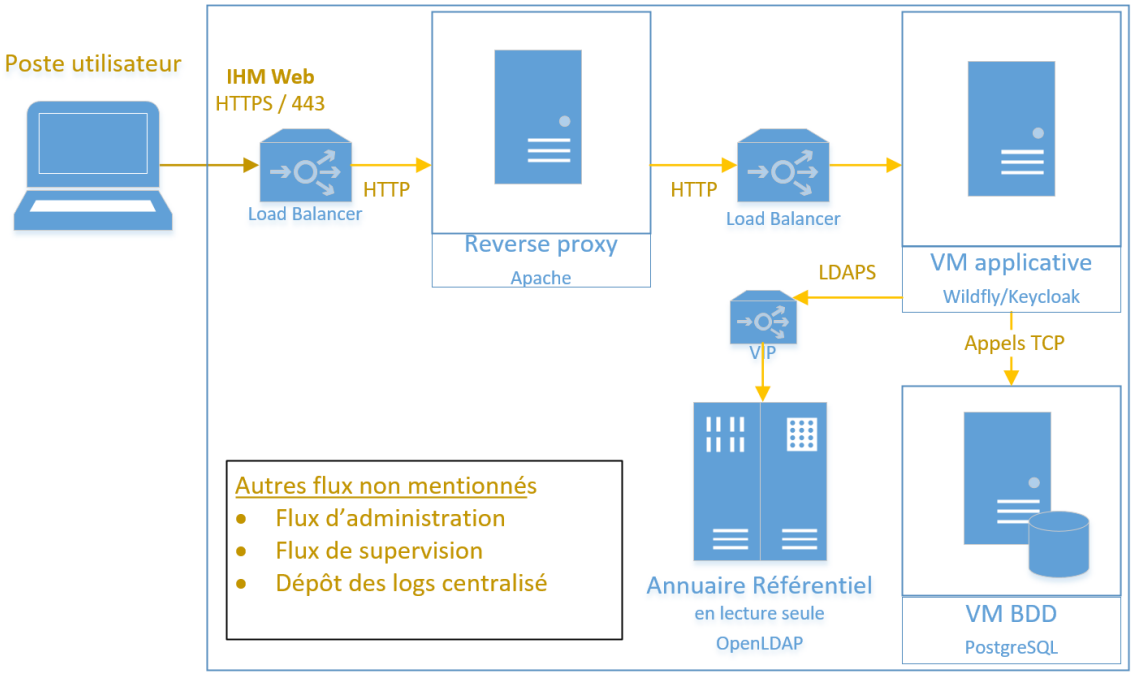

Figure 6: Architecture technique simplifiée

Le reverse proxy utilisé en amont est un serveur web Apache. La base de donnée utilisée est une base relationnelle PostreSQL. Cette même base stocke les données des utilisateurs, l'audit et la configuration de la solution.

Le déploiement de l'application Keycloak et la configuration du conteneur java WildFly sont réalisés via l'outil d'automatisation Ansible. La configuration de Keycloak et le raccordement de nouvelles applications sont réalisés via l'interface graphique d'administration de la solution.

4.2 Intégration avec les données des utilisateurs du CNRS

Keycloak se base sur le même LDAP référentiel que Janus, le SSO principal du CNRS basé sur Shibboleth IDP. Il charge une copie locale des utilisateurs dans sa base de donnée, et les enrichit avec les authentificateurs WebAuthn. Afin de limiter le chargement de données à l'essentiel, un drapeau est ajouté dans l'annuaire pour les utilisateurs autorisés à utiliser l'authentification forte. Ce drapeau est géré via l'interface centralisée de gestion des accès au SI du CNRS, dans l'application Sésame. La synchronisation avec le LDAP est programmée toutes les minutes.

4.3 Intégration avec les applications

Le mode de raccordement choisi pour connecter les applications est le protocole SAML 2.0. Malgré le fait que Keycloak soit très orienté sur le protocole OpenID Connect (OIDC), nous avons choisi de rester sur le protocole plus ancien SAML 2.0. L'avantage de ce dernier est non seulement d'être bien supporté par les applications, mais aussi de conserver une interopérabilité totale entre les différents IDP du CNRS qu'ils utilisent Shibboleth IDP ou Keycloak.

Pour aller dans ce sens, le nom et le contenu des attributs envoyés sont conservés à l'identique de ce qui est fait par Shibboleth. Keycloak ne sait pas se baser sur les métadonnées de la fédération pour choisir la liste des attributs à envoyer par application. Les attributs envoyés sont choisis dans l'interface graphique en définissant des « scope » d'attributs par application.

4.4 Intégration avec d'autres briques de sécurités du SI

Le mécanisme WebAuthn peut fonctionner sans problème derrière un reverse proxy effectuant une rupture protocolaire TLS. Les informations WebAuthn échangées entre le client et le serveur passent dans le corps des requêtes HTTP, soit dans la couche 7 du modèle OSI. La configuration WebAuthn doit obligatoirement spécifier le domaine tel que vu de l'extérieur par l'utilisateur final. Le paramètre correspondant appelé « Relying Party Identifier » est unique est doit obligatoirement correspondre au domaine effectif qui sera utilisé par le navigateur du client[53].

Nous avons rencontré des difficultés pour intégrer le service WebAuthn derrière une technologie de type « VPN SSL clientless » avec réécriture d'URL à la volée. La principale difficulté rencontrée est que le service WebAuthn ne peut fonctionner que sur une seule URL à la fois. Nos tentatives en élargissant le « Relying Party Identifier » sur le domaine de haut niveau « cnrs.fr » sont restées infructueuses[54]. Le service d'authentification renforcé avec WebAuthn fonctionne bien à travers un VPN[55] plus conventionnel utilisant un client.

La piste à l'étude pour s'intégrer avec ce type de VPN « clientless » est de placer l'authentification à l'extérieur du VPN afin de toujours avoir la même origine web pour WebAuthn. Dans ce cas, charge à la terminaison du VPN de contrôler l'authentification et de la transmettre à l'application. Si la terminaison du VPN n'en est pas capable, alors l'utilisateur doit subir une double authentification ou il s'authentifie deux fois, à l'extérieur et à l'intérieur du VPN. Pour éviter cela, la terminaison du VPN doit assurer le rôle de proxy d'authentification. Si plusieurs applications sont utilisées derrière ce même VPN, le service d'authentification n'a alors plus connaissance de l'application finale consommant les attributs. Ce mode de fonctionnement change considérablement le paradigme de l'authentification unique telle qu'elle est utilisée actuellement à la DSI CNRS. Au moment de l'écriture des travaux d'étude de l'intégration du SSO avec ce type de VPN sont encore en cours.

5 Retours d'expérience

5.1 Retours d'expérience technique

5.1.1 Sur WebAuthn

Les décalages des calendriers projets ne nous permettent malheureusement pas d'avoir un retour d'expérience avec les postes et les contraintes des utilisateurs finaux au moment de l'écriture de l'article. Pour l'instant l'utilisation limitée qui en est faite en interne de la DSI du CNRS est plutôt satisfaisante. L'utilisation de la clé est rapide et l'authentification réactive. Les différences d'intégrations et d'interface selon le navigateur et l'OS utilisé sont notables. Le fonctionnement d'un authentificateur sur clé USB peut nécessiter une mise à jour des drivers de l'ordinateur.

5.1.2 Sur Keycloak

Keycloak a la particularité d'être une solution d'authentification assez récente et disponible en open source pour une installation locale dans nos propres centres serveurs. La mise en œuvre est assez rapide et correctement documentée en anglais sur leur site internet[56]. Il y a encore un peu de lourdeur dans la configuration de WildFly[57] qui est le serveur d'application imposé. Cependant la

53 https://www.w3.org/TR/webauthn-2/#relying-party-identifier

54 Un authentificateur enrôlé sur une adresse exemple1.cnrs.fr n'a pas fonctionné pas sur une autre adresse exemple2.cnrs.fr

55 VPN : réseau privé virtuel.

56 https://www.keycloak.org/documentation

57 WildFly est la version communautaire de Jboss EAP

nouvelle version de Keycloak « X » en cours de développement prévoit de ne plus utiliser WildFly et ainsi de faciliter la configuration.

Les fonctionnalités d'authentification forte sont assez récentes dans Keycloak. Elles ne permettent pas toute la flexibilité d'authentification « adaptative », basée sur le risque, qu'on retrouve dans certaines autres solutions commerciales de SSO. On retrouve dans Keycloak un catalogue de facteurs d'authentifications modernes, limité mais suffisant pour un usage courant : en particulier le TOTP sur application mobile, WebAuthn second facteur et WebAuthn passwordless. L'enrôlement et la configuration sont gérés de manière séparée entre le WebAuthn second facteur et le passwordless, ce qui se traduit potentiellement par deux clés logiques dans le même token physique et donc deux enrôlements différents.

Une limitation de Keycloak est sa capacité à s'intégrer avec une fédération d'identités en mode « mesh ». À ce jour il n'est pas capable de s'intégrer automatiquement avec des SP ou IDP SAML à partir d'un fichier global de métadonnées chargé depuis une URL. Il faut déclarer unitairement chaque SP ou chaque IDP dans l'interface d'administration et lister les attributs à fournir au SP. Ce fonctionnement ne permet pas son intégration « automatique » à une fédération locale ou une fédération de NREN[58] de type « mesh ». Les pistes de solutions identifiées pour contourner cette limitation sont :

- utiliser un proxy d'authentification intermédiaire qui gère l'interface avec une fédération,
- planifier un script utilisant les API de Keycloak pour y charger automatiquement les fournisseurs SAML.

La DSI du CNRS n'a pour l'instant explorée aucune de ces deux pistes.

5.2 Intégration de WebAuthn dans Shibboleth

L'intégration officielle de la fonctionnalité WebAuthn est envisagée dans Shibboleth IDP[59]. Cependant, tel que précisé dans cette note de leur wiki, Shibboleth IDP est une solution focalisée plutôt sur l'authentification, que sur la gestion des moyens d'authentification. La solution décrite sur cette note serait de développer uniquement l'authentification, et d'utiliser un fournisseur d'enrôlement externe, alimenté par une source externe.

L'université de Duke aux états unis a développé et mis en production un système d'authentification sans mot de passe basé sur WebAuthn et Shibboleth IDP[60]. Cependant le stockage développé par l'université de Duke pour les enrôlements WebAuthn est spécifique, et l'auteur invite les utilisateurs du projet à développer leur propre système.

Le fait que les procédures d'enrôlement soient spécifiques à chaque organisation et chaque contexte, rend difficile la possibilité de trouver un produit proposant une solution clé en main. L'absence de stockage standard FIDO2 pour rendre les solutions d'enrôlement et d'authentification interopérable constitue aussi un frein. On peut mentionner les travaux de Tsukasa HAMANO en 2019[15], qui a présenté une proposition de schéma Standard LDAP pour WebAuthn. Ce schéma LDAP n'ajoute pas des attributs sur l'objet utilisateur, mais défini une classe d'objet propre « fido2Credential ». L'utilisation d'une base de donnée LDAP avec un schéma standard, ce qui couramment utilisé pour les mots de passe, pourrait constituer une piste sérieuse afin de proposer une intégration standard de WebAuthn dans les solutions d'authentification.

58 National Research and Education Network https://en.wikipedia.org/wiki/National_research_and_education_network

59 Selon le wiki Shibboleth : https://shibboleth.atlassian.net/wiki/spaces/DEV/pages/1210712272/WebAuthn+IdP+Authentication+Plugin

60 Code disponible sur GitHub : https://github.com/sipatel2/shibboleth-webauthn

5.3 Perspectives pour l'authentification forte au CNRS

Pour l'instant les travaux se sont concentrés sur la mise en place d'un IDP et d'un cercle de confiance dédié permettant un niveau de confiance AAL 3 ; ceci afin d'assurer les cas d'usage les plus sensibles de l'établissement.

La suite logique serait de réfléchir à comment ouvrir l'authentification forte plus largement aux autres utilisateurs du SI et pour l'ensemble des applications utilisant les services de SSO du CNRS. Les processus relevant du niveau de confiance AAL 3 pourraient être réservés aux applications les plus sensibles. Une authentification multi facteur de niveaux AAL 1 ou AAL 2, activable à la demande par les utilisateurs finaux et moins contraignante dans les processus de secours est aussi une piste à l'étude.

6 Conclusion

Au cours de ces derniers mois, à partir d'orientations stratégiques fortes pour palier la faiblesse intrinsèque du mot de passe, nous avons implémenté un service transverse d'authentification forte basé sur la technologie FIDO2. Celui-ci servira à sécuriser nos applications les plus sensibles nécessitant un dispositif d'authentification renforcée du niveau AAL 3. Pour cela, nous avons réussi à investir sur le protocole WebAuthn reconnue et en cours de mise en place chez les acteurs grands publics comme chez les acteurs professionnels. Ce protocole permet un large choix d'implémentation technique comme organisationnel. Nous avons pu intégrer ces standards à notre écosystème existant basé sur le WebSSO SAML 2.0 sans refondre nos infrastructures et applications. Du point de vue de l'utilisateur, nous avons opté pour l'adoption de clés d'authentification FIDO2 fiables, peu chère, facile d'emploi et facile à se procurer. Nous avons aussi intégré dans notre application https://sesame.cnrs.fr des modules qui permettent à l'utilisateur et aux administrateurs locaux d'être autonome sur les principaux process liés à la gestion de l'authentification (enrôlement, secours et révocation). La technologie est en cours d'adoption par nos utilisateurs d'applications sensibles.

Même si nous avons augmenté le niveau de confiance de l'authentification, ce dispositif de sécurité en est un parmi d'autres. Cela de nous abstient pas de réaliser les autres pratiques de défense en profondeur (code, infrastructure, organisation, etc.).

Aujourd'hui, nous utilisons donc, pour un coût maîtrisé, une partie seulement du standard WebAuthn. Cette dernière répondant à des besoins spécifiques d'authentification forte en environnement sensible. Mais l'implémentation relativement simple de cette technologie nous permet d'imaginer d'autres scénarios à terme à partir ce même standard. Nous pouvons par exemple penser à élargir l'authentification 2FA FIDO2 à d'autres supports (NFC, etc.), mais nous pourrions aussi élargir la technologie vers le « sans mot de passe » (passwordless), voire le « sans mot de passe » et « sans identifiant » (loginless) pour de l'AAL 1 et AAL 2. Un des enjeux pour le faire sera de réunir les qualités de Keycloak sur WebAuthn et de Shibboleth IDP sur la fédération sur dans une solution unique. Un autre enjeu résidera dans la gestion du changement. En effet, nous passerions d'une échelle de quelques centaines d'utilisateurs à plusieurs milliers, même si l'adoption se ferait sur une action volontaire de l'utilisateur, ce qui engendrera une réflexion nécessaire sur les méthodes pour assurer un support de qualité auprès des utilisateurs. Enfin le fait d'avoir un portail de gestion des moyens d'authentification développé en interne (https://sesame.cnrs.fr) reste un atout pour définir les processus d'enrôlement et de secours les plus adaptés à notre contexte et notre historique.

Bibliographie

[1] ANSSI, Recommandations relatives à l'authentification multifacteur et aux mots de passe, 2021 ; https://www.ssi.gouv.fr/guide/recommandations-relatives-a-lauthentification-multifacteur-et-aux-mots-de-passe/

[2] NIST, Digital Identity Guidelines, 2017 mis à jour en 2020 ; https://pages.nist.gov/800-63-3/

[3] CNIL, Authentification par mot de passe : les mesures de sécurité élémentaires, 2018 ; https://www.cnil.fr/fr/mot-de-passe

[4] ANSSI, La Politique de Sécurité des Systèmes d'Information de l'État (PSSIE), 2014 ; https://www.ssi.gouv.fr/entreprise/reglementation/protection-des-systemes-dinformations/la-politique-de-securite-des-systemes-dinformation-de-letat-pssie/

[5] Directive UE n° 2015/2366 du 25.11.15, dite DSP2 ; https://eur-lex.europa.eu/legal-content/FR/TXT/?uri=CELEX:32015L2366

[6] Guillaume Rousse, Ludovic Auxepaules, MFA et 2FA dans l'IdP Shibboleth, le serveur CAS d'Apereo et les Fédérations, JRES2019, Dijon, décembre 2019 ; https://conf-ng.jres.org/2019/document_revision_5332.html?download

[7] Le Monde Informatique, Jean Elyan, IDG NS, Après le piratage de ses systèmes, RSA propose de remplacer ses jetons SecurID, 8 juin 2011 ; https://www.lemondeinformatique.fr/actualites/lire-apres-le-piratage-de-ses-systemes-rsa-propose-de-remplacer-ses-jetons-securid-33906.html

[8] ANSSI, Le Référentiel général de sécurité (RGS), version 2.0, 2014 ; https://www.ssi.gouv.fr/entreprise/reglementation/confiance-numerique/le-referentiel-general-de-securite-rgs/

[9] YUBICO, Jerrod Chong, NIST publishes new authentication standards, FIDO U2F achieves AAL 3, 22 juin 2017 ; https://www.yubico.com/blog/nist-publishes-sp800-63-3-fido-u2f-achieves-aal3/

[10] Juan Lang, Alexei Czeskis, Dirk Balfanz et Marius Schilder. Security Keys: Practical Cryptographic Second Factors for the Modern Web, 2016 ; https://research.google/pubs/pub45409/

[11] Hidehito Gomi, Yahoo! JAPAN, Bill Leddy, VISA, Dean H. Saxe, Amazon, Recommended Account Recovery Practices for FIDO Relying Parties, Février 2019 ; https://media.fidoalliance.org/wp-content/uploads/2019/02/FIDO_Account_Recovery_Best_Practices-1.pdf

[12] Shibboleth Identity Provider 4 Contributions and Extensions; https://shibboleth.atlassian.net/wiki/spaces/IDP4/pages/1265631834/Contributions+and+Extensions

[13] Adeel Ahmad, Asier Aguado Corman, Maria Fava, Maria V. Georgiou, Julien Rische, Ioan Cristian Schuszter, Hannah Short and Paolo Tedesco. The new (and improved!) CERN Single-Sign-On, 2021 ;

[14] YubiKey Hardware FIDO2 AAGUIDs https://support.yubico.com/hc/en-us/articles/360016648959-YubiKey-Hardware-FIDO2-AAGUIDs

[15] Tsukasa HAMANO. Using LDAP directory for FIDO 2.0, 2019 ; https://ldapcon.org/2019/wp-content/events/presentations/ht-ldap-fido.pdf