Tuomas Hyttinen

FIDO2 SUHTEESSA SALASANAPOHJAISEN TUNNISTAUTUMISEN ONGELMIIN WEB-PALVELUISSA

JYVÄSKYLÄN YLIOPISTO
INFORMAATIOTEKNOLOGIAN TIEDEKUNTA
2023

TIIVISTELMÄ

Hyttinen, Tuomas
FIDO2 suhteessa salasanapohjaisen tunnistautumisen ongelmiin web-palveluissa
Jyväskylä: Jyväskylän yliopisto, 2023, 36 s.
Tietojärjestelmätiede, Kandidaatintutkielma
Ohjaaja(t): Riekkinen, Janne

Salasanapohjainen tunnistautuminen on säilyttänyt asemansa yleisimpänä tunnistautumismenetelmänä jo vuosikymmenien ajan. Uusi salasanaton tunnistautumisstandardi FIDO2 on osoittautunut salasanapohjaisen tunnistautumisen potentiaalisimmaksi syrjäyttäjäksi, jolla on myös edellytyksiä levitä laajasti kuluttajakäyttöön. Tämän kirjallisuuskatsauksen tarkoituksena on selvittää salasanapohjaisen tunnistautumisen parhaita käytäntöjä ja erityisesti sen toteuttamiseen liittyviä toimenpiteitä. Toisena tavoitteena on tutkia mahdollisia FIDO2:n tarjoamia parannuksia näihin havaittuihin löydöksiin.

Tutkimus aloitettiin keräämällä ja validoimalla aineisto tutkimuskysymysten perusteella. Lähteiden laadun varmistamiseksi luotiin lähdematriisi. Lopulliseen aineistoon valittiin 26 vertaisarvioitua tieteellistä julkaisua, sekä paljon standardin kehittäjien aineistoa. Tutkimuksen suurimmat ongelmat liittyivät suhteellisen vähäiseen ja uuteen FIDO2-standardin tutkimukseen.

Tutkimuksen tulokset osoittavat, että ohjelmistokehittäjillä tulee olla laaja tietämys useista eri menettelytavoista salasanapohjaisien ratkaisujen toteutuksessa. Tiedeyhteisö on havainnut puutteita kehittäjien tiedoissa ja taidoissa käytettäessä salausrajapintoja salasanojen tallentamiseen. Myös salasanojen luomista ohjaavien toimenpiteiden toteutus on havaittu puutteellisiksi yleisesti web-palveluissa. FIDO2:n on todettu parantavan salasanapohjaisissa kirjautumismenetelmissä havaittuja puutteita, erityisesti tietovuotoja ja tietojenkalasteluhyökkäyksiä vastaan, poistamalla tarpeen tallentaa salasanoja web-palveluntarjoajien palvelimille. Standardi on todettu tietoturvaltaan paremmaksi ratkaisuksi, kuin mikään muu tunnistautumismenetelmä tähän asti. Menetelmän toteuttamiseen liittyy kuitenkin API-rajapintojen käyttämisen suhteen samoja haasteita kuin salasanapohjaisilla tunnistautumismenetelmillä. Toteutusmenetelmiin, dokumentaatioon ja kehittäjien koulutusmateriaaliin tulee kiinnittää huomiota, jotta salasanapohjaisien menetelmien toteutuksissa tunnistetut virheet eivät toistuisi tulevaisuuden FIDO2 implementoinneissa.

Asiasanat: FIDO2, Tunnistautuminen, Salasana

ABSTRACT

Hyttinen, Tuomas
FIDO2 in relation to the problems of password-based authentication methods in web services
Jyväskylä: University of Jyväskylä, 2023, 36 p.
Information Systems Science, bachelor's thesis
Supervisor(s): Riekkinen, Janne

Password based authentication has endured its place as the most common authentication method for decades. However, the new password-less authentication standard FIDO2 has proven itself as the most potential alternative solution candidate. The purpose of this literature review is to find out the best practices of password-based authentication and especially the measures related to its implementation. Second goal is to investigate the improvements FIDO2 offers to the observed findings. The study started by gathering and validating the material based on the research questions. The sourcematrix was created to ensure quality of the research. After the material was gather, the actual analysis and study was made. The main problems of the study were related to relatively few and new study conducted on the FIDO2-standard.

The results of the study show, that software developers must have extensive knowledge of many different procedures when implementing password-based solutions. The scientific community has noticed deficiencies in the skills of developers when using cryptographic APIs to store passwords. The FIDO2 has been recognized to improve the shortcomings found in password-based login methods, especially in relation to phishing attacks by removing the need to store passwords to web-providers servers. However, the implementation of the method involves the same challenges in relation to API interfaces, as password-based authentication does. Attention should be given to the implementation methods, documentation, and education material for the developers, so that the past mistakes with weak password-based implementations wouldn't be repeated.

Keywords: FIDO2, Authentication, Password

KUVIOT

TAULUKOT

-

SISÄLLYS

1 JOHDANTO

Salasanat ovat vaivanneet tutkijoita jo vuosikymmeniä säilyttäen asemansa ensisijaisena tunnistautumismenetelmänä kaikista tietoturvaongelmistaan huolimatta. (Bonneau, Herley, Oorschot & Stajano, 2012). Salasanojen kaikista yleisimpinä heikkouksina on pidetty loppukäyttäjien tekemiä virheitä kuten salasanojen luovuttamista tietojenkalastelijoille, niiden heikkoa laatua sekä käyttäjien taipumusta uudelleen käyttää salasanoja (Floréncio & Herley, 2007). Salasanojen käyttämistä on yritetty helpottaa monilla tavoilla. Yksi niistä on esimerkiksi salasananhallintaohjat, jotka eivät kuitenkaan ratkaise salasanojen heikkolaatuisuuden tai uudelleenkäyttämisen ongelmaa (Bonneau ym., 2012). Myös yleisesti suosituksi tunnistautumismenetelmäksi noussut kertakirjautuminen (SSO) ei ole havaintojeni mukaan täysin hävittänyt salasanoja kuluttajakäytöstä.

Koska salasanoihin liittyy paljon edellä mainittuja loppukäyttäjiin liittyviä tietoturvaongelmia, ovat tietoturvatutkijat keskittyneet tutkimuksessaan pääosin käyttäjänäkökulmiin (Naiakshina ym., 2017). Tämä on tapahtunut siitä huolimatta, että myös ohjelmistokehittäjät tekevät tunnistetusti virheitä salasanapohjaisen tunnistautumisen toteutuksissa rakentamalla tietoturvan näkökulmasta huonoja tietojenvarastointiratkaisuja (Naiakshina ym., 2017). Ohjelmistokehittäjät käyttävät salasanojen suojaamiseen kryptografisia rajapintoja, joiden huono tuntemus, dokumentointi tai implementointi ovat usein syitä liian heikolle kirjautumistietojen suojaukselle tietovarastoinnissa (Naiakshina ym., 2017). Esimerkiksi valmiit web sovellusten rajapinnat eivät tutkimusten mukaan tarjoa nykyaikana aina riittävää suojausta oletusasetuksillaan, vaan ohjelmistokehittäjän tulisi ymmärtää valita tietoturvallisimmat asetukset itse (Naiakshina ym., 2017). Tästä huolimatta, ohjelmoijat tukeutuvat monissa tapauksissa sovellusten oletusasetuksiin puutteellisen tiedon ja suunnittelun vuoksi. (Ntantogian, Malliaros & Xenakis, 2019).

Vaikka salasanapohjaisen tunnistautumisen ongelmia tunnistetaan hyvin, on se pitänyt asemansa yleisimpänä tunnistautumismenetelmänä. Tutkimusten mukaan olemassa olevien vaihtoehtoisten tunnistautumistapojen hyödyt ovat vielä liian pieniä, jotta ne syrjäyttäisivät salasanoihin perustuvat tunnistautumismenetelmät (Bonneau ym., 2012). Yksi syy tälle on se, että nykyisillä

vaihtoehtoisilla kirjautumismenetelmillä, joissa tietoturvan taso paranee, käytettävyyden taso huononee (Bonneau ym, 2012). Salasanatonta kirjautumisstandardia kehittävä Fido Alliance on jo pitkään yrittänyt ratkaista tätä ongelmaa. Järjestö julkaisi vaihtoehtoisen web-tunnistautumisstandardi FIDO2:n huhtikuussa 2018. FIDO2-standardia on pidetty salasanapohjaisen kirjautumisen uusimpana potentiaalisena haastajana, koska sillä nähdään tutkimusten mukaan olevan realistiset mahdollisuudet laajempaan leviämiseen myös kuluttajien keskuudessa (Lyastani ym., 2020). Tämän arvion tutkijat olivat kirjoittaneet jo ennen kuin FIDO2-standardiin perustuvat älypuhelimenkautta biometrisesti toimivat tunnistautumismallit alkoivat saapua kuluttajien saataville vuonna 2022. Älypuhelimen kautta toimivan tunnistautumisen myötä myös FIDO2:n aikaisempana tunnistetut käytettävyysongelmat kokivat mullistavan parannuksen.

Tutkimukseni tarkoituksena on tehdä kirjallisuusuuskatsaus, jossa tutkitaan salasanapohjaisen kirjautumisen toteutuksen parhaita käytänteitä, nykytoteutusta sekä yleisimpiä heikkouksia web-palveluissa. Toisena tavoitteena on selvittää kuinka FIDO2-standardi vastaa näihin tieteessä tunnistettuihin ongelmiin. Tutkimuksesta jätetään pois käyttäjän ja tietovaraston välisten tietoliikenneyhteyksien suojaaminen ja keskitytään pääosin tunnistautumistietojen varastoinnin toteutukseen, sekä vaiheisiin, joissa tarvitaan ohjelmistokehittäjän työtä tai asiantuntemusta. Näin ollen myös käyttäjien käytettävyys jätetään vähemmälle huomiolle. Lisäksi tutkimus rajataan koskemaan web-palveluita ja ympäristöjä. Aihe on ajankohtainen, sillä ensimmäiset FIDO2 protokollaan perustuvat, kuluttajille laajempaan käyttöön tarkoitetut rajapinnat ovat alkaneet ilmestyä ohjelmistokehittäjien hyödynnettäväksi. Esimerkiksi Apple julkaisi FIDO2 teknologiaan perustuvan Passkeys-rajapinnan iOS16 järjestelmää tukeville laitteilleen heinäkuussa 2022 (Apple, 2023).

Tämän kandidaattityön tavoitteena on vastata seuraaviin tutkimuskysymyksiin:

- *Mitkä ovat salasanapohjaisen tunnistautumisen parhaat käytänteet ja tietoturvaongelmat web-palveluissa toteutuksen näkökulmasta?*
- *Tarjoaako FIDO2-standardi ratkaisuja tunnistettuihin ongelmiin?*

Tutkimuksen tarkoitus on tarjota lukijalle jäsennelty ja kompakti kokonaiskuva salasanapohjaisen kirjautumisen toteutuksesta ja siihen liittyvistä ongelmista, sekä analysoida lähdeanalyysin pohjalta, tarjoaako FIDO2-standardi ratkaisuja tunnistettuihin ongelmiin. Tutkimuksessa painotetaan ohjelmistokehittäjien näkökulmaa, joten tarkoituksena on osaltaan selvittää, helpottaako FIDO2-standardi ohjelmistokehittäjien työtä ja tarjoaako se selkeitä etuja suhteessa salasanapohjaisen kirjautumisen toteuttamiseen.

Työ on luonteeltaan kirjallisuuskatsaus, jonka lähdeaineiston kerääminen toteutettiin hyödyntämällä pääosin Jykdok-, Scopus ja Google Scholarin tarjoamia tietokantoja. Julkaisijoiden validiteetti on pääosin arvioitu hyödyntäen Julkaisufoorumin tarjoamaa tietoa. Koska tutkittava teknologia on hyvin uusi, on siitä saatava aineisto uutta ja melko vähän viitattua. Mikäli Julkaisufoorumi ei

ole tarjonnut tutkimuksessa käytetystä julkaisijasta tietoa, on julkaisija erikseen tutkittu tahoksi, joka voidaan todeta antamiensa tietojen perusteella toteuttavan anonyymiä vertaisarviointia ja korkeaa tieteellistä laatua. Salasanapohjaisten menetelmien osalta lähdeaineistosta rajattiin pois kaikki ennen vuotta 2000 julkaissut lähteet. Lähteiden valinnassa painotettiin vertaisarvioituja ja eniten lainattuja tieteellisiä julkaisuja. FIDO2:een liittyvä aineisto on teknologian julkaisuajankohdan vuoksi huomattavasti uudempaa ja tätä myötä vähemmän lainattua. Koska FIDO2 on protokolla, näyttelee teknologian kehittäjän (Fido Alliance) aineisto suurta roolia siihen liittyvässä tiedossa tieteellisten julkaisujen lisäksi. Myös teknologia-alalla asiantuntijoiden yleisesti hyväksi luokittelemat hyviä käytänteitä tarjoavat tahot kuten Google, NIST ja OWASP on luokiteltu lähes standardin omaisina tahoina tarjoavan alan parhaita käytänteitä, joita on hyödynnetty tässä tutkimuksessa. Lopulliseen aineistoon valittiin 26 vertaisarvioitua tieteellistä julkaisua, useita standardi dokumentteja ja yksi Passkeys-teknologiaa esittelevä verkkosivu.

Tutkimus aloitettiin hakusanojen valinnalla ja aineiston keruulla. Tämän jälkeen aineistosta rajattiin pois materiaali, joka ei täyttänyt edellä mainittuja laatustandardeja. Aineisto sisältö analysoitiin tarkemmin ja rajattiin kahteen osaan, jotka olivat salasanapohjaiseen kirjautumiseen liittyvät aineistot, sekä FIDO2-teknologiaan liittyvät aineistot. Aineistoista luotiin taulukkomuotoinen matriisi, jonka avulla pidettiin huolta, että kaikille lähteille on suoritettu laadullinen arviointi. Lopuksi valikoituneet artikkelit analysoitiin tutkimuskysymysten mukaisesti.

Kandidaatintutkielman ensimmäinen sisältöluku keskittyy määrittelemään salasanpohjaisen tunnistautumisen toteutuksen nykykäytäntöjä, tunnistautumista yleisesti, sekä FIDO2-teknologiaa. Toinen sisältöluku keskittyy salasanapohjaisten tunnistautumismenetelmien haasteisiin. Kolmannessa luvussa kerrotaan FIDO2:n ratkaisuista salasanapohjaisen tunnistautumisen ongelmiin. Viimeisessä luvussa tutkimuksen sisältö kootaan tiiviiksi yhteenvedoksi.

2 TUNNISTAUTUMISMENETELMÄT WEB-PALVELUISSA

Tässä luvussa perehdytään tunnistautumismenetelmiin yleisesti, nykyaikaiseen salasanapohjaiseen kirjautumiseen erityisesti ohjelmistokehittäjän näkökulmaa painottaen, sekä FIDO2-standardiin. Vähemmälle huomiolle jätetään menetelmien käytettävyys asiakkaan näkökulmasta. Mainittakoon kuitenkin erikseen, että tunnistautumismenetelmien toteuttamiseen liittyy paljon käyttäjien tunnistautumisprosessin ohjaamiseen liittyviä, ohjelmistokehittäjän vastuulla olevia tehtäviä, joita tutkimuksessa pyritään käsittelemään. Lisäksi käsitellään salasanapohjaisen tunnistautumisen toteuttamiseen liittyviä parhaita käytänteitä.

2.1 Tunnistautuminen

Tunnistautuminen on nykyaikaisten web-palveluiden perustoimenpide, jolla palveluntarjoaja pyrkii varmistamaan tietojen luovuttamisen oikeille henkilöille. Lowen (1997) mukaan tunnistautumisella tarkoitetaan yleisesti sitä, että yhden siihen osallistuvan osapuolen pitää varmistua toisen osapuolen identiteetistä. Hänen mukaansa jokseenkin kiisteltyä kuitenkin on, kuinka varmalla tasolla esimerkiksi palveluntarjoajan täytyy tietää käyttäjän tila tunnistautumishetkellä. Tällä tutkija tarkoittaa kysymystä siitä, onko kirjautunut käyttäjä juuri se, joksi palveluntarjoaja hänet olettaa. Hän kuitenkin toteaa erilaisten tunnistautumistilanteiden hyvin paljon määrittelevän tarvittavaa tunnistautumisen tasoa (Lowen, 1997). National Institute of Standards and Technology (NIST) (2020) määrittelee tunnistautumisen hyvin samoin tavoin, kuin Lowen. Kuitenkin NIST määrittelee tunnistautumisen olevan lisäksi prosessi, jolla varmistetaan tiedon lähteen lisäksi sen eheys istunnoissa, joissa käsitellään kommunikaatiota, viestejä, asiakirjoja tai tallennettua dataa tai varmistutaan järjestelmän kanssa kommunikoivasta entiteetistä (NIST, 2020, s.7).

Tunnistautumismenetelmät jaetaan tyypillisesti kolmeen kategoriaan sen mukaan, mitä menetelmää ne käyttävät tunnistautumiseen. Kategoriat ovat 1)

jotain mitä käyttäjä tietää, 2) jotain mitä käyttäjällä on ja 3) jotain mitä käyttäjä on (Dong, Clark & Jacob, 2008; Maddox & Moschetto, 2019b; NIST, 2017a). NIST:in (2017b) määrittelyn mukaan salasanapohjaiset kirjautumismenetelmät kuuluvat kategoriaan ”jotain mitä käyttäjä tietää”, sillä ne perustuvat käyttäjän luoman yksilöidyn salasanan muistamiseen. NIST:in määritelmien perusteella fyysisiin kirjautumisavaimiin perustuvat kirjautumismenetelmät, kuten myöhemmin esiteltävä FIDO2-standardi, perustuvat ”johonkin mitä käyttäjällä on”. Biometriikka, kasvojentunnistus ja puheentunnistus ovat NIST:in mukaan tunnistautumismenetelmiä, jotka pohjautuvat ”johonkin mitä käyttäjä on”, sillä ne perustuvat käyttäjän fyysisiin ominaisuuksiin (NIST, 2017b). FIDO2-tunnistautuminen hyödyntää toiminnassaan myös biometriikkaa, joten NIST:in (2017b) määritelmien mukaisesti se toteuttaa ”jotain mitä käyttäjällä on” kategorian lisäksi ”jotain mitä käyttäjä on” kategorian.

2.2 Salasanapohjaisen tunnistautumisen toteutus nykyaikana

Tunnistautumista on jo pitkään hallinnut salasanoihin pohjautuva tunnistautuminen. Tämä on selkeästi tiedeyhteisön julkaisuista tunnistettava käsitys, joka on havaittavissa muun muassa Florencion ja Herleyn (2007), Bonneaun ym. (2012) ja Bonneaun ym. (2015) artikkeleissa, sekä Woodsin (2016) väitöskirjassa. Salasanapohjaisissa tunnistautumismenetelmissä salasana tyypillisesti varastoidaan palveluntarjoajan tietovarastoon (NIST, 2017b). Käyttäjän syöttämää salasanaa verrataan varastosta löytyvään salasanaan tunnistautumisen yhteydessä (OWASP, 2023b). Oikeaoppisesti salasana suojataan hajautusalgoritmilla ennen sen tallentamista tietovarastoon suojattuja yhteyksiä käyttäen (NIST, 2017b; OWASP, 2023a).

Nykyaikana ohjelmistokehittäjät käyttävät tyypillisesti kryptografisia rajapintoja (API, eng. *Application programming interface*) salatakseen arkaluonteisen kirjautumisdatan tietovarastointiin liittyvissä toimenpiteissä (Naiakshina ym., 2017; OWASP, 2023a). Kryptografisella rajapinnalla tarkoitetaan erillistä ohjelmaa, joka automaattisesti salakirjoittaa sille syötetyn datan. Kryptografiset rajapinnat levisivät yleiseen käyttöön 90-luvulla tarjoten kirjastoja kryptografisille toiminnoille kuten salaukselle, digitaalisille allekirjoituksille ja turvallisille yhteyksille (Green & Smith, 2016). Erilaisille koodikielille ja ratkaisuille on tyypillisesti tarjolla hieman erilaisia ratkaisuja. Nadin, Krugerin, Mezinin ja Bodden (2016) tekemän tutkimuksen mukaan ohjelmistokehittäjien täytyy yhä tyypillisesti todentaa käyttäjiä, varastoida ja suojata monenlaista dataa, sekä luoda suojattuja yhteyksiä, vaikka he käyttäisivät rajapinta ratkaisuja implementoidessaan Java -kielistä ratkaisua. Myös Greenin ja Smithin (2016) mukaan pelkkä huoleton rajapintojen käyttäminen ei riitä, vaan nykyaikaiset kryptografiset rajapinnat edellyttävät ohjelmistokehittäjiltä niiden korkeaa asiantuntemusta. Vaikka ohjelmistokehittäjät käyttävät valmiita rajapintoja toimintojen toteutukseen, on ohjelmistokehittäjä lopulta aina vastuussa loppukäyttäjän salasanan turvaamisesta

(Naiakshina ym. 2019). Siitäkin huolimatta, että ohjelmistokehittäjien käyttämät kryptografiset rajapinnat huolehtivat monista parhaisiin käytänteisiin liittyvistä toimenpiteistä, on tiedeyhteisössä tunnistettu rajapintojen heikkoon tietoturvalliseen käyttämiseen liittyviä ongelmia (Green & Smith, 2016). Näihin ongelmiin palataan vielä myöhemmin tässä tutkimuksessa.

2.3 Salasanojen käsittelyn ja tietovarastoinnin parhaat käytänteet

Koska ohjelmistokehittäjien on edelleen ymmärrettävä toimintaprosesseja rajapintojen takana (Naiakshina ym. 2019), täytyy heidän ymmärtää näiden prosessien parhaat toteutustavat varmistaakseen tietoturvallisen tunnistautumisen toteutuksen. Tämän turvaamiseksi on ohjelmistokehittäjille luotu alan parhaita käytänteitä, joita tarjoavat teknologia-alan suuret toimijat kuten Google, OWASP (*Open Worldwide Application Security Project*) ja NIST (*National Institute of Standards and Technology*). Analyysini pohjalta käytänteet jakautuvat pääosin kolmeen osaan, jotka ovat 1) salasanan luomisen ohjaaminen, 2) turvallinen tietovarastointi ja 3) datansiirron suojaaminen.

2.3.1 Salasanan luomisen ohjaaminen

Salasanat ovat tyypillisesti käyttäjän luomia (Bonneau ym, 2015). Leen, Sjöbergin & Narayanan (2022) mukaan web-palvelut käyttävät tiedeyhteisön laajasti tutkimia ohjelinjoja ohjatakseen käyttäjää salasanan luonnin yhteydessä. Näitä suosituksia ovat heidän mukaansa estolistat, koostumussäännöt ja voimamittarit. Tutkijoiden mukaan estolistat koostuvat tietoturvakentällä yleisesti tunnistetuista heikoista salasanoista ja niiden avulla tunnistetaan yleisimmät ja liian heikot salasanat, sekä tyypillisesti estetään käyttäjää käyttämästä listalla esiintyviä merkkijonoja. Koostumissäännöt määrittelevät heidän mukaansa salasanassa pakollisina käytettäviä merkkejä vaikeuttaen salasanojen murtamista väsytyshyökkäyksillä (*brute-force attack*), pidentämällä salasanojen arvaamiseen tarvittavaa laskenta-aikaa. Voimamittareilla määritellään tutkijoiden mukaan salasanan laatua ja tulostetaan mittarin tulos käyttäjälle näkyville esimerkiksi graafiseen käyttöliittymään (Leen, Sjöbergin & Narayanan, 2022). Google Cloud arkkitehdit Maddox ja Moschetto (2019b) perustelevat Google Cloudin ohjeistuksessa monien käyttäjän ohjaamiseen liittyvän toimen olevan tärkeä osa parhaita käytäntöjä. Tällaisina toimenpiteinä he mainitsevat esimerkiksi seuraavat kohdat:

1) Laajan merkistön (*UTF-8*) käyttäminen.
2) Salasanan suuren minimipituuden ja erittäin suuren maksimipituuden mahdollistaminen.
3) Voimamittarin näyttäminen graafisesti käyttäjälle.
4) Salasanan tarkistaminen suhteessa yleisimpiin heikkoihin salasanoihin.
5) Salasanan liittämisen salliminen.

6) Käyttäjän ja tietovaraston salasanahajautuksen vertaaminen merkkijonomuotoisen vertaamisen sijaan.
7) Salasanojen uudelleenkäytön estäminen.
8) Salasanan palauttaminen vain uuden salasanan luomisen avulla.

Jälkimmäisellä kohdalla arkkitehdit tarkoittavat sitä, että palautussalasanaa ei lähetetä käyttäjälle esimerkiksi sähköpostin välityksellä. NIST:in (2017a) salasanojen luomisen ohjaamiseen tarjoamien suositusten mukaisesti salasanojen tulisi olla vähintään kahdeksan merkkiä pitkiä ja luotuja salasanoja tulisi verrata Leen ym. (2022) esittämään tapaan listaan yleisistä käytetyistä salasanoista. NIST:in suositusten mukaan luomisen yhteydessä tulisi myös käyttää käyttäjälle näkyviä visuaalisia voimamittareita, jotka kertovat salasanan vahvuuden. Palveluntarjoajan tulisi myös NISTI:n mukaan toteuttaa mekanismi, joka rajoittaa kirjautumista usean epäonnistuneen yrityksen jälkeen. NIST ei suosittele muita luomista ohjaavia toimia, kuten salasanan merkkejä ohjaavia sääntöjä. Palveluntarjoajan tulisi suosituksen mukaan kuitenkin mahdollistaa esimerkiksi salasanojen liittäminen tekstikenttiin. Lisäksi palveluntarjoajan tulee NIST:in mukaan käyttää suojattuja yhteyksiä salasanojen todennukseen (NIST, 2017a). Vertailtaessa edellä esiteltyjä Googlen (2019) ja NIST:in (2017a) tarjoamia parhaita käytänteitä salasanan luomisen ohjaamiselle, voidaan todeta niiden sisältävän pääpiirteisesti hyvin samoja ohjelinjoja.

2.3.2 Salasanojen turvallinen tietovarastointi

Salasanan välittäminen palvelimelle, salasanan asianmukainen suojaaminen tietovarastoinnissa, sekä käyttäjän tunnistaminen vertaamalla varastoitua ja syötettyä salasanaa ovat yksiä yleisimpiä toimenpiteitä web-kehittäjille (Naiakshina ym., 2017). Kehittäjille suunnattuja parhaita käytänteitä salasanojen käsittelylle tallentamisen yhteydessä Maddoxin ja Moschetton (2019b), OWASP:in (2023a) ja NIST:in (2023) tarjoaman materiaalin pohjalta ovat:

1) Salasanojen suolaaminen (eng. *salting*)
2) Salasanojen pippurointi (eng. *peppering*)
3) Hajauttaminen hajautusaloritmilla (eng. *hashing*)
4) Hajautettujen salasanojen iteroiminen (eng. *key stretching*)
5) Tekstimuotoisen salasanan poistamisesta huolehtiminen hajautuksen jälkeen.

Suolaus operaatiossa käyttäjän tekstimuotoiseen salasanaan lisätään satunnaisia kirjaimia, numeroita ja merkkejä, ennen kuin se hajautetaan ja tallennetaan tietovarastoon (OWASP, 2023a). Suola-osan tulisi olla vähintään 32-bittiä pituudeltaan ja valittu satunnaisista merkeistä (NIST, 2017a).

OWASP:in (2023a) mukaan suolatut salasanat hajautetaan hajautusalgoritmilla. Baumanin ja Linin (2015) mukaan hajauttaminen pakottaa mahdollisen murtajan suorittamaan äärimmäisen raskaiden ja monimutkaisten, käytännössä lähes mahdottomien laskelmien suorittamisen, jotta alkuperäinen salasana

saadaan selville. Salasanojen hajautusta suunniteltaessa on aina tehtävä valinta laskenta-ajan ja suojaustason välillä, sillä korkeamman suojauksen tarjoavat algoritmit tarvitsevat suuremman laskentatehon (Kabir & Elmedany, 2022). Naiakshinan ym. (2017) mukaan nykyisesti suositeltavat salaustavat ovat PBKDF2 (*Password Based Key Derivation Function*), bcrypt ja scrypt, joissa hajautettu salasana myös iteroidaan. He kuitenkin näkevät Argon2 hajautusalgoritmin parhaana nykyaikaisena vaihtoehtona, sillä siinä on lisäksi parannettu suojausta välimuistin ajoitushyökkäystä (eng. *cache-timing attack)* ja garbage-collector hyökkäyksille (Kabir & Elmedany, 2022). OWASP:in (2023a) ohjeistus tukee Naiakshinan ym (2017) näkemystä parhaista algoritmeista lisäten perustelun siitä, että Argon2, bcrypt ja PBKDF2 hoitavat automaattisesti salasanojen suolaamisen. NIST:in (2017a) määrittelee hajautustavan valinnan suppeammin toteamalla, että salaustavan tulisi käyttää algoritmia, jonka avulla salatun tietueen murtamiseksi vaaditaan erittäin suurta muistikuormaa. Lopulta sekä hajautuksessa käytetty salainen suolaosa, että hajautettu kokonaisuus tulee tallentaa erillään tietovarastoon käyttämällä muistiin tallennettua salaista todentajaa, jota kutsutaan pippuriosaksi (NIST 2017a). Yhdessä yleisimmistä pippurointimenetelmistä luotu salasanhajautus suojataan symmetrisellä avaimella (pippuri), ennen kuin hajautus tallennetaan tietovarastoon (OWASP, 2023a). OWASP:in (2023a) mukaan pippuriosat tulisi tallentaa erillisiin salaisiin tietovarastoihin ja niiden salausta tulee myös harkita. Mahdollinen pippuri on siis merkkijono, joka sijaitsee sovelluksen lähdekoodissa, eikä itse suolatussa salasanan merkkijonossa. (Kabir & Elmedany, 2022).

2.3.3 Datansiirron suojaaminen

Maddox ja Moschetto (2019b) painottavat Google Cloudin ohjeistuksessa salattujen tiedonsiirtoyhteyksien käytön tärkeyttä, sekä salasanahajautusten salausta tietovarastossa. Kaikki yhteydet palvelimen kanssa tulee heidän mukaansa tehdä käyttäen HTTPS-protokollaa. Arkkitehtien mukaan TLS-yhteyttä tulee käyttää, kun vaihdetaan dataa palvelimien tai palveluiden välillä. Nämä palvelut ovat heidän mukaansa tarjolla yleisesti käytössä olevilla tahoilla kuten Google Cloudissa, Amazon Web Servicessä sekä Azuressa (Maddox & Moschetto, 2019b). Myös OWASP (2023b) suosittelee TLS:n käyttöä kaikissa tunnistautumista vaativissa yhteyksissä. TLS:n käyttö vaikuttaa hyvin standardilta suositukselta, sillä myös NIST (2020) suosittelee sen käyttämistä OWASP:in (2023b) ja Maddoxin ja Moschetton (2019b) tapaan.

Datansiirron suojaaminen on tärkeä ja laaja osa tunnistautumisprosessia. Johdannossa esiteltyjen aiherajausten mukaisesti sen osuus kuitenkin jätetään pienemmälle tarkastelulle tässä tutkimuksessa.

2.4 Salasanapohjaisten tunnistautumismenetelmien arviointi

Useat tahot, kuten edellä esitellyt NIST, OWASP ja Google Cloud tarjoavat parhaita käytänteitä ja ohjeistuksia salasanojen käsittelylle ja tietovarastoinnille. Salasanapohjaisen tunnistautumisen toteutuksesta tehdyn tutkimukseni pohjalta on havaittavissa, että ohjelmistokehittäjien on huolehdittava ja ymmärrettävä monia eri vaiheita tunnistautumisen implementoinnin yhteydessä. Koska salasanapohjainen kirjautuminen on vallinnut tunnistautumisen kenttää pitkään, on myös erilaisia tunnistautumistapoja kehitetty useita ja eri tunnistautumistavat tarjoavat erilaisia etuja (Bonneau ym., 2012).

Bonneau ym. (2012) kehittivät arviointimenetelmän tunnistautumismenetelmille arvioidakseen eri menetelmien käytettävyyttä, käyttöönotettavuutta ja tietoturvaa. He arvioivat menetelmän pohjalta 35 tunnistautumismenetelmää ja muodostivat tutkimustuloksista matriisimuotoisen taulukon (*Liite 1*) (Bonneau ym., 2012). Menetelmä on aineistoanalyysini perusteella saanut vahvan jalansijan tiedeyhteisössä. Tähän perusteluna on se, että menetelmää hyödyntäviä ja siihen viittaavia tutkimuksia on paljon. Tästä esimerkkinä on Lyastanin ym. (2020) tekemä tutkimus FIDO2-standardista käytettäessä Yubicon kirjautumisavainta. Bonneaun ym. (2012) tekemän tutkimuksen mukaan salasanapohjaiselle kirjautumiselle ei löytynyt varteenotettavaa haastajaa, jolla olisi realistisia mahdollisuuksia syrjäyttää tällä hetkellä suosituimpana tunnistautumismenetelmänä käytettäviä perinteisiä salasanoja. Tutkimuksessa otettiin huomioon myös useita menetelmiä, joilla pyrittiin parantamaan salasanapohjaista kirjautumista. Tällaisia menetelmiä tutkimuksessa olivat esimerkiksi salasananhallintaohjelmat, joilla pyritään hallinnoimaan käyttäjän salasanoja keskittämällä ne hallintasovellukseen yhden pääsalasanan taakse. Salasanojen vahvuus suhteessa muihin menetelmiin on tutkijoiden mukaan niiden käyttöönotettavuus, joka on myös syynä muiden kirjautumismenetelmien heikkoon suosioon suurien käyttäjämassojen keskuudessa. Käyttöönotolla tutkijat tarkoittavat sitä, että muistin kautta toimiva salasana on heti käytettävissä, eikä käyttäjän tarvitse suorittaa monia prosessivaihteita kirjautumisen yhteydessä (Bonneau ym., 2012).

2.5 Kertakirjautuminen

Bonneaun ym. (2012) tutkimuksen kirjoitushetkellä parhaiten vertailussa suhteessa perinteisiin salasanoihin pärjäsivät kertakirjautumiseen (SSO, eng. *Single Sign-On*) pohjautuvat tunnistautumismenetelmät, jollaisiksi laskettiin federoidut SSO-menetelmät, salasananhallintaohjelmat ja jotkin tunnisteita (eng. *tokens*) hyödyntävät menetelmät. Federoidussa SSO-menetelmässä käyttäjä uudelleenohjataan luotetun web-palvelun palvelimelle, jossa tunnistautuminen tapahtuu (Bonneau ym. 2012). Maddoxin & Moschetton (2019a) mukaan tällä tavoin yksitäisen web-palvelun ei tarvitse huolehtia käyttäjän kirjautumistietojen hallinnasta, vaan kirjautuminen voidaan hoitaa yhteistyössä palvelua tarjoavan

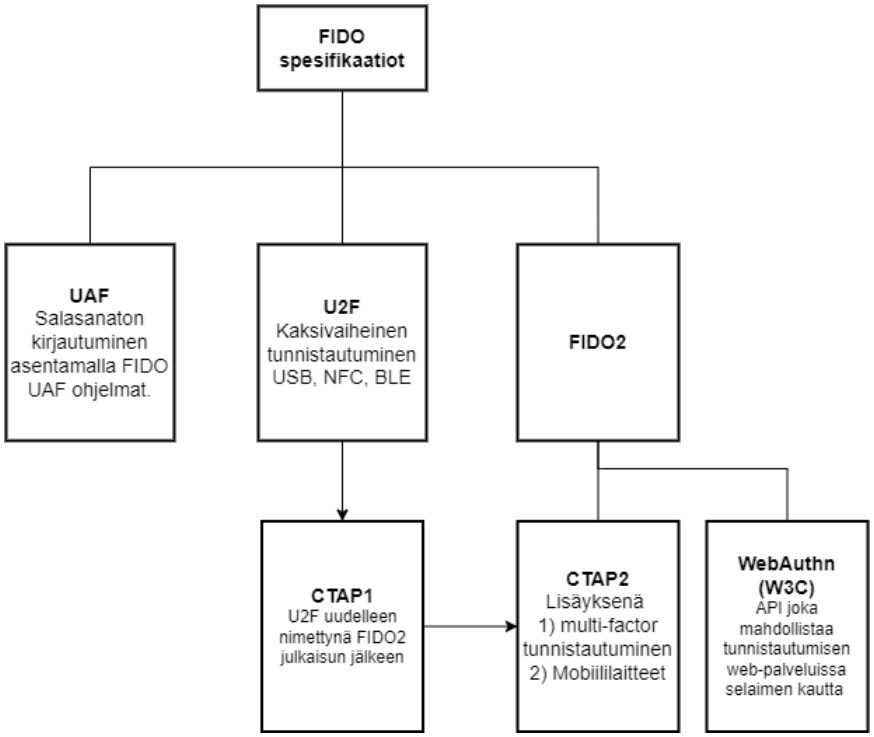
kolmannen tahon kanssa. Käyttäjän ei siis tarvitse luoda web-palveluun salasanaa, vaan kirjautuminen hoidetaan tunnistautumispalvelussa jo käytössä olevilla käyttäjätunnuksilla (Maddox & Moschetto, 2019b). Kertakirjautumisen käyttö vähentää käyttäjän luomien salasanojen määrä, sekä vapauttaa palveluntarjoajan palvelun salasanojen varastoinnista. Nykypäivänä tunnettuja kertakirjautumista tarjoavia palveluita ovat esimerkiksi Google, Facebook ja Twitter (Maddox & Moschetto, 2019a). Bonneaun ym. (2012) tutkimuksen mukaan SSO tarjosi käyttäjille parempaa käytettävyyttä, mutta sen tarjoamat tietoturvaparannukset jäivät vähäisiksi.

2.6 FIDO2, salasanattoman tunnistautumisen standardi

Tunnistautumismenetelmänä FIDO2-standardin perusolemus rakentuu tunnistautumiseen käyttäjän laitteen hallussa olevan yksityisen avaimen avulla. Siispä standardi toteuttaa NIST:in (2017b) määrittelemää ”jotain mitä käyttäjällä on” -kategorian pohjaista tunnistautumista, toisin kuin salasanat, jotka perustuivat ”johonkin mitä käyttäjällä tietää”. Koska yksityinen kirjautumisavain sijaitsee ainoastaan käyttäjän laitteella, eikä sitä ikinä välitetä laitteen ulkopuolelle tunnistautumisen yhteydessä, ei palveluntarjoaja saa käsiteltäväkseen käyttäjän kirjautumiseen käyttämä salaisuutta prosessin aikana (Fido Alliance, 2023a). Fido Alliancen (2020) mukaan FIDO2-standardin päätavoite on ollut mahdollistaa peruskäyttäjille tietoturvallinen, helppokäyttöinen ja yksityinen tunnistautumismenetelmä, jossa tunnistautuminen tapahtuu lokaalisti käyttäjän laitteella tunnetuimpien selaimien avulla. Fido Alliance myös painottaa, että standardin kehityksessä on ollut tärkeää, että toimintamekanismista tulee avoin, skaalautuva ja yhteen toimiva (Fido Alliance, 2020).

2.6.1 Standardin sisältö

Fido Alliance on kehittänyt joukon protokollia (*FIDO-protokollat*), jotka mahdollistavat salasanattoman kirjautumisen. Tämän työn kirjoitushetkellä FIDO protokollia on julkaistu kolme. Nämä protokollat ovat 1) Fido Universal Second Factor (*U2F*), 2) Fido Universal Authentication Framework (*FIDO UAF*), sekä 3) FIDO2. FIDO2 perustuu World Wide Web Consortiumin (*W3C*) Web Authentication (*WebAuthn*) spesifikaatioon, sekä Fido Alliancen Client-to-Authenticator Protokollaan (*CTAP2*) (Fido Alliance, 2018). CTAP2 on uusin versio CTAP1:stä, joka puolestaan on FIDO2-standardin julkaisemisen myötä uudelleen nimetty U2F standardi (Fido Alliance, 2018). Antaakseni selvennystä tähän melko monimutkaiseen standardien väliseen riippuvuuteen, olen luonut riippuvuuksista kuvan ’FIDO standardien väliset riippuvuudet’ (kuvio 1*)*. FIDO2 on tunnistautumisstandardi, joka sisältää useita Fido Alliancen aikaisemmin kehittämiä standardeja.



KUVIO 1 FIDO standardien väliset riippuvuudet

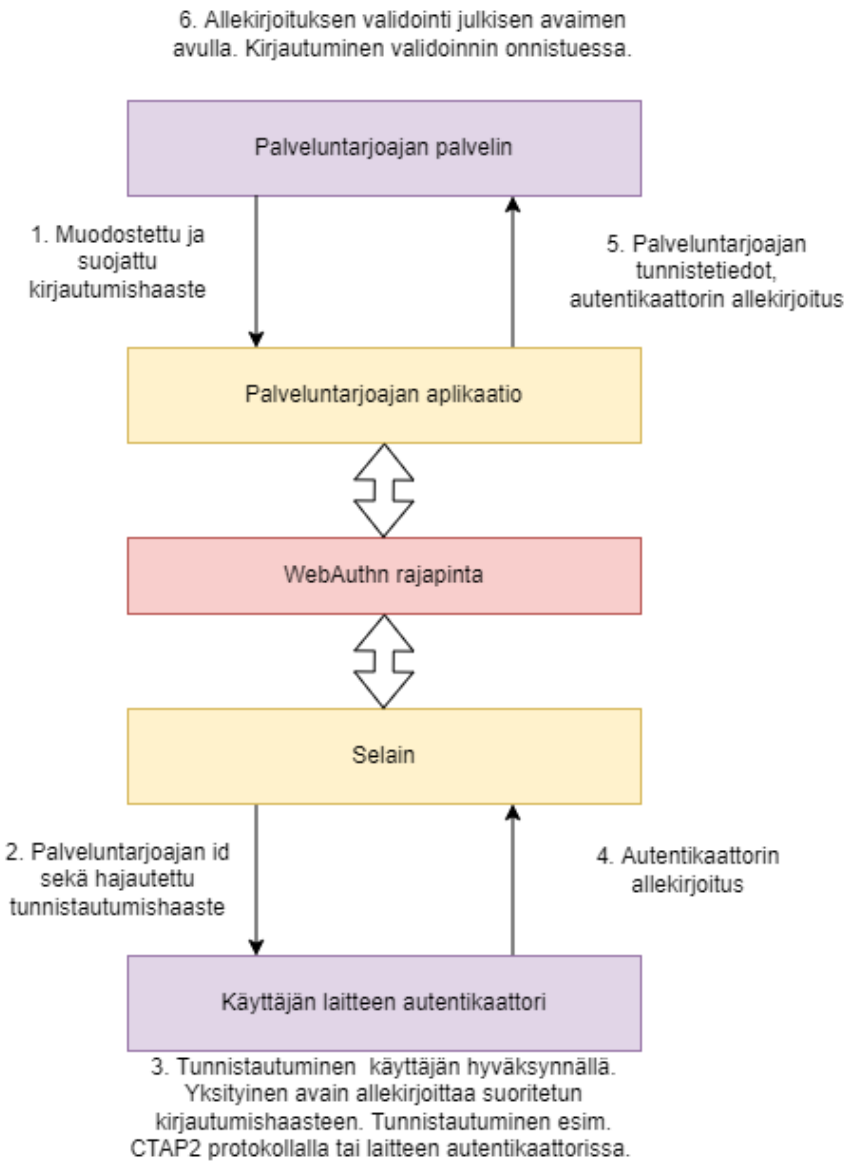
Fido Alliancen (2023b) mukaan WebAuthn tarjoaa standardin mukaisen web-rajapinnan, joka on sisäänrakennettu moderneihin FIDO2-integroituihin selaimiin tarjoten mahdollisuuden FIDO2-tunnistautumiselle. Tällä hetkellä tuettuja selaimia/järjestelmiä ovat muun muassa Google Chrome, Mozilla Firefox, Microsoft Edge, Apple Safari, Win10 ja Android (Fido Alliance 2023b). Fido Alliancen (2018) FIDO2-spesifikaation mukaan WebAuthn-rajapinta mahdollistaa tunnistettavien julkisten ja yksityisten avaimien luomisen ja hakemisen jokaisessa palvelussa, jossa se on otettu käyttöön. Se toimii valmiina rajapintana, jonka avulla web-palvelut voivat helposti päivittää kirjautumissivunsa FIDO2-pohjaiseksi. Rajapinta toimii myös tiedonvälittäjänä käytettäessä CTAP2 protokollaa, joka mahdollistaa tunnistautumisen käyttämällä ulkoisiin fyysisiin avaimiin liittyviä teknologioita kuten USB, NFC tai BLE (Fido Alliance, 2018).

2.6.2 Standardin toimintaperiaate

FIDO2-standardiin perustuvien kirjautumisen käyttöönoton yhteydessä käyttäjän laite luo web-palvelu- ja käyttäjäkohtaisen uniikin ja asymmetrisen kryptografisen avainparin, josta julkinen osa (eng. *public key*) jää palveluntarjoajalle ja yksityinen osa (eng. *private key*) jää käyttäjälle (Fido Alliance, 2021). Avaimet ovat matemaattisesti sidoksissa toisiinsa ja WebAuthn rajapinta luo sekä suojaa avaimet automaattisesti (W3C, 2021). Näin ollen avaimesta tulee aina laatumittareilla mitattuna tarpeeksi vahva, eikä käyttäjä saa avainta omaan tietoonsa (Fido Alliance, 2020).

Kun käyttäjä kirjautuu palveluun, tarjoaa verkkosivu tunnistautumista suoraan kirjautumisen yhteydessä, mikäli käyttäjä on rekisteröitynyt palveluun laitteellaan ja ottanut käyttöön FIDO2-tunnistautumisen (W3C, 2020). Yksi Fido Alliancen (2021) tietoturvalupauksista liittyykin siihen, että FIDO2-tunnistautumista tarjotaan vain oikeaksi todennetuilla verkkosivuilla. Näin ollen Fido Alliancen mukaan käyttäjä ei pääse kirjautumaan esimerkiksi huijaussivustoille automaattisen sivustojen tunnistamisen vuoksi (Fido Alliance, 2021) Kirjautumisen yhteydessä palveluntarjoajan palvelin lähettää käyttäjälle niin sanotun kirjautumishaasteen, jonka voi ratkaista ja allekirjoittaa vain käyttäjän hallusta löytyvällä yksityisellä avaimella (Fido Alliance, 2020). Luotetulta osapuolelta, eli web-palveluntarjoajan palvelimelta saatu haaste on muodoltaan korkean entrypologian kokonaisluvuista muodostuva merkkijono (Kabir & Elmedany, 2022). Kirjautumishaaste sisältää myös tiedon palveluntarjoajasta (W3C, 2020).

Kun käyttäjän laite on vastaanottanut kirjautumishaasteen, kysyy laite lupaa sen ratkaisemiseksi hyödyntäen käyttäjän laitteella sijaitsevaa yksityistä avainta (Fido Alliance, 2020). Käyttäjä hyväksyy ratkaisemisen tunnistautumalla esimerkiksi biometristen tietojensa avulla älylaitteella tai ulkoisella fyysisellä todentajalla kuten usb-laitteella (Fido Alliance, 2018). Biometriset tiedot eivät Fido Alliancen lupausten mukaan koskaan poistu käyttäjän laitteelta (Fido Alliance, 2020). Kun haaste on ratkaistu käyttäjän laitteella yksityisen avaimen avulla, lähetetään se allekirjoitettuna palveluntarjoajalle julkisella avaimella todennettavaksi ja käyttäjä pääsee kirjautumaan palveluun (Fido Alliance, 2023). Allekirjoitettu haaste palautetaan luotetulle osapuolelle käyttäen RSA tai ECDSA (*Elliptic Curve Digital Signature Algorithm*) protokollaa (Kabir & Elmedany, 2022). Prosessin jälkeen käyttäjä on kirjautunut palveluun, eikä käyttäjän salainen avain ole Fido Alliancen (2021) mukaan poistunut prosessin aikana käyttäjän laitteelta. WebAuthn rajapinta ei ole myöskään saanut privaattia avainta tietoonsa yhdessäkään prosessin vaiheessa (W3C, 2021). Protokollan toimintaprosessista kirjautumisen yhteydessä olen luonut yksinkertaistetun ja havainnollistavan kuvan W3C (2021) tarjoaman materiaalin pohjalta (kuvio 2).



KUVIO 2 FIDO2 kirjautuminen web-palveluissa.

2.6.3 FIDO2 ja tietoturvalupaukset

Fido Alliance (2021) listaa tietoturvadokumentaatiossaan FIDO2-standardeihin liittyviä tietoturvatavoitteita ja varjeltavia elementtejä. Varjeltavilla elementeillä tarkoitetaan tässä kontekstissa tietoturvan kannalta kriittisiä protokollan toiminnan osia. Varjeltavat elementit liittyvät tiivistetysti TLS yhteyden käyttämiseen sekä julkisen- ja yksityisen avaimen suojaamiseen. Tietoturvatavoitteina ja lupauksina FIDO2-kirjautumisella on Fido Alliance (2021) dokumentaation mukaan tutkimukseni kannalta oleelliset asiat tiivistettynä seuraavat asiat:

1. Vahvan kryptografian varmistaminen automaattisen kirjautumisavainten luonnin avulla.

2. Suoja arvaamishyökkäyksille muun muassa säilyttämällä vain avainparin julkista osaa palveluntarjoajan tietovarastossa.
3. Suoja tietojenkalastelulle julkisen ja yksityisen avainparin avulla.
4. Suoja tietovuodoille niin, että mikään, mitä todentaja voisi vuotaa, ei voi auttaa hyökkääjää esiintymään käyttäjänä tai toisena käyttäjänä toiselle luottavalle osapuolelle. Tämä toteutuu muun muassa säilyttämällä vain avainparin julkista osaa palveluntarjoajan tietovarastossa.
5. Palveluntarjoajalle alttiina olevien tunnistetietojen määrän rajaaminen minimiin.
6. Käyttäjien informoiminen, ennen kuin uusi suhde palveluntarjoajaan muodostetaan.
7. Kirjautumislaitteen tietojen varmistaminen lisävarmisteena kirjautumisen yhteydessä.
8. Toimintaympäristön turvallisuusrajojen kunnioittaminen varmistaen, että rekisteröinnit ja yksityisen avaimen materiaali jaettuna järjestelmäresurssina, on asianmukaisesti suojattu FIDO-käyttäjälaitteen käyttöympäristön käyttöoikeusrajojen mukaisesti.

Lisäksi W3C (2021) listaa WebAuthn dokumentaatiossaan rajapinnan hyötyjä palveluntarjoajille. Se esimerkiksi kertoo ratkaisun hyödyiksi seuraavat asiat:

1) Ratkaisu on laajasti yhteensopiva ja helppokäyttöinen.
2) Palvelun ei tarvitse huolehtia käyttäjän käyttämästä laitteesta.
3) Tunnistautuminen on vahvaa väliintulo hyökkäyksiä (eng. *man-in-the-middle attack*) vastaan.
4) Käyttäjä voi käyttää monia eri keinoja tunnistautumiseen laitteella ilman, että palveluntarjoajan koodiin tarvitsee tehdä muutoksia.
5) Palveluntarjoajan ei tarvitse tallentaa salaisuuksia tietovarastoon saavuttaakseen edellä mainitut edut (1–4).

W3C kertoo dokumentaatiossaan kuitenkin, että palveluntarjoajan vastuulla on huolehtia kirjautumishaasteiden riittävästä pituudesta, joka on suosituksen mukaan minimissään 16-bittiä, sekä niiden satunnaisgeneroinnista (W3C, 2021). Näin ollen FIDO2-kirjautuminen edellyttää palveluntarjoajalta joitakin kryptografisia toimenpiteitä, eikä se havaintojeni mukaan ole välttämättä täysin huoleton ratkaisu implementoida.

3 SALASANAPOHJAISEN TUNNISTAUTUMISEN TIETOTURVAHAASTEET WEB-PALVELUISSA

Salasanojen tietoturva on ollut ongelma jo vuosikymmenien ajan (Bauman & Lin, 2015; Bonneau ym. 2015). Yleisimmät salasanojen tietoturvaongelmat liittyvät salasanojen varastamiseen käyttäjältä esimerkiksi tietojenkalastelun kautta. Tietovarastointiin liittyvät riskit liittyvät tunnettuihin haavoittuvuuksiin, joita hyödyntämällä rikollinen voi päästä käsiksi salasanojen tietovarastoon (Bauman & Lin, 2015). Myös salasanojen vanhanaikainen tai vanhentunut suojaaminen tietovarastossa altistaa käyttäjän tiedot tietomurron yhteydessä (Ntantogian, Malliaros & Xenakis, 2019). Tässä luvussa perehdytään salasanapohjaisen tunnistautumisen ongelmiin ja tietoturvahaasteisiin web-palveluissa erityisesti toteutuksen näkökulmaa painottaen.

3.1 Toteutuksen ongelmat

Baumanin ja Lin (2015) mukaan salasanojen tietovarastoinnin toteutus on monimutkaista ja siinä voi mennä moni asia väärin. Tutkijat kertovat tutkimuksessaan, että vaikka salasanojen suojaamisen ja hajauttamisen tärkeys pitäisi olla nykyaikana jo selvää, osoittavat monet tapahtuneet tietomurrot todeksi sen, että salasanoja edelleen säilytetään ja välitetään tekstimuotoisena tietovarastoissa (Bauman & Li, 2015). Tekstimuotoinen salasanojen säilyttäminen riitelee vahvasti luvussa kaksi käsiteltyjen salasanapohjaisen kirjautumisen toteutukseen liittyvien parhaiden käytänteiden kanssa, sillä kaikki kolmesta käytänteitä tarjoavista tahoista (NIST, OWASP ja Google Cloud) edellyttivät salasanojen jopa hieman monimutkaista suojaamista hajautusalgoritmeilla, ennen tietovarastoon tallentamista.

Naiakshinan ym. (2017) mukaan ohjelmistokehittäjien heikko kryptografisten rajapintojen tuntemus on johtanut siihen, että he luovat nykystandardien mukaan heikkoja salasanojen varastointiratkaisuja. Tähän suurimpana syynä on heidän mukaansa rajapintojen valmiina tarjoamien salasanojen suojaukseen liittyvien oletusasetusten heikko taso (Naiakshina ym., 2017). Saman asian ovat

huomanneet selvästi muutkin tutkijat. Esimerkiksi Ntantogian, Malliaros ja Xenakis (2019) kertovat artikkelissaan WordPressin käyttävän edelleen NIST:in ja OWASP:in vanhanaikaiseksi ja riittämättömäksi luokittelemaa MD5-suojausta vain pienellä määrällä iteraatioita. Heidän mukaansa ongelma on ratkaistavissa WordPressiin asennettavan laajennuksen kautta. Tämä kuitenkin heidän mukaansa edellyttää, että ohjelmistokehittäjä on tietoinen ongelmasta ja osaa implementoida vaihtoehtoisen ratkaisun. Tämän he kertovat olevan ristiriidassa WordPressin markkinoiman ratkaisuvalmiin filosofian kanssa, jolla tarkoitetaan sitä, että kehittäjien ei juurikaan tarvitsisi WordPressin mukaan joutua huolehtimaan asiasta (Ntantogian, Malliaros & Xenakis, 2019). Wurster ja van Oorschot (2008) kiteyttävät ohjelmistokehityksen nykypäivänä olevan enemmän erilaisten kirjastojen yhteen liimaamista, kuin monimutkaisten kokonaisuuksien kokonaisvaltaista toteuttamista. Heidän mukaansa monimutkaisuus on tietoturvan vihollinen ja juuri rajapintojen monimutkaisuus saa ohjelmistokehittäjät tekemään virheitä (Wurster ja van Oorschot, 2008). Tätä näkemystä puoltavat myös edellä mainitut Naiakshinan ym. (2017) löydökset.

Naiakshina ym. (2019) mukaan salasanojen tietovarastoinnin toteutuksen tietoturvan tutkiminen on jäänyt liian vähälle huomiolle tiedeyhteisössä, fokuksen painottuessa tutkimaan käyttäjien ongelmia. Heidän artikkelinsa mukaan useat tutkimukset osoittavat kehittäjien toteuttavan myös huomattavan usein liian heikkoja salasanojen tietovarastointiratkaisuja (Nakaishina ym., 2019). Myös Bonneau ym. (2012) tunnistivat salasanapohjaisia menetelmiä arvioivassa artikkelissaan ongelman siitä, että jo pelkästään huono salasanojen suojauksen implementointi voi tehdä tyhjäksi kirjautumismenetelmän muut hyödyt. Toisaalta he toteavat saman tapahtuvan tilanteissa, joissa käyttäjä luo esimerkiksi huonon tai useita kertoja käytetyn salasanan. Tällä he tarkoittavat ymmärrettävästi sitä, että palveluntarjoajan huolellisuus salasanojen käsittelyssä voi vesittyä, jos salasana on yksinkertaisesti helposti arvattavissa (Bonneau ym., 2012). Green & Smith (2016) ilmaisevat artikkelissaan huolensa siitä, että tutkimukset keskittyvät pääosin loppukäyttäjien ongelmiin, vaikka ohjelmistokehittäjien tekemät virheet voivat johtaa kokonaisten tietokantojen ja näin ollen käyttäjämassojen tietoturvan vaarantumiseen.

Vaikka ohjelmistokehittäjien vastuu on selkeästi hyvin suuri salasanojen suojauksen osalta, Naiakshinan ym. (2017) mukaan ohjelmistokehittäjillä on suuria vaikeuksia toteuttaa tietoturvallista tietovarastointia kirjautumistiedoille, vaikka rajapinnat tarjoavat täysin käyttövalmiita ratkaisuja kirjautumistietojen salaukseen. Bauman, Lu ja Li (2015) tunnistivat ongelman siitä, että salasanojen varastointiin ei löydy kunnollisia standardeja, jotka voisivat edesauttaa ohjelmistokehittäjien tietoturvallisia implementaatioratkaisuja. Bonneau ja Preibusch (2013) tekivät laajan tutkimuksen web-palveluiden salasanojen keräämis-, käyttö ja resetointimekanismeista. Tutkimuksen perusteella tutkijat totesivat web-palveluiden toteuttavan mekanismeja yhtä huonosti kuin käyttäjät toteuttivat hyviä salasananhallinta ohjeistuksia (Bonneau ja Preibusch, 2013). Näiden tieteellisten tutkimusten perusteella ei ole yhtään yksiselitteistä todeta, että käyttäjät olisivat pääsyyllisiä salasanapohjaisen kirjautumisen ongelmiin.

3.2 Salasanojen luomisen ongelmat

Keith, Shao & Steinbart (2009) kertovat, että salasanojen muistamisen problematiikka ajaa käyttäjät uudelleenkäyttämään salasanoja tai käyttämään liian helppoja salasanoja. Heidän mukaansa salasanojen uudelleenkäyttäminen vaarantaa käyttäjän tiedot useissa palveluissa, vaikka murto olisi tapahtunut vain yhteen palveluun (Keith, Shao & Steinbart, 2009). Florencion ja Herleyn (2007) tekemän tutkimuksen mukaan tavallisella käyttäjällä on käytössään noin seitsemän salasanaa ja yhtä salasanaa käytetään heidän mukaansa keskimäärin neljällä sivustolla. Heikot salasanat altistavat käyttäjätilit myös arvaamishyökkäyksille, joissa hyökkääjät yrittävät murtaa käyttäjätilejä yleisimmin tiedossa olevilla heikoilla salasanoilla (Ntantogian, Malliaros & Xenakis, 2019). Salasanojen muistamisen ongelmaa on yleisesti pyritty ratkaisemaan salasananhallintaohjelmilla, joissa salasanat tallennetaan suojattuun ohjelmaan niin sanotun pääsalasanan taakse. Salasananhallintaohjelmat yleisesti voidaan katsoa parantavan tietoturvallisuutta, vaikka niiden tunnistettuna ongelmana ovat esimerkiksi automaattisen täytön tietoturvaan liittyvät ongelmat (Silver, Jana, Boneh & Jackson, 2014). Automaattisessa täytössä salasananhallintaohjelma tunnistaa sivuston, jolle on tallennettu salasana ja tunnistamisen jälkeen ohjelma kirjoittaa kirjautumiskenttään automaattisesti sivuston kirjautumistiedot käyttäjää helpottaakseen (OWASP, 2023b).

Vaikka käyttäjän ohjaaminen salasanan luomisen yhteydessä on tunnistettu alalla yhdeksi parhaista käytännöistä, on monilla web-palveluilla aineistoanalyysini perusteella selkeästi ongelmia sen toteuttamisessa. Leen, Sjöbergin ja Narayanan (2022) tekemässä tutkimuksessa käsitellyistä 120 web-palvelusta 71 ei käyttänyt estolistoja, 97 ei käyttänyt voimamittareita ja 54 ei noudattanut koostumussääntöjä. Heidän mukaansa esimerkiksi voimamittareita toteuttaneista 23 sivusta 10 toteutti niitä puutteellisesti, esimerkiksi jättämällä salasanan arvattavuuden kokonaan tutkimatta. Näin ollen tutkimuksen mukaan vain 13/120 sivustoa toteutti voimamittareita ilman puutteita. Tutkijat myös ilmaisevat tutkimuksessa huolensa siitä, että uudet salasanateknologiat kuten kaksivaiheinen tunnistautuminen (MFA, eng. *Multi-factor authentication*) hämärtävät web-palvelujen käsitystä turvallisuudesta. He olivat myös huolissaan siitä, että erilaisilla salasanan luomista ohjaavilla menetelmillä voitaisiin luoda käyttäjälle valheellista turvallisuuden tunnetta (Lee, Sjöberg & Narayana, 2022). Baumanin, Lun ja Lin (2015) näkemyksen mukaan käyttäjien onkin hyvin vaikeaa tai lähes mahdotonta arvioida, käsitteleekö web-palvelu tunnistautumistietoja asianmukaisesti. Tutkimuksessaan he myös havaitsivat huolestuttavasti, että web-palvelut eivät usein noudattaneet parhaita käytäntöjä, vaikka niillä oli asianmukainen tieto ja koulutus tietoturva-asioista. Heidän tutkimuksensa mukaan palveluntarjoajat saattoivat ajatella esimerkiksi, että koska palvelussa ei käsitellä maksutietoja, ei kirjautumistietoja tarvitse lainkaan suojata (Bauman, Lu & Li, 2015).

3.3 Kertakirjautumisen ongelmat

Suurimmat verkkopalvelut ovat yrittäneet luoda lisäturvaa salasanaperusteiselle kirjautumiselle koneoppimismallien ja älyn avulla (Bonneau ym. 2015). Käyttäjälle tämä näkyy niin, että esimerkiksi Microsoftin käyttämä Microsoft Authenticator kirjautumissovellus saattaa lähettää käyttäjän laitteelle varmistuskysymyksen kirjautumisesta, jos kirjautuminen tapahtuu poikkeavassa sijainnissa (Microsoft, 2023). Tämän menettely liittyy vahvasti kertakirjautumisella toteutettuihin palveluihin. Kertakirjautumiseen perustuva tunnistautuminen vähentää palveluntarjoajan vastuuta, koska tietovarastoinnin hoitaa ulkoinen palveluntarjoaja (Kabir & Elmedany, 2022). Kertakirjautuminen parantaa käytettävyyttä sekä poistaa ohjelmistokehittäjiltä kokonaan salasanojen turvalliseen varastointiin, menetelmiin ja menettelytapoihin liittyvät vastuut, koska vastuu tietoturvasta siirtyy todennuksen toteuttavalle ulkoiselle organisaatiolle (Kabir & Elmedany, 2022). Bonneaun ym. (2015) mukaan kertakirjautumisen kaltaisten tunnistautumistapojen ongelmana voidaan pitää käyttäjän tietojen ja tapojen välittymistä ja keskittymistä yhdelle suurelle palveluntarjoajalle. Tutkijoiden mukaan tietojen keskittymisen kustannuksella kuitenkin saadaan salasanojen tunnistamisen rinnalle muita käyttäjän todentamistapoja IP-osoitteen, geolokaation, selaintietojen, evästeiden ja kirjautumisajan avulla (Bonneaun ym., 2015). Esimerkiksi Googlen automaattinen suojaus suojaa käyttäjää kerätyn datan avulla tilanteissa, joissa hyökkääjä tietää käyttäjän kirjautumistunnukset (Moschetto & Maddox, 2019b). Bonneaun ym. (2012) toteuttaman analyysin mukaan kertakirjautumiseen perustuvat tunnistautumismenetelmät tarjoavat enemmän käytettävyyshyötyjä, kuin varsinaisia tietoturvaparannuksia.

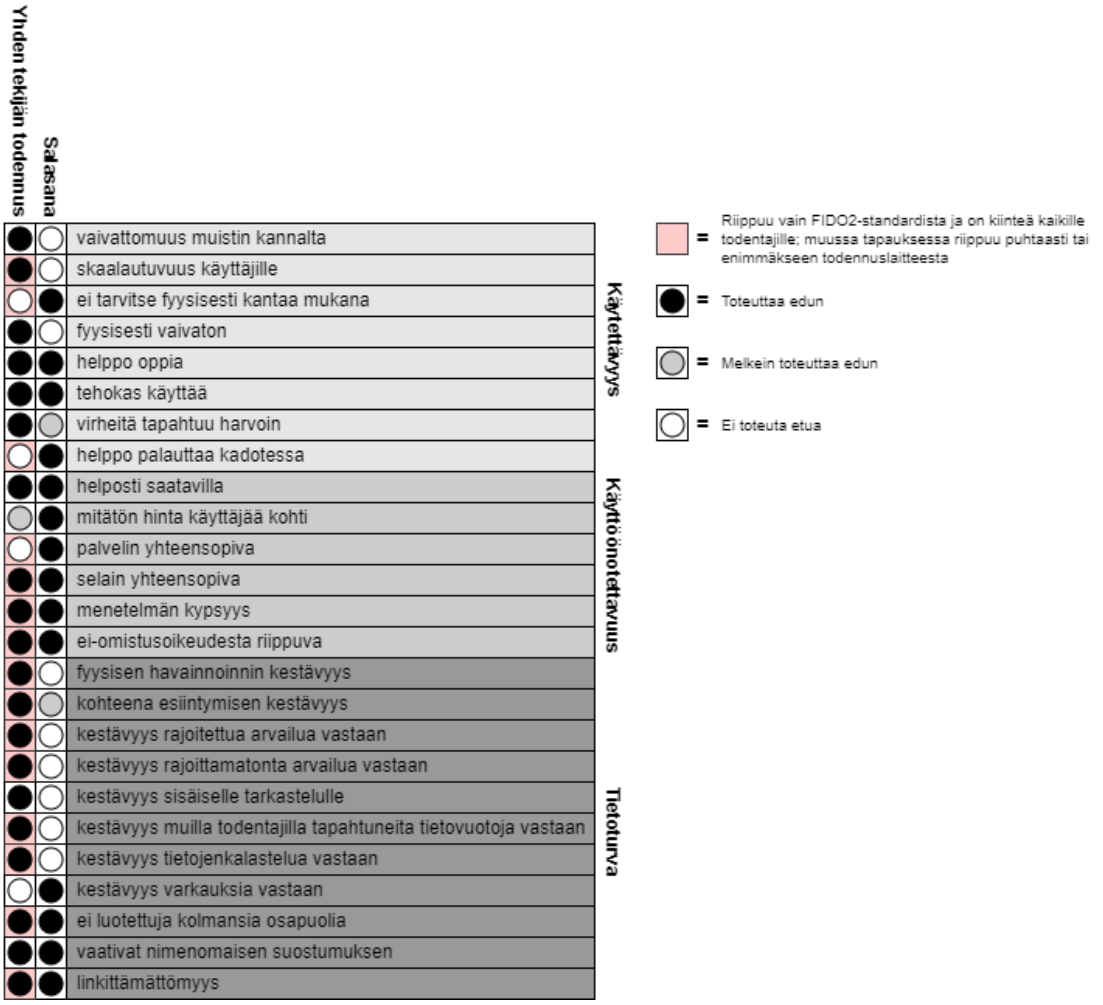
Vaikka kertakirjautumisen tarjoama tunnusten keskittyminen vähentää käyttäjän käytössä olevien salasanojen määrää näyttäisi siinä säilyvän salasanapohjaisen kirjautumisen perustavanlaatuiset ongelmat sekä aineistoanalyysin pohjalta tunnistetut ongelmat liittyen käyttäjän yksityisyydensuojaan.

4 FIDO2 SUHTEESSA SALASANPOHJAISEN KIRJAUTUMISEN ONGELMIIN

Tässä luvussa peilataan FIDO2-standardin kehittäjän lupauksia ja tieteellisen tutkimukseen löydöksiä suhteessa aiemmassa luvussa esiteltyihin salasanallisten kirjautumismenetelmien tieteessä tunnistettuihin haasteisiin. Lisäksi käsitellään FIDO2-standardiin pohjautuvan tunnistautumisen nykyongelmia ja ehdotuksia jatkotutkimukselle.

4.1 Hyödyt suhteessa salasanoihin

FIDO2-standardia on pidetty selkeimpänä salasanapohjaisen kirjautumisen uutena ja potentiaalisena haastajana. (Lyastani ym., 2020; Keil, Markert & Durmuth, 2022). Lyastanin ym. (2020) tutkimuksen mukaan käyttäjät pitivät FIDO2- tunnistautumista fyysisen Yubicon-avaimen kanssa käytettävänä ratkaisuna. Tutkijat suorittivat artikkelissaan Bonneaun ym. (2012) esittämän arviointimenetelmän pohjalta tehdyn arvioinnin FIDO2-tunnistautumismenetelmälle käytettäessä fyysistä USB-avainta nimeltä Yubico-avain (Kuvio 3). He totesivat USB-muotoisen fyysisen avaimen valinnan vaikuttavan arviointiin, joten lopputulos esimerkiksi älypuhelinta käytettäessä voi erota heidän tarjoamastaan analyysista. Tutkijoiden lopputulos oli kuitenkin se, että yksikään muu vaihtoehtoinen tunnistautumismenetelmä ei saanut Bonneaun ym. (2012) arviointimenetelmän pohjalta yhtä vahvaa tulosta kuin FIDO2-tunnistautuminen Yubicon-avaimen kanssa (Lyastani ym., 2020).



KUVIO 3 Salasana- ja FIDO2-pohjaisen (yhden tekijän todennus, *eng. 1FA*) - kirjautumisen välinen vertailu Bonneau ym. (2012) arviointimenetelmän pohjalta. (Lyastani ym., 2020, s.270).

FIDO2-tunnistautumisen vahvuudet suhteessa salasanoihin liittyvät aineistoanalyysini pohjalta erityisesti tietoturvaan ja käytettävyyteen. Lyastani ym. (2020) määrittelivät FIDO2:een perustuvan yhden tekijän todennuksen olevan perinteistä salasanapohjaista tunnistautumista parempi seuraavilla tietoturvaan liittyvillä osa-alueilla:

1) Suoja fyysiselle tarkkailulle.
2) Suoja toisena henkilönä esiintymiselle.
3) Suoja hyökkääjiltä, joiden salaisuuksien arvausnopeus on rajoitettu.
4) Suoja hyökkääjiltä, joiden salaisuuksien arvausnopeus ei ole rajoitettu.
5) Suoja sisäiselle tarkastelulle.
6) Suoja ulkopuolisille tietovuodoille.
7) Suoja tietojenkalastelulle.

Myös esimerkiksi Kabirin ja Elmedanyn (2022) tunnistivat FIDO2:n antavan suojan tietojenkalastelulta ja käyttäjien manipulaatiolta. Fido Alliance (2023) lupaa

standardinmukaisen kirjautumisen myös tunnistavan sivustot, joille käyttäjä kirjautuu varmistaen, että käyttäjä ei pääse kirjautumaan tekaistuille nettisivuille. Tämä lupaus liittyy Kabirin ja Elmedanyn (2023) tunnistamaan käyttäjän manipulaation eliminoimiseen.

Tutkimusten mukaan FIDO pohjaisella tunnistautumisella on salasanapohjaista tunnistautumista pienempi hyökkäyspinta, koska salasanapohjaiseen tunnistautumiseen perustuvissa menetelmissä on enemmän toisiinsa liittyviä ja riippumattomia elementtejä (Alqubaisi, Wazan, Ahmad ja Chadwick, 2020). Alqubaisi ym. (2020) luonnehtivatkin FIDO2-pohjaista tunnistautumista puolisuljetuksi järjestelmäksi. Kabirin ja Elmedanyn (2022) mukaan julkiseen ja yksityiseen avaimeen perustuva FIDO2-tunnistautuminen poistaa mahdollisuuden tietojen hyödyntämiselle tietomurtojen avulla sekä väsytyshyökkäysten ja Pass-the-Hash hyökkäysten hyödyntämisen. Myös Bicakci & Uzunay (2022) tunnistavat FIDO2-standardin mukaisen tunnistautumisen eliminoivan mahdollisuuden väsytyshyökkäyksille ja arvaamishyökkäyksille tietojenkalastelun lisäksi. He tunnistavat FIDO2:n myös vähentävän kustannuksia yrityksille salasanojen palauttamiseen liittyvien teknisen asiakaspalvelun yhteydenottojen pois jäännin myötä (Bicakci & Uzunay, 2022).

Käytettävyyden näkökulmasta esimerkiksi Apple (2023) kertoo kehittäjädokumentaatiossaan, että sen vuonna 2022 julkaisemassa FIDO2-standardiin pohjautuvassa ratkaisussa yksityiset avaimet luodaan automaattisesti ja ne tallennetaan käyttäjän laitteelle Apple Keychain:iin ilman, että käyttäjän tarvitsee tietää tai muistaa yksityistä kirjautumisavainta. Keychain:in avulla FIDO2-tunnistautuminen on mahdollista kaikilla käyttäjän yksityisillä laitteilla, eikä ulkoinen palveluntarjoaja saa yksityistä avainta tietovarastoonsa säilytettäväksi (Applen, 2023). WebAuthn rajapinnan kautta avaimet luodaan automaattisesti, ilman käyttäjän kontribuutiota (W3C, 2021). FIDO2-ratkaisussa noudatetaan siis Bonneaun ym. (2012) määrittelemää muistamisen kannalta vaivatonta ratkaisua, jonka myötä useat tietoturvamääritykset myös analyysini perusteella toteutuvat, koska käyttäjällä ei ole yksityistä avainta muistinvaraisesti saatavilla.

Koska käyttäjä ei FIDO2-ratkaisussa luo itse tunnistautumisavaimia, ei käyttäjää tarvitse oletettavasti myöskään ohjata salaisuuksien luonnissa. Kehittääkseen turvallisia salasanapohjaisia tunnistautumisratkaisuja on ohjelmistokehittäjien huolehdittava monista aiemmin tässä tutkimuksessa esitellyistä hyviin käytänteisiin liittyvistä osa-alueista, kuten käyttäjän salasanan luomisen ohjaamisesta, salasanojen salaamisesta ja turvallisesta tietovarastoinnista käyttäen kryptografisia rajapintoja, joiden asianmukainen tuntemus on aineistoanalyysissä käytettyjen tutkimusten mukaan heikkoa. Fido Alliancen FIDO2-tunnistautumismenetelmän yksi päälupauksista on se, että käyttäjän tietoturvakriittisiä tunnistetietoja ei säilytetä palveluntarjoajien servereillä (Fido Alliance, 2023). Alqubaisi, Wazan, Ahmad ja Chadwick (2020) tunnistivat tutkimuksissaan tämän yhdeksi FIDO tunnistautumisen selkeistä hyödyistä suhteessa salasanapohjaiseen tunnistautumiseen. Myös Würsching, Putz, Haesler ja Hollick (2023) tunnistavan FIDO2-tunnistautumisen poistavan web-palveluilta salaisuuksien tallentamisen vastuun ja näin ollen parantavan palveluntarjoajan asemaa verrattuna

mahdollisiin tietovuotoihin. He myös kertovat FIDO2:n olevan selaimen kautta toteutetun WebAuthn rajapinnan kautta helppo toteuttaa teknisesti (Haesler & Hollick, 2023). FIDO2-standardiin perustuvassa ratkaisuissa kuitenkin jotkin aikaisemmin tässä tutkimuksessa esitellyt osa-alueet jäävät palveluntarjoajan huolehdittavaksi. Tällainen vastuu on W3C:n (2023) mukaan esimerkiksi kirjautumishaasteiden muodostaminen ja käsittely. W3C:n (2023) dokumentaation analyysin perusteella voidaankin todeta, että ratkaisu ei välttämättä ole vielä täysin ratkaisuvalmis, vaan se edellyttää edelleen töitä ja erityisosaamista ohjelmistokehittäjiltä.

4.2 FIDO2 standardin haasteet ja kehityskohdat

Vaikka FIDO2 näyttää saaneen hyvän vastaanoton tiedeyhteisössä, on sitä myös tutkimuksissa arvosteltu sen 1) verrattain kalliista implementoinnista, 2) rajallisista tunnusten palautustavoista, 3) biometriikkaan liittyvistä heikkouksista ja 4) fyysisten avainten heikkouksista, kuten kirjautumislaitteen varkaudesta, rikkoutumisesta tai vahingoittumisesta (Kabir & Elmendany, 2022). Näitä havaintoja tukee myös Lyastanin ym. (2020) tekemä tutkimus. Vaikka tunnistautumiseen käytettävän laitteen häviäminen koettiin useissa tutkimuksissa yhdeksi FIDO2-tunnistautumisen heikkouksista, tuovat Alqubaisi, Wazan, Ahmad ja Chadwick (2020) artikkelissaan esiin näkökulman siitä, että laitteen häviämisen käyttäjän on mahdollista huomata kohtuullisen ajan sisällä. Tällä tutkijat ymmärtääkseni tarkoittavat sitä, että salasanapohjaisessa tunnistautumisessa sama ei ole käyttäjälle mahdollista, sillä mahdollinen tietomurto tapahtuu web-palvelun palvelimilla käyttäjältä piilossa (Wazan, Ahmad & Chadwick, 2020).

Keilin Markertin ja Dürmuthin (2022) tutkimuslöydösten mukaan FIDO2-pohjainen tunnistautuminen koettiin käyttäjien keskuudessa hieman vaikeasti käyttöön otettavalta. Bicakcin ja Uzunayn (2022) mukaan muita FIDO2:n käytettävyyteen liittyviä heikkouksia ovat jaettujen käyttäjätilien käytön vaikeudet. He myös kyseenalaistivat artikkelissaan FIDO2-kirjautumisen käytettävyyden järkevyyden suhteessa salasanahallintaohjelmiin. Tutkijat tuovat artikkelissaan esiin myös ongelman vaihdettaessa fyysistä laitetta niin, että laitteen merkki vaihtuu esimerkiksi Applesta Googleen. Tällöin ongelmia FIDO2-avaimien siirtämisessä heidän mukaansa ilmenee, sillä molemmat osapuolet tarjoavat omat erilliset FIDO2-pohjaiset ratkaisunsa (Bicakci & Uzunay, 2022). Tämä voi mielestäni muodostua ongelmaksi myös ohjelmistokehittäjille, mikäli eri FIDO2-ratkaisut eivät esimerkiksi keskustele keskenään. Lisätutkimus tästä olisi varmasti tarpeen eri FIDO2-toteutusten julkaisujen myötä.

Jingjingin, Lin, Yen ja Zhaon (2022) tekemän laajan tietoturvatutkimuksen mukaan aiemmin löydettyjä UAF (*Universal Authentication Framework*) heikkouksia kuten rebinding-hyökkäys ja rinnakkaisistuntohyökkäykset on edelleen olemassa FIDO2-standardissa. He myös havaitsivat CTAP2 protokollassa olevan mahdollisuus väliintulohyökkäyksille (Jingjing, Li, Ye & Zhao, 2022). Vuotta aiemmin myös Barbosa, Boldyreva, Chen & Warinschi (2021) löysivät

tutkimuksessaan heikkouksia CTAP2 protokollasta ja ehdottivat siihen parannuksia, jotka antaisivat paremman suojan väliintulohyökkäyksille. Fido Alliancen (2021) tietoturvadokumentaation mukaan väliintulohyökkäyksiä vastaan olisi kehitetty useita eri toimenpiteitä, mutta edellä mainittujen tiedeyhteisön löydösten perusteella standardin kehittäjän ja tiedeyhteisön tulosten välillä on havaittavissa ristiriitaa.

Koska tiedeyhteisö on tunnistanut salasanapohjaisen kirjautumisen toteutuksessa monia puutteita, herää FIDO2:n kohdalla kysymys siitä, onko menneestä opittu ja toteutusratkaisuista tehty parempia ja turvallisempia. Jingjingin, Lin, Yen ja Zhaon (2022) kertoivat löytäneensä FIDO2:n määritelmien olevan vielä puutteellisia turvallisten yhteyksien ja tietovaraston suhteen. Alam, Krombholz ja Bugiel (2019) tekivät tutkimuksen WebAuthn käyttöönotosta ja esittivät huolensa siitä, että ohjelmistokehittäjät olivat tutkimuksen perusteella toteuttaneet huonoja FIDO2-ratkaisuja puutteellisen dokumentaation ja koulutusratkaisuiden myötä. Tämä oli havaittavissa esimerkiksi tutkijoiden tutkimassa kymmenessä suosituimmassa WebAuthn-kirjastossa GitHub-palvelussa. Tutkijat esittivät huolensa siitä, että heidän tutkimuksensa mukaan paras dokumentaatio löytyi standardin kehittäjän kirjaston sijaa ulkopuolisien asiantuntijabloggaajien sivustoilta (Alam, Krombholz & Bugiel, 2019). Tekemäni tutkimuksen pohjalta ymmärrän tutkijoiden esittämää huolta, sillä esimerkiksi tutkimukseni luvussa yksi esitelty monimutkainen standardien välinen riippuvuus ei ollut kovin selkeänä suoraan esillä Fido Alliancen omassa dokumentaatiossa.

FIDO2-tunnistautuminen käyttää salasanapohjaisen kirjautumisen tapaan myös rajapintaa (WebAuthn) ratkaisun implementoinnissa. Kabirin ja Elmadany (2022) arvioivat artikkelissaan siirtymän käyttämään FIDO2-pohjaista tunnistautumista olevan hidas, sillä ohjelmistokehittäjät ovat hyvin tottuneita käyttämään valmiita salasanapohjaisia ratkaisuja. Toisaalta Naiakshina ym. (2017), sekä OWASP (2023a) mukaan rajapintojen käyttö on salasanapohjaisissa menetelmissä hyvin normaali käytäntö. Tämä voisi osaltaan helpottaa ohjelmistokehittäjien työtä FIDO2-ratkaisun käyttäessä hyvin saman tapaisia rajapintamenetelmiä toteutuksessa, kuin mitä ohjelmistokehittäjät ovat jo tutkimusten mukaan tottuneet käyttämään.

4.3 Pohdinta ja ehdotukset jatkotutkimukselle

Salasanattoman- ja salasanapohjaisen kirjautumisen tieteellisissä tutkimuksissa näyttää kirjallisuuskatsaukseni perusteella olevan yhteinen ongelma ohjelmistokehittäjien puutteellisessa koulutuksessa ja riittämättömässä tukidokumentaatiossa suhteessa kirjautumismenetelmien toteutukseen käytettäviin rajapintoihin. FIDO2 näyttäytyy standardin kehittäjän tarjoaman kirjallisuuden perusteella olevan toteutusratkaisuna monilla tavoilla saman tapainen toteuttaa, kuin salasanapohjainen kirjautuminen. Tämä liittyy erityisesti siihen, että vaikka FIDO2-ratkaisut käyttävät valmista rajapintaa (WebAuthn) toteutuksessa, on siinä monta

vaihetta, joista ohjelmistokehittäjällä tulisi olla tietotaitoa. Tätä näkemystä tukee se, että ohjelmistokehittäjien vastuulle jää W3C:n (2021) dokumentaation perusteella vielä työvaiheita, joita rajapinta ei automaattisesti hoida. WebAuthn rajapinnan tietoturvaa ja tietoturvallista käyttämistä tiedeyhteisön tulisi tulevaisuudessa tutkia lisää. Lisäksi tulevaisuuden haasteena voidaan kirjallisuuskatsaukseni perusteella nähdä se, että kertakirjautumismenetelmien tapaan FIDO2-standardi on teknologiajättien kehittämää. Jatkotutkimusta tulisikin tehdä teknologiajättien osuudesta FIDO2-standardin kehityksessä ja implementoinneissa tarkastellen käyttäjien datan yksityisyyttä.

5 YHTEENVETO

Salasanapohjainen kirjautuminen on hallinnut web-tunnistautumista lukuisista tunnistetuista tietoturvaongelmistaan huolimatta. Etenkin tietojenkalastelu, tietovuodot ja salasanojen laatuongelmat ovat yleisesti havaittuja salasanojen heikkouksia, joihin eri tahot ovat yrittäneet löytää parempia vaihtoehtoisia ratkaisuja. Vaihtoehtoisia kirjautumismuotoja onkin keksitty monia, mutta tutkimusten mukaan niistä ei vielä ole haastamaan salasanoja yleisimpänä tunnistautumismenetelmänä (Bonneau ym., 2012). Salasanaan pohjautuvien tunnistautumismenetelmien vahvimpana trendinä on nähty kertakirjautuminen (SSO, eng. *Single-Sign-On*), jonka perustavanlaatuisena ongelmana on käyttäjien yksityisen datan välittyminen teknologiajäteille (Bonneau ym., 2012). Toisaalta uusi tunnistautumisstandardi FIDO2 on tunnistettu potentiaalisena salasanattoman tunnistautumisen vaihtoehtona. Standardia kehittää on Fido Alliance, joka koostuu teknologiajäteistä kuten Google, Apple, W3C ja Microsoft. Standardin leviäminen kuluttajakäyttöön näyttää alkaneen, sillä teknologiajäteistä ensimmäisenä Apple julkaisi oman FIDO2-standardiin perustuvan Passkeys teknologiansa kesällä 2022. Koska FIDO2-standardi on suhteellisen uusi ja lupaava standardi, on selvää, että siitä tarvitaan vielä paljon tieteellistä tutkimusta.

Kirjallisuuskatsaukseni tarkoituksena oli selvittää salasanapohjaisen tunnistautumisen parhaita käytäntöjä ja erityisesti sen toteuttamiseen liittyviä ongelmia. Toisena tavoitteena oli tutkia FIDO2:n tarjoamia parannuksia havaittuihin löydöksiin. Käyttäjien tekemät virheet jätettiin pienemmälle huomiolle, sillä tutkimuksen tarkoitus oli keskittyä menetelmien toteuttamiseen. Tutkimus aloitettiin valitsemalla hakusanat tutkimuskysymysten pohjalta sekä keräämällä aineisto. Tämän jälkeen aineistosta rajattiin pois materiaali, joka ei täyttänyt määriteltyjä laatustandardeja, joita olivat muun muassa aineiston tuoreuteen, vertaisarviointiin ja viittausten määrään liittyvät rajaukset. Aineisto analysoitiin ja siitä luotiin taulukkomuotoinen matriisi laadun varmistamiseksi. Tämän jälkeen aineiston pohjalta suoritettiin analyysi. Tutkimuksen haasteina olivat suhteellisen uusi ja vähäinen FIDO2-standardin tutkimus, sekä vähäinen tieteellinen tutkimus salasanapohjaisten kirjautumismenetelmien parhaista käytänteistä.

Aineistoanalyysin pohjalta voidaan tehdä johtopäätelmä, että salasanoihin perustuvan tunnistautumisen toteutuksessa on ohjelmistokehittäjistä johtuvia ongelmakohtia, kuten käyttäjien salasanan luonnin huono ohjaaminen ja tietovaraston puutteellinen toteuttaminen. Ongelmat ovat luonteeltaan sellaisia, että ne poistuisivat, mikäli salasanoja ei tarvittaisi tunnistautumisessa. Alalla tunnistettujen parhaiden käytäntöjen mukaisesti ohjelmistokehittäjien tulee ymmärtää monimutkaisia ja laajoja toteutuskokonaisuuksia tehdessään salasanojen tietovarastointia. Nämä toteutukset tehdään tyypillisesti rajapinnoilla, joiden asianmukaisessa käytössä on tutkimusten mukaan ilmennyt merkittäviä puutteita (Naiakshina ym. 2019). Nämä puutteet voivat johtaa käyttäjien kirjautumistietojen vaarantumiseen mahdollisten tietomurtojen yhteydessä. Tieteellinen tutkimus tunnistaa FIDO2-standardin toimintaperiaatteeseen tukeutuvan lupauksen siitä, että tietojen varastointi käyttäjän hallussa olevaan kirjautumislaitteeseen parantaa käyttäjien tietoturvaa eliminoimalla web-palveluntarjoajien tietovarastoihin kohdistuvien tietomurtojen merkityksen (Kabir & Elmedany, 2022; Alqubaisi ym., 2020; Würsching ym., 2023). FIDO2-pohjaisen kirjautumisen toteutuksessa on kuitenkin havaittu samoja rajapintojen käyttöön liittyviä ongelmia, kuin salasanapohjaisien tunnistautumismenetelmien toteutuksissa (Alam, Krombholz ja Bugiel, 2019). Tiedeyhteisössä FIDO2:n kuitenkin tunnistetaan antavan merkittäviä parannuksia myös tietojenkalastelua vastaan, sillä käyttäjä ei saa omaan tietoonsa kirjautumisen yhteydessä käytettyjä yksityisiä avaimia (Bicakci & Uzunay, 2022). Tieteellisessä yhteisössä laajasti tunnistetun ja käytetyn Bonneaun ym. (2012) tekemän kirjautumismenetelmien arviointimenetelmän perusteella FIDO2-standardiin perustuva tunnistautuminen saa merkittävästi paremman tuloksen, kuin salasanapohjaiset kirjautumismenetelmät (Lyastani ym., 2020). FIDO2:n voidaan aineistoanalyysin perusteella päätellä parantavan käyttäjien tietoturvaa. Sen käyttöönotto kuitenkin vaatii kehittäjiltä aineiston mukaan edelleen korkeaa asiantuntemusta ratkaisun toteuttamisesta salasanapohjaisen toteuttamisen tapaan.

FIDO2:n käyttöönoton toteutuksista löytyy vielä melko suppeasti tieteellistä tutkimusta. Tähän on todennäköisesti syynä teknologian tuoreus sekä mielikuva siitä, että salasanapohjaisessa tunnistautumisessa uusimpana trendinä käytetyn kertakirjautumisen tavoin, FIDO2-standardin ajatellaan poistavan kirjautumistietojen säilyttämisen vastuuta palveluntarjoajalta. Koska FIDO2-toteutukset ovat pääosin teknologiajättien toteuttamia, tulisi lisätutkimusta tehdä siitä, keräävätkö teknologiajätit kertakirjautumisen tapaan käyttäjädataa teknologian avulla. Tutkimukseni aineistoanalyysin pohjalta voidaan todeta, että FIDO2:n tutkimusta vaivaa samat ongelmat kuin salasanapohjaisien kirjautumismenetelmien tutkimusta. Tämä on havainto siitä, että tutkimus keskittyy huomattavasti enemmän käytettävyyden ja käyttäjien ongelmiin, kuin teknologian implementoinnin ongelmiin. Jatkotutkimusta tulisi tehdä esimerkiksi FIDO2:n käyttämän WebAuth:n rajapinnan käytöstä kehittäjäyhteisössä, jotta uutta teknologiaa käyttöönotettaessa kehittäjät eivät lähtisi toistamaan samoja virheitä, jotka tiedeyhteisössä tunnistetaan salasanapohjaisen kirjautumisen toteutuksissa.

LÄHTEET

Apple (2023*) Meet passkeys.* Haettu 21.2.2023 osoitteesta: https://developer.apple.com/videos/play/wwdc2022/10092/

Alam, A., Krombholz, K. & Bugiel, S. (2019). Poster: Let History not Repeat Itself (this Time) -- Tackling WebAuthn Developer Issues Early On. Teoksessa *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security (CCS '19) (2669-2671).* Association for Computing Machinery, New York, USA.

Alqubaisi, F., Wazan, A., S., Ahmad, L. & Chadwick, D., W. (2020). "Should We Rush to Implement Password-less Single Factor FIDO2 based Authentication?," *12th Annual Undergraduate Research Conference on Applied Computing (URC) (1-6)*, Dubai, United Arab Emirates

Barbosa, M., Boldyreva, A., Chen, S. & Warinschi, B. (2021). Provable Security Analysis of FIDO2. *Teoksessa: Malkin, T., Peikert, C. (eds) Advances in Cryptology – CRYPTO 2021. CRYPTO 2021. Lecture Notes in Computer Science(), vol 12827*. Springer, Cham.

Bonneau, J. & Preibusch, S. (2013). *The Password Thicket: technical and market failures in human authentication on the web.* The Ninth Workshop on the Economics of Information Security. Harvard University, USA.

Bonneau, J., Herley, C., van Oorschot, P., C. & Stajano, F. (2012). "The Quest to Replace Passwords: A Framework for Comparative Evaluation of Web Authentication Schemes," *2012 IEEE Symposium on Security and Privacy*, 553-567.

Bonneau, J., Herley, C., van Oorschot, P., C. & Stajano, F. (2015). Passwords and the evolution of imperfect authentication. *Communications of the ACM 58, 7* (78-87).

Bicakci, K. & Uzunay, Y. (2022). Is FIDO2 Passwordless Authentication a Hype or for Real?: A Position Paper. *15th International Conference on Information Security and Cryptography (ISCTURKEY) (68-73).*

Dong, X., Clark, J.A., Jacob, J.L. (2008). Threat Modelling in User Performed Authentication. In: Chen, L., Ryan, M.D., Wang, G. (eds) *Information and Communications Security. ICICS 2008. Lecture Notes in Computer Science, vol 5308. Springer.* Berlin, Heidelberg.

Florencio, D. & Herley, C. (2007) A large-scale study of web password habits. Teoksessa *Proceedings of the 16th international conference on World Wide Web (WWW '07). Association for Computing Machinery (657-666),* New York, USA.

Fido Alliance (2023a) *FIDO Authentication, a passwordless vision.* Haettu 5.2.2023 osoitteesta: https://fidoalliance.org/fido2/

Fido Alliance (2018) *User Authentication Spesifications Overview*. Fido Alliance. Haettu 10.3.2023 osoitteesta: https://fidoalliance.org/specifications/

Fido Alliance (2022) *How FIDO Addresses a Full Range of Use Cases*. Fido Alliance. Haettu 5.2.2023 osoitteesta: https://fidoalliance.org/white-paper-multi-device-fido-credentials/

Fido Alliance (2020*) FIDO2 spesifications* Fido Alliance. Haettu 10.3.2023 osoitteesta: https://fidoalliance.org/specifications/download/

Fido Alliance (2023b) FIDO2:Web Authentication (WebAuthn) Haettu 10.3.2023 osoitteesta: https://fidoalliance.org/fido2-2/fido2-web-authentication-webauthn/

Fido Alliance (2021) FIDO Security Reference. Fido Alliance review draft 25.5.2021. Haettu 13.5.2023 osoitteesta: https://fidoalliance.org/specs/common-specs/fido-security-ref-v2.1-rd-20210525.html

Green, M. & Smith, M., (2016). "Developers are Not the Enemy!: The Need for Usable Security APIs," in *IEEE Security & Privacy*, 14(5), 40-46.

Jinjing, G., Li, H., Ye, H. & Ziming, Z. (2022). A Formal Analysis of the FIDO2 Protocols *27th European Symposium on Research in Computer Security (ESORICS) (3–21).*

Kabir, M., A., Al & Elmedany, W. (2022). "An Overview of the Present and Future of User Authentication," *2022 4th IEEE Middle East and North Africa COMMunications Conference (MENACOMM) (10-17)*, Amman, Jordan.

Keith, M., Shao, B. & Steinbart, P. (2009). "A Behavioral Analysis of Passphrase Design and Effectiveness," *Journal of the Association for Information Systems*, 10(2,3) (63-89).

Keil, M., Markert, P. & Dürmuth, M. (2022). "It's Just a Lot of Prerequisites": A User Perception and Usability Analysis of the German ID Card as a FIDO2 Authenticator. *European Symposium on Usable Security (EuroUSEC '22) (172-188)*, Karlsruhe, Germany. ACM, New York.

Lyastani, S., G., Schilling, M., Neumayr, M., Backes M. & Bugiel, S. (2020). "Is FIDO2 the Kingslayer of User Authentication? A Comparative Usability Study of FIDO2 Passwordless Authentication," *2020 IEEE Symposium on Security and Privacy*, 268-285.

Lee, K., Sjöberg, S. & Narayanan, A. (2022). Password policies of most top websites fail to follow best practices, *Proceedings of the Eighteenth Symposium on Usable Privacy and Security* (561-580), Boston, United States.

Lowe, G. (1997). A hierarchy of authentication specifications, *Proceedings 10th Computer Security Foundations Workshop (31-43)*, Rockport, USA NY, USA, 17 pages.

Microsoft (2023) Using the location condition in a Conditional Access policy. Haettu 30.5.2023 osoitteesta: https://learn.microsoft.com/en-us/azure/active-directory/conditional-access/location-condition

Moschetto, K. & Maddox, I. (2019a). *Best practices for password management, 2019 edition.* Google Cloud. Haettu 1.2.2023 osoitteesta: https://cloud.google.com/blog/products/identity-security/best-practices-for-password-management-2019-edition

Moschetto, K. & Maddox, I. (2019b). *Modern password security for system designers.* Google Cloud. Haettu 1.2.2023 osoitteesta: https://cloud.google.com/static/solutions/modern-password-security-for-system-designers.pdf

Nadi, S., Krüger, S., Mezini, M. & Bodden, E. (2016). ""Jumping Through Hoops": Why do Java Developers Struggle with Cryptography APIs?," *2016 IEEE/ACM 38th International Conference on Software Engineering (ICSE) (935-946)*, Austin, USA

Naiakshina, A., Danilova, A., Tiefenau, C., Herzog, M., Dechand, S. & Smith, M. (2017). Why Do Developers Get Password Storage Wrong? Teoksessa *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security (CCS '17). Association for Computing Machinery*, New York, USA, 311–328.

Naiakshina, A., Danilova, A., Gerlitz, E., von Zezschwitz, E. & Smith, M. (2019). "If you want, I can store the encrypted password": A Password-Storage Field Study with Freelance Developers. Teoksessa *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19). Association for Computing Machinery, Paper 140 (1–12),* New York, USA.

National Institute of Standards and Technology (NIST) (2017a). *Digital Identity Guidelines: Authentication and Lifecycle Management (Special Publication 800-63B).*

National Institute of Standards and Technology (NIST) (2020) Recommendation for Key Management *Special Publication 800-57 Part 1, Revision 5 Natl. Inst. Stand. Technol. Spec. Publ. 800-57 Part 1 Rev. 5* CODEN: NSPUE2

National Institute of Standards and Technology (NIST) (2017b) Digital Indentity Guidlines. *Special Publication 800-63-3 Natl. Inst. Stand. Technol. Spec. Publ. 800-63-3*

Ntantogian, C., Malliaros, S. & Xenakis, C. (2019). Evaluation of password hashing schemes in open source web platforms, *Computers & Security*, Volume 84, 206-224. Greece: University of Piraeus.

Silver, D., Jana, S., Chen, E., Jackson, C. & Boneh, D. (2014). Password Managers: Attacks and Defenses. Teoksessa *Proceedings of the 23rd USENIX Security Symposium (449-464).* CA, San Diego.

The Open Worldwide Application Security Project (OWASP) (2023a). Password Storage Cheat Sheet. Haettu 2.3.2023 osoitteesta: https://cheatsheetseries.owasp.org/cheatsheets/Password_Storage_Cheat_Sheet.html

The Open Worldwide Application Security Project (OWASP) (2023b). Authentication Cheat Sheet. Haettu 15.4.2023 osoitteesta: https://cheatsheetseries.owasp.org/cheatsheets/Authentication_Cheat_Sheet.html

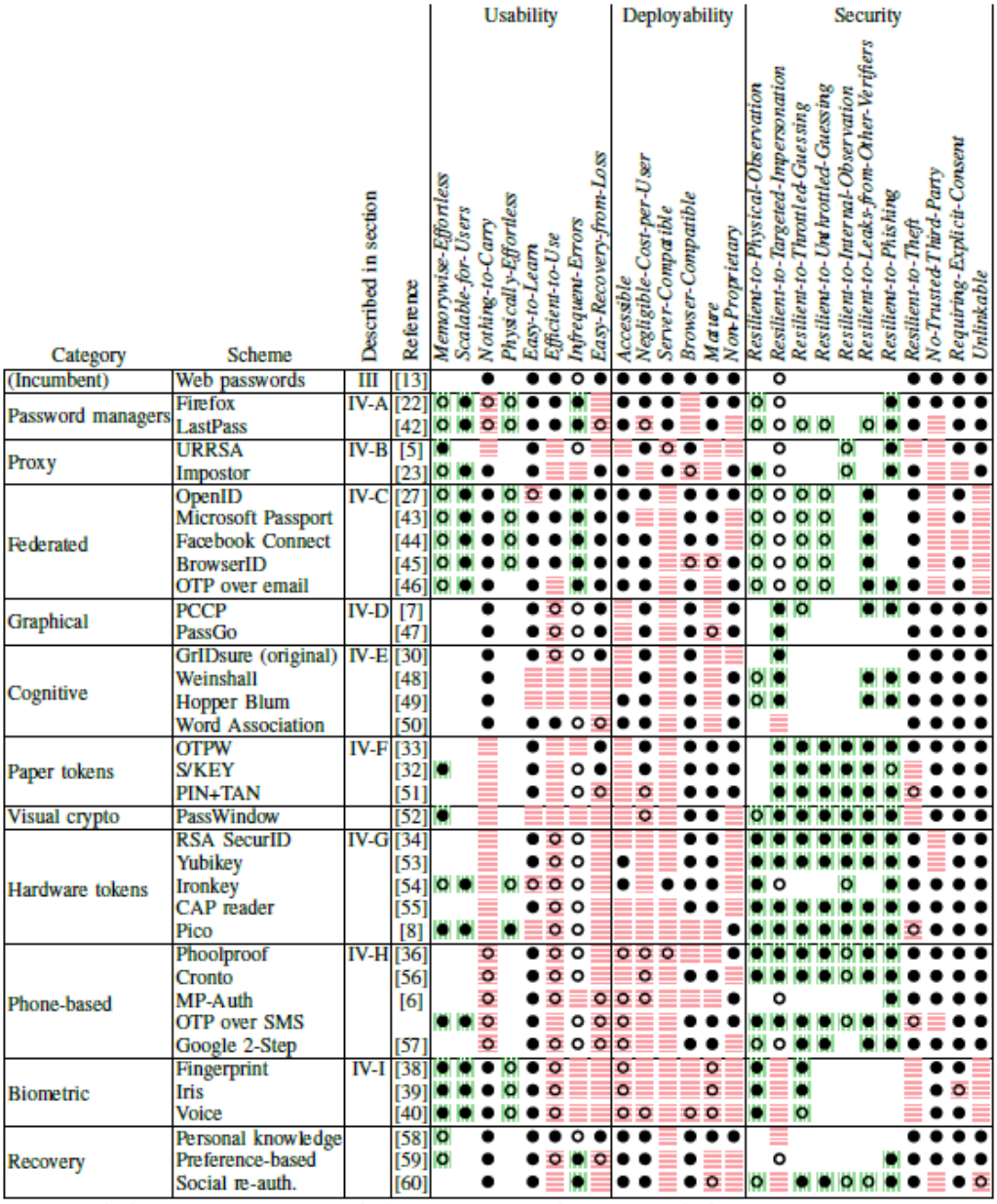
Woods, N. (2016). *"Improving the security of multiple passwords through a greater understanding of the human memory" (*Tohtorinväitöskirja) Jyväskylä studies in computing 249. Jyväskylän yliopisto. Haettu osoitteesta https://jyx.jyu.fi/handle/123456789/51882

Wurster, G. & van Oorschot, P. C. (2008). The developer is the enemy. *Teoksessa Proceedings of the 2008 New Security Paradigms Workshop (NSPW '08) (89-97).* Association for Computing Machinery, New York, USA.

Würsching, L., Putz, F., Haesler, S. & Hollick, M. (2023). FIDO2 the Rescue? Platform vs. Roaming Authentication on Smartphones. Teoksessa *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems (CHI '23),* Hamburg, Germany.

W3C (8.4.2021) Web Authentication: An API for accessing Public Key Credentials Level 2, *W3C Recommendation.* Haettu 10.3.2023 osoitteesta: https://www.w3.org/TR/webauthn/

LIITE 1 ENSIMMÄINEN LIITE



Category	Scheme	Described in section	Reference	Usability: *Memorywise-Effortless*	*Scalable-for-Users*	*Nothing-to-Carry*	*Physically-Effortless*	*Easy-to-Learn*	*Efficient-to-Use*	*Infrequent-Errors*	*Easy-Recovery-from-Loss*	Deployability: *Accessible*	*Negligible-Cost-per-User*	*Server-Compatible*	*Browser-Compatible*	*Mature*	*Non-Proprietary*	Security: *Resilient-to-Physical-Observation*	*Resilient-to-Targeted-Impersonation*	*Resilient-to-Throttled-Guessing*	*Resilient-to-Unthrottled-Guessing*	*Resilient-to-Internal-Observation*	*Resilient-to-Leaks-from-Other-Verifiers*	*Resilient-to-Phishing*	*Resilient-to-Theft*	*No-Trusted-Third-Party*	*Requiring-Explicit-Consent*	*Unlinkable*
(Incumbent)	Web passwords	III	[13]			●		●	●	○	●	●	●	●	●	●	●	○							●	●	●	●
Password managers	Firefox	IV-A	[22]	○	●	○	○	●	●	●	▤	●	●	●	▤	●	●	○	○					●	●	●	●	●
	LastPass		[42]	○	●	○	○	●	●	●	○	●	○	●	▤	●	▤	○	○	○	○		○	●	●	▤	●	●
Proxy	URRSA	IV-B	[5]	●		▤		●	▤	○	▤	▤	●	○	●	▤	▤		○			○		●	▤	▤	●	●
	Impostor		[23]	○	●	●		●	▤	▤	●	●	▤	●	○	▤	●	●	○			○		●	●	▤	▤	●
Federated	OpenID	IV-C	[27]	○	●	●	○	○	●	●	●	●	●	▤	●	●	●	○	○	○	○		●		●	▤	●	▤
	Microsoft Passport		[43]	○	●	●	○	●	●	●	●	●	▤	▤	●	●	▤	○	○	○	○		●		●	▤	●	▤
	Facebook Connect		[44]	○	●	●	○	●	●	●	●	●	●	▤	●	●	▤	○	○	○	○		●		●	▤	▤	▤
	BrowserID		[45]	○	●	●	○	●	●	●	●	●	●	▤	○	○	●	○	○	○	○		●		●	▤	●	▤
	OTP over email		[46]	○	●	●		●	▤	●	●	●	●	▤	●	▤	●	○	○	○	○		●	●	●	▤	●	▤
Graphical	PCCP	IV-D	[7]			●		●	○	○	●	▤	●	▤	●	▤	●		●	○			●	●	●	●	●	●
	PassGo		[47]			●		●	○	○	●	▤	●	▤	●	○	●		●						●	●	●	●
Cognitive	GrIDsure (original)	IV-E	[30]			●		●	○	○	●	▤	●	▤	●	▤	▤		●						●	●	●	●
	Weinshall		[48]			●		▤	▤	▤	▤	▤	●	▤	●	▤	●	○	●				●	●	●	●	●	●
	Hopper Blum		[49]			●		▤	▤	▤	▤	●	●	▤	●	▤	●	○	●				●	●	●	●	●	●
	Word Association		[50]			●		●	●	○	○	●	●	▤	●	▤	●		▤						●	●	●	●
Paper tokens	OTPW	IV-F	[33]			▤		●	▤	▤	●	▤	●	▤	●	●	●		●	●	●	●	●	●	●	●	●	●
	S/KEY		[32]	●		▤		●	▤	○	●	▤	●	▤	●	●	●		●	●	●	●	●	○	▤	●	●	●
	PIN+TAN		[51]			▤		●	▤	○	○	▤	○	▤	●	●	●		●	●	●	●	●	●	○	●	●	●
Visual crypto	PassWindow		[52]	●		▤		▤	▤	▤	▤	▤	○	▤	●	●	▤	○	●	●	●	●	●	●	▤	●	●	●
Hardware tokens	RSA SecurID	IV-G	[34]			▤		●	○	○	▤	▤	▤	▤	●	●	▤	●	●	●	●	●	●	●	●	▤	●	●
	Yubikey		[53]			▤		●	○	○	▤	●	▤	▤	●	●	▤	●	●	●	●	●	●	●	▤	●	●	●
	Ironkey		[54]	○	●	▤	○	○	○	○	▤	●	▤	●	●	●	▤	●	○		○		●	●	●	●	●	●
	CAP reader		[55]			▤		●	○	○	▤	▤	▤	▤	▤	●	▤	●	●	●	●	●	●	●	●	●	●	●
	Pico		[8]	●	●	▤	●	▤	○	○	▤	▤	▤	▤	▤	▤	●	●	●	●	●	●	●	●	▤	●	●	●
Phone-based	Phoolproof	IV-H	[36]			○		●	○	○	▤	○	○	○	▤	▤	●	●	●	●	●	○	●	●	●	●	●	●
	Cronto		[56]			○		●	○	○	▤	▤	○	▤	●	●	▤	●	●	●	●	○	●	●	●	●	●	●
	MP-Auth		[6]			○		●	○	▤	○	○	○	▤	▤	▤	●		○					●	●	●	●	●
	OTP over SMS			●	●	○		●	▤	○	○	○	▤	▤	●	●	●	●	●	●	●	○	●	●	○	▤	●	●
	Google 2-Step		[57]			○		●	○	○	○	○	▤	▤	●	●	▤	○	○	●	●		●	●	●	●	●	●
Biometric	Fingerprint	IV-I	[38]	●	●	●	○	●	○	▤	▤	○	▤	▤	▤	○	▤	●	▤	●					▤	●	●	▤
	Iris		[39]	●	●	●	○	●	○	▤	▤	○	▤	▤	▤	○	▤	●		●					▤	○	▤	▤
	Voice		[40]	●	●	●	○	●	○	▤	▤	○	○	▤	○	○	▤	●	▤	○					▤	●	▤	▤
Recovery	Personal knowledge		[58]	○		●		●	●	○	●	●	●	▤	●	●	●		▤						●	●	●	●
	Preference-based		[59]	○		●		●	○	●	○	●	●	▤	●	▤	▤		○					●	●	●	●	●
	Social re-auth.		[60]			●		●	▤	●	▤	●	●	▤	●	○	▤	○	▤	●	●	○	○	●	●	▤	●	○

●= offers the benefit; ○= almost offers the benefit; *no circle* = does not offer the benefit.
▥= better than passwords; ▤= worse than passwords; *no background pattern* = no change.
We group related schemes into categories. For space reasons, in the present paper we describe at most one representative scheme per category; the companion technical report [1] discusses all schemes listed.

Table I
COMPARATIVE EVALUATION OF THE VARIOUS SCHEMES WE EXAMINED

Bonneau ym. arviointimenetelmän pohjalta tehty matriisi tutkimustuloksista. (Bonneau ym., 2012, s.563)