nextAuth

The Requirements Checklist for Multi-Factor Authentication Solutions

The checklist that helps you evaluate MFA solutions and vendors, and define MFA requirements.

SECURITY IMPACT

□ **Does the solution exclude the use of passwords, not even as a fallback method?**

Plenty of so-called passwordless authentication solutions do require a backup password. 81% of data breaches are due to compromised password credentials.

□ **The solution does not require the user to enter a username and codes.**

Username and codes make the solution more susceptible to social engineering like, e.g., phishing.

□ **Does the solution use factors from two or more categories (possession, inherence and knowledge)?**

Having two or more factors from the same category does not provide multi-factor authentication (MFA).

□ **Can multi-factor requirements dynamically be changed, e.g., depending on the authentication context?**

□ **Does the solution provide strong proof of a user login or signature, that holds up to a third party, i.e. does it guarantee non-repudiation?**

Non-repudiation ensures that users can't deny approving an authentication or signature. It provides proof of authentication/signing that holds up to a third party.

□ **Does the solution provide strong proof that the rightful user logged in or signed a transaction, i.e. does the solution provide proof that the rightful user used the mobile device to perform those acts?**

□ **Can the registration be restricted, e.g., one-time usage, restricted in time?**

This is notably not the case with many One Time Password (OTP) apps where the registration QR code simply contains the symmetric key necessary for generating the OTPs.

□ **If the solution can be used offline, can the usage be restricted in a configurable way?**

Providing offline usage limits the security mechanisms that can be used and will weaken the security of the default mechanism. Offline usage should be limited to cases where it is indispensable. At the very least, you should be able to tell if the solution was used in an online or offline way.

Authentication factors

□ **Is the verification of each authentication factor done in zero-knowledge?**

Authentication data (PINs, passwords, biometric samples, symmetric cryptographic keys, private cryptographic keys ...) shouldn't be stored on any server, nor should the server learn this data during the authentication process. Learning authentication data could enable an attacker to impersonate the user. If authentication data is stored on servers, you'll need specific security hardware and the necessary processes to ensure that this data is only known inside this secure hardware.

□ **The solution does not rely on SMS, phone calls or email as a possession factor.**

SMS, phone calls, and email are insecure communication channels and thus only provide a fragile form of possession.

□ **Does the solution rely on cryptography for the possession factor?**

A challenge-response pattern based on cryptographic keys establishes a strong form of possession, as the actual authentication factor (i.e., a cryptographic key) is not sent over during authentication. This is not the case with cached passwords or tokens. These are simply sent to the backend for verification and could thus be intercepted and copied.

- □ **If so, is the cryptographic key material generated on the mobile device?**

- □ **Do you know the level of brute-force security offered? What is the level of security?**

  Brute-force security is usually expressed in bits and determines if an attacker can get «lucky» by guessing the correct response without knowing the cryptographic keys. This level also impacts the proof value of an authentication. Twenty bit corresponds to a probability of 1 in a million (2^{20}) of being right (6 digit OTP). If the solution offers at least 128-bit of brute-force security, it guarantees protection from all practical brute-force attacks.

- □ **Can the vendor guarantee that the solution does not rely on OTPs or OCRA?**

  Typical implementations of OTPs and OCRA output 6-8 digits, making them highly susceptible to brute-force attacks. When OTPs are displayed to the user for entering a third device, the solution becomes vulnerable to phishing attacks.

- □ **Is it built on public key cryptography?**

  Public-key or asymmetric cryptography does not require storing user secrets on the server. It is essential to guarantee non-repudiation, as otherwise users can always claim the server was abused to fake an authentication.

□ **Does the solution provide biometric user verification?**

Biometric user verification provides a user-friendly factor in multi-factor authentication.

- □ **If so, what is the false acceptance rate (FAR), false rejection rate (FRR) and spoof acceptance rate (SAR) for the used biometric verification?**

- □ **Is the used biometric verification in line with GDPR?**

  Be extra careful when the solution processes the user's (raw) biometric data on devices that are not under the user's control and when your target audience also includes minors.

- □ **Does the biometric verification take user presence (liveness detection) into account?**

- □ **Is the biometric user verification impossible to bypass?**

  A solution on the mobile phone that implements biometric user verification as a mere «yes»/ «no» can easily be bypassed.

- □ **Can it be tied to the user's current the set of biometric features?**

  For additional security, invalidating key material on adding a new biometric sample (e.g. new fingerprint) is advisable as this is done at the OS level.

□ **Can the solution make use of a knowledge factor, like a PIN?**

There should always be a fallback for biometrics to authenticate the user.

- □ **If so, can you define a PIN policy?**

- □ **Are there measures in place to prevent brute-forcing of PINs?**

  nextAuth has a proprietary patented MFA solution that verifies the PIN with the aid of both the mobile and the server, ensuring that no brute-force attacks are possible on either the user's mobile or the server.

Mobile devices

□ **Does the solution take full advantage of the security features that are provided by the mobile device and operating system (OS)?**

□ **Is the solution regularly updated to stay compatible with the latest mobile OS versions?**

□ **Does the solution stay up-to-date with new security features offered by the latest mobile devices and OS versions?**

□ **The solution comes with an embedded cryptographic library. Has this library been vetted and is it up to date?**

> A vendor has no control over the mobile OS's cryptographic libraries, which may be outdated and not even updatable by your users.

□ **Does the solution provide insights into which device, OS version, OS patch version… is being used for authentication?**

□ **Does the solution detect rooted/jailbroken devices, debuggers, emulators, hooking, and interception attempts?**

□ **Does the solution have built-in anti-debugging mechanisms and does it protect against reverse engineering?**

Secure Communication Channels

□ **Does the solution provide an extra layer of cryptography to secure mobile to backend communication? Does the solution ensure authenticity and confidentiality of messages?**

> Although TLS provides a secure channel at the transport layer, it does not do so at the application layer (i.e. user authentication or signatures). Moreover, TLS typically terminates at load balancers, not the actual authentication server. An end-to-end channel between mobile device and the actual authentication server, with binding of the user authentication gives stronger security guarantees.

□ **Does the solution enable the mobile and server to engage in continuous authentication thanks to this secure channel?**

□ **Are push messages encrypted and authenticated as originating from your backend?**

> The push message mechanisms, i.e., FireBase Cloud Messaging (FCM for Android devices) and Apple Push Notification service (APNs for iOS devices) do not provide end-to-end security.

Compliance

□ **Does the solution help you meet the following requirements?**

□ PSD2 / Strong Customer Authentication □ eIDAS □ GDPR

□ **Does the solution implement the security controls as defined by OWASP's Mobile Application Security Verification Standard? If so, to which level?**

> Mobile apps should at least implement OWASP MASVS L1 controls and preferably L2+R controls.

□ **Has the solution been penetration tested or has its security been reviewed?**

□ **Is the vendor open to discussing the used security mechanisms?**

□ **Does the solution enable you to keep detailed logs about your users' activity to create custom reports, ideal for security analysis and compliance auditors?**

□ **Does the vendor track vulnerabilities and release patches within a reasonable time frame?**

User experience

□ **Does the solution provide an almost frictionless user authentication, i.e., does it ensure a minimal number of user interactions?**

□ **Does the solution support push login and transaction signing?**

□ **Can the solution be branded or integrated within your own applications? E.g., does the vendor provide white-label components, a comprehensive API and SDKs?**

□ **Can users use their own devices to complete authentication?**

> Solutions that depend on mobile device management (MDM) are not suitable for deployment towards a general user population.

□ **Does the vendor support Android and iOS?**

□ **Can the solution be used for web and non-web applications, such as, e.g., authentication in and between mobile apps, kiosks, etc.**

□ **Does the solution provide the right amount of context to the user to help make informed decisions?**

□ **Does the solution support registration of multiple devices and authentication methods per user?**

> Supporting multiple devices and authentication methods allows users to authenticate even when they don't have their primary device or are unable to authenticate under normal circumstances.

□ **Does the vendor provide modules or APIs to allow user self-registration and self-management?**

> Self-service capabilities, such as registering new devices and choosing between authentication mechanisms, lighten the IT team's administrative load, accelerate user adoption, and are crucial to user retention.

Strategic business value

- □ **Does the solution align with your deployment model?**
  In your infrastructure, in your cloud, or as a SaaS?

- □ **Does the solution enable you to keep control over your users' authentication process?**
  Do you need a 3rd party provider, or does the solution allow you to fully own the authentication process and dispose of the users' authentication data?

- □ **Is your solution compatible with other business initiatives, such as enabling remote work or onboarding cloud applications? Does it integrate with your Single Sign On (SSO)?**

- □ **Does the authentication solution support the full breadth of users' authentication use-cases?**

- □ **Is the solution usable for both your users (enrollment, activation and daily authentication) and administrators (user and solution management)?**

- □ **Is the vendor willing and able to extend their product offering, potentially with third-party technology, to precisely solve your authentication challenges?**

- □ **Does the vendor have an R&D department that keeps the solution relevant?**

Integration

Analysis and Proof of Concept

□ **Does the vendor provide you with a solution that meets your requirements, instead of force-fitting their solution into your environment?**

□ **Do you have the opportunity to run a proof of concept?**

Deployment

□ **What do you need to do to deploy the solution regarding server-side integration, app development?**

□ **Can the solution be deployed in a matter of days instead of months?**

□ **Does the solution provide extensive documentation on integration?**

□ **Does the solution run on both on-premises architectures and emerging cloud models?**

□ **Is the solution cloud-agnostic?**

> Avoid being locked into one specific cloud provider, e.g., AWS, Azure, Google.

□ **Does the solution include RESTful APIs, SDKs, or plug-ins?**

> These simplify integration into other applications and make it easier for your developers to quickly add authentication and get back to developing new, strategic features that increase business value.

□ **Does the vendor support Security Assertion Markup Language (SAML) to expedite the single sign-on (SSO) integration?**

Integration

Scalability

□ **Does the solution scale horizontally and vertically?**

> Horizontal scaling refers to adding additional nodes, vertical scaling describes adding more power to your current machines. In general, horizontal scaling is cheaper.

□ **Is the server of the solution capable of processing large volumes of authentication requests, and can it support mass deployments in various customer interfacing applications?**

Availability (SaaS only)

□ **Is the solution's availability statistically higher than 99,99%?**

□ **Do you know what the vendor would do in case of the 0,01%?**

Basic Price Model

□ **Is the vendor's pricing per user, active user, device, integration, authentication, etc.?**

Payment per (active) user makes it easier to predict costs and scale.

□ **Is there a cost associated with every authentication attempt?**

Next to a possible direct cost as per the agreed pricing, make sure communication costs (e.g., sending out SMSes, emails, making phone calls, etc.), if any, are included, or costs are taken into account.

□ **Is the solution a SaaS, or is it deployed as a managed service or in-house?**

Running software as a SaaS puts most of the responsibility for keeping it running on the provider. This comes with a cost and reduces your control with strong vendor lock-in. A good SLA is crucial but will increase the price. Running software as managed service or in-house will enable you to maintain control over the solution, its deployment, the service and performance level, and its costs. Deploying authentication software in the same environment as your other systems also has the benefit of reducing issue resolution times and limiting the number of suppliers/vendors involved in this process.

□ **For tiered pricing plans, are all required functionalities available in your license tier or make a list of upcharges?**

Direct Cost Components

□ **Are you proactively informed of the typical time and labor commitment setup and integration take, including all testing and troubleshooting?**

□ **What is the cost of hosting the server software (servers, network, maintenance, monitoring, etc.)?**

□ **What is the annual license cost for ongoing upgrades, patches and support?**

□ **What is the cost of performing updates of the solution?**

□ **Are high availability (redundancy) and business continuity (disaster recovery) configurations included in the price?**

> Some vendors charge additional costs for setting up multiple instances of their server software. This can end up tripling your costs.

Indirect Costs/Benefits

□ **Does the solution enable you to save on password resets by not supporting passwords?**

> Organizations pay between €63 and €184 per password reset, and up to 50% of helpdesk calls are password reset-related.

□ **Does the solution enable you to save on SMS?**

> Organizations pay up to 10 euro cents per SMS.

□ **Does the solution enable you to get rid of HSMs?**

□ **Does the solution rely on hardware tokens as an authentication factor?**

> Token-based solutions are often more expensive to distribute and manage than they are to buy. Furthermore, up to 25% of the IT support's workload revolve around resolving issues around tokens.

□ **Does the solution provide upfront value or incur hidden costs to your organization?**

End-user Rollout and Migration Costs

□ **Can you roll out the solution using your in-house resources?**

□ **Will the end-user enrollment/migration take less than a week?**

> You should be able to enroll users in a matter of days.

□ **Does using the solution require any additional administrative training and helpdesk time? If so, does the vendor provide training?**

□ **Is bulk enrollment/migration possible?**

User Support

□ **Is the solution designed to reduce helpdesk interventions?**

E.g., by providing a self-service portal that allows end-users to manage their own accounts, add or delete devices, recover credentials, and perform other simple tasks? Or by providing a unified login experience, reducing the complexity for end-users of the entire authentication process.

□ **Is it easy to add new users or revoke credentials?**

Support

□ **Does your vendor provide live third-line support via email, chat, or phone? What is the associated SLA?**

nextAuth provides a mobile, passwordless, and patented True Multi-Factor Authentication™ solution. Our passwordless True MFA™ solution couples a frictionless user experience with strong multi-factor authentication. We enable your users to log in, approve transactions, and sign documents using only a PIN or biometric. No more passwords, no more usernames,
no more OTPs.

Our unique asymmetric cryptography enables you to better safeguard your users' data, provides strong non-repudiation of e-signatures, and helps you meet SCA (PSD2) and eIDAS requirements. Whether you deploy our white-label app or implement our SDK, you're sure to provide your users with a smooth and highly-secure authentication and e-signature process.

We'll help you turn your authentication process into a competitive advantage; Request a demo with our Partnership Director, Jan Waegemans, at jan.waegemans@nextauth.com
or click the button below to pick a time slot in his calendar.

Schedule a conversation