FIDO Alliance White Paper:
Considerations for Deploying FIDO Servers in the Enterprise

October 2020

Editors:

Salah Machani, RSA
Shane B Weeden, IBM
Norman Field, StrongKey

Abstract

Today, secure access to online applications and services has evolved into a model based on devices, public key cryptography and biometrics to replace the anachronistic use of passwords as shared secrets. Since 2013, the FIDO Alliance has developed open and scalable advancements to eliminate phishing and other security attacks. To introduce these improvements and to educate employees throughout corporate management and IT security, FIDO Alliance has developed a series of best practices and how-to white papers that match the Alliance's goals with the responsibilities and titles of technology professionals. This work is dedicated to eliminating passwords and securing the simple act of logging on within all companies.

A FIDO server is a necessary component in a FIDO implementation. The FIDO server stores the user's public key credential and account information. During a FIDO Authentication or registration flow, the server generates a cryptographic challenge in response to a request from the application. The server then verifies the signature provided by the client using the server's corresponding public key, and logs the user in.

This white paper is intended for IT professionals and identity architects to guide them in choosing the right FIDO server implementation and deployment architecture when integrating and enabling FIDO-based authentication in enterprise applications. Enterprises must consider several factors in their planning to select and deploy a FIDO server, including build vs. buy assessment (and the risks and benefits associated with each), the desired deployment model, the required server capabilities, and the security and privacy requirements.

Deployment Models

There are three common deployment options for FIDO server functionality in the enterprise:

1. FIDO server as part of the Federated Identity Provider (IdP)
2. FIDO server as a product
3. FIDO server as a service

FIDO Server as Part of the Federated IdP

Enterprises that have an IdP solution in place, either as an on-premises product, in a single tenant SaaS environment, or in a multi-tenant SaaS environment, will likely select this option. Typically, the IdP acts as the authentication authority and provides support for a variety of authentication methods. FIDO-based authentication can be added to augment or replace existing authentication methods.

In a federated IdP deployment model, the enterprise owns employee identities and manages the identity sources. The enterprise's on-premises IdP product or tenant in a SaaS environment is the FIDO/WebAuthn relying party (RP). User-registered FIDO credentials will be bound to the enterprise and cannot be used outside the enterprise security domain. The enterprise may choose to enable users to register one FIDO credential for access to all the enterprise applications or require that users register different FIDO credentials for different applications or groups of applications.

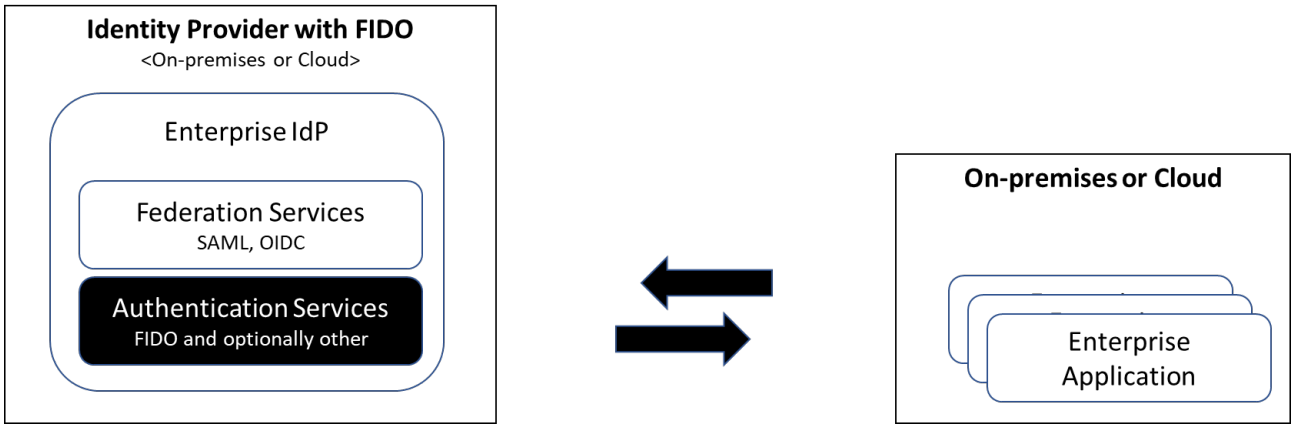

Figure 1: FIDO server as part of a federated IdP

While the enterprise has visibility into the apps/resources that users access with FIDO Authentication within its security domain, it does not have visibility into other non-enterprise RPs that users may access using the same FIDO Authenticator.

FIDO Server as a Product

Companies who want to enable FIDO Authentication for non-federated applications can choose to host FIDO server functionality in their intranet or on dedicated servers in the cloud. The FIDO server functionality can be part of an existing multi-factor authentication system supporting other methods of authentication or a standalone authentication system dedicated to FIDO Authentication. Integration between enterprise on-premises applications and FIDO servers may use an inter-components custom API. In this case, the enterprise has the full control and responsibility to manage, maintain and update the FIDO functionality and the infrastructure required to host and scale FIDO Authentication services.

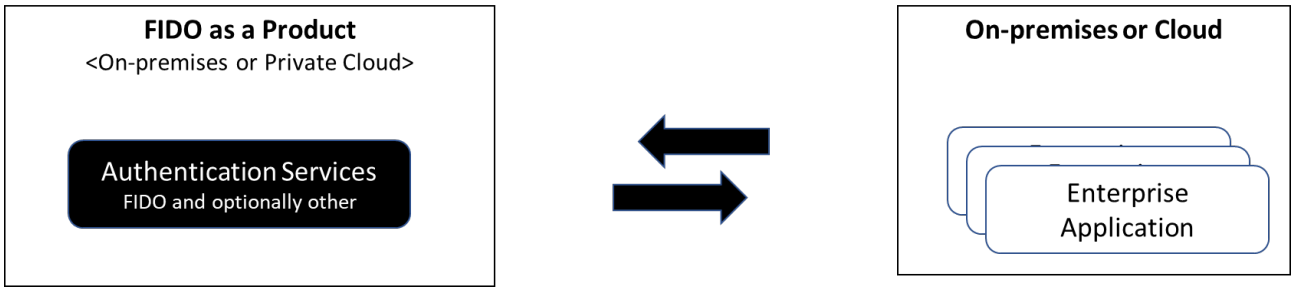

Figure 2: FIDO server as a product

FIDO Server as a Service

Companies may want to enable FIDO-based authentication in non-Federated applications to have full control of the user experience. This means they need to host, manage, and maintain FIDO servers on-premises or on dedicated cloud servers that leverage third-party cloud services offering FIDO registration and authentication as a service. In this case, enterprise applications can access FIDO server registration and authentication services via an API Gateway.

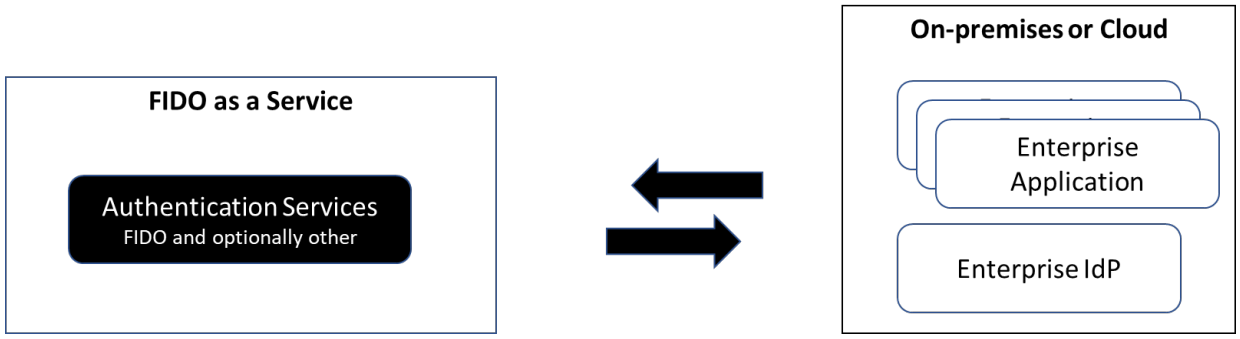

Figure 3: FIDO server as a service

Enterprise Grade Capabilities

When considering FIDO2 server deployment for an enterprise, it is important to assess all the non-functional features that you look for in any other enterprise software deployment. Such requirements are not the focus of this paper; however, it would be remiss not to mention that they should be part of any enterprise software decision-making process, including for FIDO servers.

For example:

- Build vs. Buy
- Open Source vs. Commercial Offering
- Self-managed vs. Managed Service vs. SaaS
- High availability, scalability, audit and reporting, regulatory compliance, integrations, data sovereignty, offline backups, data export, etc.

The following table focuses on a set of FIDO-specific considerations for enterprise deployment:

Capability	**Considerations**
FIDO Protocol support	Do you have requirements for WebAuthn/FIDO2, UAF and/or U2F protocols and does the server support those you require? Make sure the server is FIDO Certified for your required protocols.
Attestation and MDS Policy Enforcement	Does your enterprise have regulatory or internal compliance requirements forcing you to restrict authenticator usage to an allowlisted set of capabilities? If so, ensure that your FIDO server supports policy authoring and enforcement aligned with metadata servers (FIDO MDS).

Capability	Considerations
Fine-grained FIDO Policy Enforcement	Ensure the server can support the runtime flows and associated policy enforcement for your specific FIDO Authentication scenarios such as: • Usernameless authentication (you should require user verification in this scenario) • Requirements related to platform vs. roaming authenticators • Scenarios where user presence may not be required (silent authentication)
Extensions	WebAuthn especially and FIDO2 and UAF more generally support the concept of extensions. Some scenarios will require the use of extensions. It is important to consider if your scenarios have this requirement and if so, ensure that your FIDO server provides the ability to process and act upon extension data. For more details, refer to the "Accepting FIDO Credentials in the Enterprise" white paper.
Registration Lifecycle Management	Beyond just processing registration (aka attestation) and authentication (aka assertion) runtime flows, the FIDO server should support interfaces and integration capabilities for both user self-service and administrative identity and credential lifecycle management. This should include the ability to integrate with the identity onboarding, offboarding, and reconciliation workflows of existing enterprise CRM systems.

Security Considerations

The FIDO2 protocol delivers strong authentication. There are no shared secrets and privacy is built in by design. The protocol can meet the highest level of authentication specified by the National Institute for Standards and Technology (NIST), i.e. Authentication Assurance Level 3[1]. However, like all security technology, the way the technology is implemented influences the overall security stance of the deployed system. This section covers security considerations when deploying, implementing, or selecting a FIDO server.

Public Key Substitution Attack

The FIDO server stores no permanent secrets, nor does it require storage of private data. However, it does enable an attacker to take over an unsuspecting user's account in server implementations that do not take this public key substitution vulnerability into account.

[1] To qualify for NIST AAL3, FIDO Authentication protocol must be used with Token Binding to support verifier impersonation resistance. At the time of writing, there is limited support for Token Binding in web browsers.

When a user registers an authenticator to a relying party (RP) web site, the FIDO Server must store the public key and associate it with both the authenticator and the username in order to use it for future FIDO Authentications. If the database server housing registered FIDO public keys is breached, an attacker may substitute their own public key into one or multiple user entries. In this way, the attacker may compromise and take over any account on the RP or choose to delete the registration data altogether and create a denial of service.

A FIDO server must store the public key (and other user registration data) in permanent storage and therefore must protect that data from being compromised. Protecting the integrity of data at rest is a topic beyond the scope of this document, but many companies, including members of the FIDO Alliance, have expertise in this field. If you are deploying a commercial FIDO server for your enterprise, make sure to study the implementation of the user registration data storage. The vulnerability described in this section is real and in the worst case can result in a major breach.

Relying Party Identifier (RP ID) Override

Relying Parties using FIDO protocol are identified using an RP ID. The RP ID is bound to a FIDO public key credential and determines its scope. By default, the RP ID is set by the FIDO client to the origin's effective domain. RP web applications may override the default value, expand the scope and the list of origins on which the public key credential may be exercised by setting the RP ID to a registrable domain suffix of the origin's effective domain. While this can be desired to allow users to register through one origin and authenticate to multiple origins in the same domain, attackers may take advantage of this feature to escalate their access privileges. For example, an enterprise policy may require that unique FIDO public key credentials be bound to different effective origins within the same domain, such as sales.example.com and hr.example.com. In this case, the RP should not allow unauthorized users to register through sales.example.com, change the scope[2] of their credential to example.com origin and access hr.example.com applications. The FIDO server should verify and reject the registration of FIDO credentials scoped to the higher-level domain.

Similarly, and more importantly, when a FIDO server is part of the SaaS offering, the FIDO credential must not be scoped and exercised on RP origins from different organizations. The RP ID must be scoped to the SaaS customer origin's effective domain and must not be set to a registrable domain suffix that gives access to all customers within the SaaS. For example, when an employee of foo company registers FIDO credentials at foo.saasexample.com, the SaaS provider must have controls in place to prevent any attempt to scope the credential to saasexample.com.

Attestation

Attestation is the means by which a FIDO server establishes trust with the authenticator during registration. Attestation uses the same fundamental technology (public key cryptography) as FIDO registration and authentication. Authenticators are injected at manufacturing time with a public/private key pair specific to the model of the device.

The public key forms part of the attestation certificate, which is chained to a trusted root certificate. The private key is used to sign registration data during a new credential registration. This is how the FIDO server establishes trust in knowing the make and model of authenticator used during a registration operation.

Every time a user registers a credential, the FIDO authenticator generates a new user public/private key pair for that relying party. The registration data, including the public key, is then signed with the attestation private key and is sent to the FIDO server as part of registration. The service that is creating the new account for the user can verify that the "attestation signature" on the newly created public key came from the device.

While the process may appear straightforward, the FIDO 2.0: Key Attestation Format[3] defines several different attestation models that establish trust ranging from very high to virtually no trust. A FIDO server must account for the different attestation models in order to properly interpret the level of trust being supplied by the authenticator.

[2] This is possible in web browsers by editing the RP web application JavaScript at run time.
[3] https://fidoalliance.org/specs/fido-v2.0-ps-20150904/fido-key-attestation-v2.0-ps-20150904.html

Organizations adopting FIDO Authentication may have specific business reasons for restricting the types of authenticators they want to support. This could be an enterprise purchasing a bulk set of authenticators for employees or it could be related to regulatory compliance. Regardless of the business reasons, the FIDO server should be able to properly evaluate the attestation and have the flexibility to set policy based on the result.

Privacy Considerations

FIDO Authenticators may be company-provided or BYO depending on an enterprise's policy. While it is possible for an enterprise to allowlist authenticator makes & models with specific security characteristics to be used with the Enterprise RP, there are situations where the employee may use the same authenticator for work purposes with the Enterprise RP and for personal purposes with one or more external RPs. In such circumstances, the enterprise should not be able to determine from the FIDO protocol exchange details of any external websites that employees may have registered or used a credential. The FIDO specifications and certified products have technical and operational safeguards to prevent the use of one FIDO credential across multiple RPs or the use of global authenticator identifiers which might otherwise allow correlation of a user across RPs.

When an enterprise chooses to outsource FIDO Authentication to a third party IDaaS / SaaS provider, the third-party provider must ensure that there is logical data isolation across FIDO RPs and the user must register and bind a unique FIDO credential per SaaS RP. It should not be possible, for instance, for a user to register a FIDO credential with abc.saasexample.com and use it with xyz.saasexample.com. Registering and binding a FIDO credential to saasexample.com in this case should be prohibited as it would otherwise allow the SaaS provider to correlate a user identity as the user navigates different web applications from different RPs on the SaaS using the same authenticator.

Conclusion

The FIDO server could be part of a federated identity provider, an on-premises product, or a cloud-based authentication service. An organization deploying FIDO must ensure that the FIDO server implementation supports enterprise-grade features. The FIDO protocol protects against phishing and man-in-middle attacks, is privacy-preserving by design, and does not require the storage of secrets on the server. However, poor implementation of the protocol and server functionality can result in vulnerabilities that put the organization's assets at risk. To mitigate and manage these risks, organizations should opt for FIDO server implementations that (a) conform to high standards and adopt common secure coding best practices (such as OWASP secure coding practices[4]) and (b) are FIDO Certified[5]. A FIDO Certified server implementation is compliant with FIDO security and privacy requirements and ensures interoperability with FIDO clients and FIDO Authenticators from different vendors.

Additionally, organizations deploying FIDO-based authentication need to be aware of the various FIDO protocol versions. Protocol extensions and options are available to them to configure and enforce their internal policies and customize the user experience.

For further information on the FIDO security and privacy requirements, the following resources are available:

- FIDO Security Reference: This document analyzes FIDO security. The analysis is performed based on the FIDO Universal Authentication Framework (UAF) specification and FIDO Universal 2nd Factor (U2F) specifications as of the date of this publication.
- FIDO Privacy Principals: This paper describes the privacy-preserving principles that are a core part of the FIDO Alliance's technologies and explains how they reinforce the FIDO Alliance's approach to strong authentication.

[4] https://owasp.org/www-project-secure-coding-practices-quick-reference-guide/migrated_content
[5] https://fidoalliance.org/certification/fido-certified-products

Acknowledgments

The authors acknowledge the following people for their valuable feedback and comments:

- John Bradley, Yubico
- John Fontana, Yubico
- Bill Leddy, Visa
- Dario Salice, Facebook
- Andrew Shikiar, FIDO Alliance