Multi-Factor Authentication Deployment Guide

A complete guide to selecting your MFA solution and planning your MFA rollout

Introduction

As threats to password security have increased in recent years, multi-factor authentication (MFA) has rapidly gained adoption as a method for increasing the assurance of authentication for consumer and enterprise web and mobile applications.

Authentication is generally accomplished by validating one of three types of factors: something you know (e.g. a password), something you have (e.g. an ID card), and something you are (e.g. a fingerprint). Multi-factor authentication employs two or more types of factors.

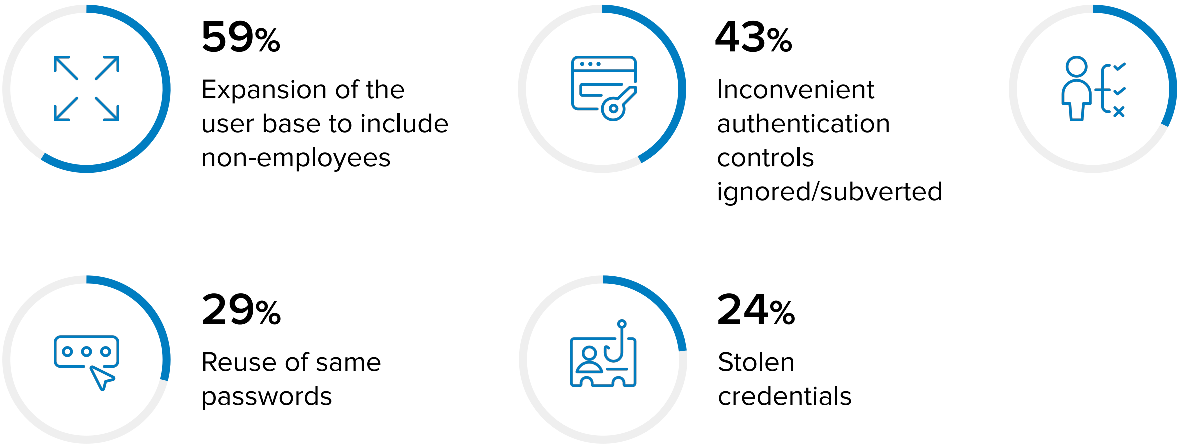
Web and mobile products most commonly employ the use of multi-factor authentication with a password used in conjunction with a time-based token that the user possesses, a push notification to a mobile app, or biometrics. However, the various approaches to MFA vary widely and present different tradeoffs.

In this guide, we compiled information on why an MFA solution is a no-brainer, and the best practices for deploying MFA. We review the results of a survey completed in partnership with IDG that shows where the priorities of your peers lie and how Identity and Access Management (IAM) play a part in strong authentication and security. Next, we explore things to consider before deploying your MFA solution, like policies and access needs. Finally, we provide further practical advice for people building multi-factor authentication for their applications, based on our observations working with engineering and product teams.

Using IAM with MFA in the Age of Megabreaches

There's no shortage of threats, including: malware, hacking, phishing, and social engineering and these tactics often lead to account compromise and credential theft.

Top Identity-Related Security Concerns



59% Expansion of the user base to include non-employees

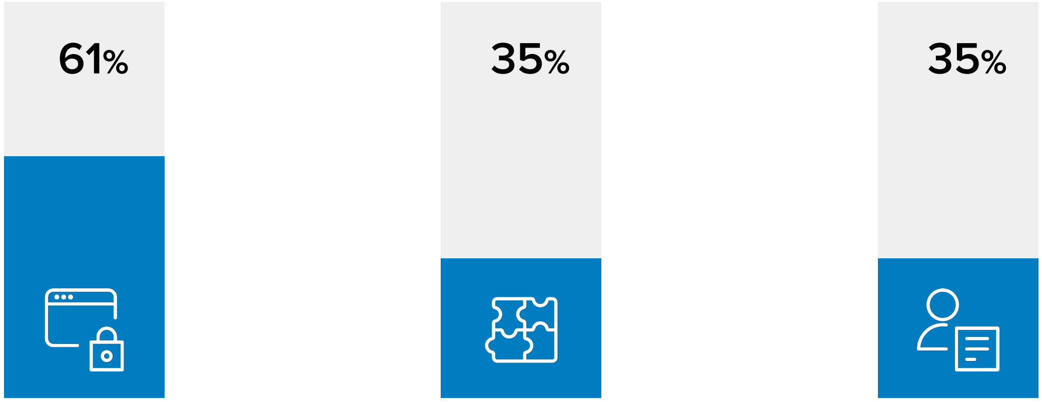
43% Inconvenient authentication controls ignored/subverted

33% Lack of IAM policies

29% Reuse of same passwords

24% Stolen credentials

Top Challenges in Managing Identity and Access



61% Managing identity and access across application environments

35% Integration with current security solutions (50% for larger enterprises)

35% Ability to collect and report on user access info and patterns

Looking Ahead—Priority of IAM and Assessing Current IAM Capabilities:

92%

77%

of managers do

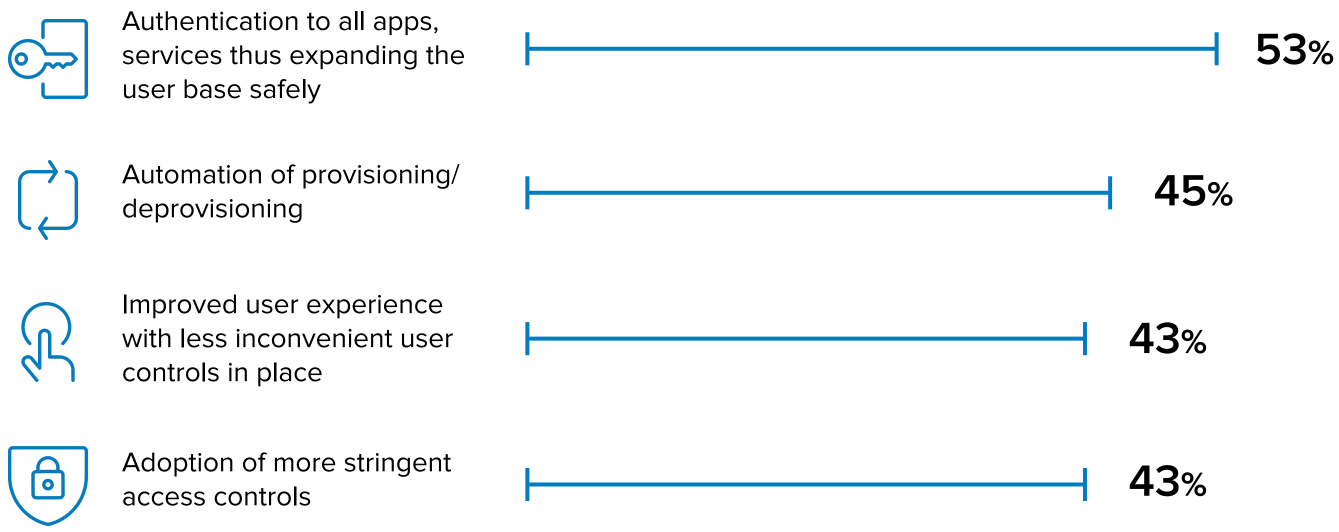
30%

report a good or better ability to detect compromise of credentials

45%

integrate IAM data into their Security Operations Center (SOC)

Addressing Security Concerns: Most Important Potential Benefits of IAM Solutions



Authentication to all apps, services thus expanding the user base safely — **53%**

Automation of provisioning/ deprovisioning — **45%**

Improved user experience with less inconvenient user controls in place — **43%**

Adoption of more stringent access controls — **43%**

* Research from IDG

8 Things to Consider Before Enabling Multi-factor Authentication (MFA)

Passwords are hard. The (what feels like constantly) growing list of security requirements are intended to make passwords secure, but, in many cases, they have had the opposite effect. Complex passwords that meet all the security requirements are often difficult to remember, so they are reused across many sites. Users scribble them on sticky notes. They weave in easily discoverable pet's names, birthdays, and phone numbers. It's no way to keep data secure.

Thankfully, organizations are starting to not just understand, but also support the concept that while access should be hard for hackers, it needs to be easy for legitimate users. And, the best way to make that happen is with multi-factor authentication, or MFA. MFA is a great way to secure your users' apps and services from unauthorized access. Here are some points to consider as you plan your deployment.

1. User education

You're deploying multi-factor authentication to reduce security risks from password-only access, but some users may see this as an inconvenience. They may be worried that this process change will take up time they feel could be better spent elsewhere - after all, entering an OTP or accepting a push notification does add time to the login process. Nonetheless, it's critical to ensure everyone — from management to IT teams to security teams to end users — are aligned on why you're making the shift to MFA. It is important to achieve buy-in from the entire organization to ensure everyone plays a role in keeping the company secure. Do this through education, so each user can appreciate the security benefits they contribute to by taking this additional step.

For example, a common approach is to send out emails coming from IT on upcoming changes - well in advance of when these changes will happen. Be sure to include screenshots, FAQs, and contact information for employees to reach out for assistance.

2. Consider your MFA policies

A good MFA deployment will balance security with usability to avoid becoming too onerous, so you will want to consider how you define MFA policies to govern how and when a second factor is required. It may seem a bit counterintuitive, but sometimes the key is to prompt for step-up authentication less often instead of more. A well-considered risk-based policy configuration should trigger step-up authentication challenges only when necessary. For example, a policy could ensure that a second factor is required every 8 hours when logging in from a known network, or only required when logging in from a new device or new geolocation. Or, maybe you have a certain group of user accounts with broad access to sensitive data, and you need a stricter policy for them. For example, developers in your organization with access to source code, or executives with access to sensitive data may need to provide a stronger factor type or require additional MFA prompts when logging into sensitive apps. MFA allows you to require a second-factor when these types of user groups attempt to access the sensitive resource, but not, say, when they access the company events calendar. The basic idea is that additional verification should be as transparent as possible to the user to foster a good user experience without compromising on security.

3. Plan and provide for a variety of access needs

There will be scenarios where a user has internet access but has little or no service from their cell phone carrier. This could be on a wifi-enabled airplane, at a rural home, or simply in the basement of a large concrete building. In these cases, where voice and SMS may not be feasible, Okta Verify with push or one-time password (OTP) are better choices, as their communication is encrypted over the phone's Internet connection. Hardware devices that generate event-based or time-based one-time passwords (TOTP) don't require a communication channel at all. They are also more difficult to tamper with or copy. But, along with the cost to deploy, a physical device becomes one more thing for employees to carry around, forget at home, or lose. Thus, these factor types may not be the go-to choice for short-term contractors or in situations where there is substantial churn in workers. When it comes to MFA factors, there are many options to solve for a wide variety of scenarios. Choose what works best for each scenario in your organization, keeping in mind that multiple policies and factors can be used when there isn't (and there hardly ever is!) a one-size-fits-all solution to accommodate all situations.

Generally speaking, these deployment tips ensure both enhanced security and a great end-user experience -

- For hardware that supports it, allow users to use biometrics as their second factor (Windows Hello, Touch ID, etc). This simplifies end user experience and also addresses scenarios where users may not have internet access
- Make at least two types of factors available for users, so that they have one as a backup
- Allow users to self-service reset their factor (for example, reset an authenticator app on a lost phone)
- Start your deployment right by only enabling strong factor types (mobile app authenticators, push notifications, biometrics)

4. Think twice about using SMS for OTP

SMS is familiar and easy to rollout. And, with the prevalence of cell phones and tablets, it's nearly everywhere, and has become a common communications channel for OTP delivery. SMS has generally been assumed to be secure enough for this purpose, but that is due in part to the fact that the infrastructure is mostly proprietary and opaque. Research shows that SMS security is lacking, and not only when it comes to documented vulnerabilities. With SMS, you are trusting security to the telecom companies, and even if you trust that they have security best practices in place, there is always a risk of compromise through spoofing and social engineering. In many cases, it's not that technically challenging for an attacker to port your number to a device they control, and gain access to your SMS messages and OTPs.

Common issues with using SMS OTP as an MFA factor

1. **SIM Swapping/SIM Hacking**

   The SIM card in your phone essentially tells your phone which wireless carrier to connect to, and what phone number to connect with. In a SIM swap/SIM hack attack, a threat actor impersonates you and convinces the carrier that they are, in fact, you. Ultimately, your phone number is then assigned to a new SIM card on a different phone.

While SIM swapping/SIM hacking has been an issue for years, this attack type became very publicized in 2019, when Twitter CEO Jack Dorsey's own Twitter account was victim to a group of vandals that convinced the wireless carrier tied to his phone number to switch that number to a new phone in their possession. In a SIM swap/SIM hack, threat actors do not need access to any of your physical devices to gain access to your accounts - once your number has been switched to a device in their possession, they can receive all SMS OTP messages tied to your online accounts.

2. Lost devices & synced devices

You've lost your phone — it's annoying, but happens from time to time. But, what happens when your phone number is connected to your banking apps, social media, and more? In general, multi-factor authentication is considered a combination of two pieces of evidence which prove you are who you say you are - a knowledge factor (something you know), an inherent factor (something you are), or a possession factor (something you have). Using a password and an SMS OTP as a factor is a combination of knowledge and possession factors. But if you've lost your phone, in theory, you should no longer be able to receive messages to validate your identity. However, because we can now sync messages across multiple devices, even if you have lost the device which should be considered your second factor, you still have access to your accounts. This is considered insecure when you can forward text messages to your email — which may have an insecure password, or if you're using a VoIP number that can be accessed on any device which may or may not have a PIN code.

3. Taking over your online wireless account

Keep in mind that most of the common wireless providers allow you to view text messages via your online account, within their web portal. If your account for the web portal itself isn't protected with a second factor, and if you are using an easily guessed password which you use with many online accounts, a threat actor could monitor your account for an SMS OTP message that you initiated for a banking app, Facebook, etc, giving them access to those accounts.

4. Social engineering & phishing

Unfortunately, SMS OTP is not the only form of authentication susceptible to social engineering phishing attacks. Less secure factors like passwords and security questions are equally susceptible. In a social engineering attack, a threat actor posing as an employee from a service you trust convinces you to hand over your account credentials, and in many cases, the SMS OTP sent to your device as well. For example, if you get a call from your "bank" telling you that they need immediate access to your account for security purposes, you may inadvertently give a threat actor your username/password combination, as well as the SMS OTP code which gets sent to your phone during the login process. Phishing attacks aren't just specific to email. You can receive a phishing text message as well, and if you inadvertently type a username/password combination into a malicious website, the threat actor could then use a few of the aforementioned attack types to take over your account.

While NIST recommends against using SMS for these reasons, ultimately you need to perform your own risk assessment based on your users, use cases, and the data being secured. After all, MFA with SMS is still better than no MFA at all.

5. Check compliance requirements carefully

Most IT compliance standards such as PCI DSS, SOX, and HIPAA mandate strong user authentication controls, making them likely motivators for an MFA deployment. It seems obvious, but if your goal is to meet such standards, make sure to have a detailed understanding of the requirements so you can tailor configuration and policies to them. For example, PCI and HIPAA compliance both require strong authentication, which is at least two strong authentication methods out of these three: something you know, something you have, and something you are. And SOX focuses less on technology—but to pass an audit, you'll still need to prove that your organization's finance and accounting data is secure. IT compliance requires implementing relevant standards, but it also requires an ability to prove that you've met them. Make documentation part of your configuration and implementation so you'll be able to quickly and confidently prove in an audit that they've been met. Your future self (and your org!) will thank you.

6. Have a plan for lost devices

The second authentication factor type in a typical MFA deployment is "something you have" (the first being "something you know" and third being "something you are"). In the case of SMS, voice, or an authentication app like Okta Verify or Google Authenticator, the user has their phone. In the case of a hardware token from YubiKey, RSA, or similar, the user has their token. But anything a user has, a user can lose. A procedure for handling lost devices should already be part of your comprehensive IT helpdesk playbook. Extend it to include devices used for MFA, and ensure that reporting a lost device results in:

- Expiring any current sessions and requesting the user re-authenticate
- Disassociating the device from the user's account and access rights
- Remote wiping of corporate information on mobile devices (if necessary; usually done on company-owned devices)

It's also important to audit the user account's activity prior to the point in time when the device was lost to note any unusual activity. If there is anything suspicious, consider the possibility of a breach and escalate accordingly. Once the immediate security concerns are handled, focus should shift to getting the employee back to work with a replacement device or login method. For example, an alternative process like calling the IT helpdesk to verify identity requirements can allow the employee to be productive while replacement factors are implemented.

7. Have a plan to deploy MFA to remote workers

Remote work is on the rise, so it's critical that your organization bolsters security as more employees work remotely. Ideally, new employee onboarding is done in the office and existing employees have in-person access to IT. However, remote work brings a new challenge for both deployment and troubleshooting. To address deployment-related issues, it's best to enable factors that allow users to quickly get up and running - for example, built-in device biometrics or mobile app authenticators like Okta Verify.

This way, your users do not need to wait for an additional hard token to be shipped to them. This is also where end user communication becomes critical - ensure they have the resources they need to get set up and troubleshoot. In the case of new employee onboarding, some organizations will host virtual onboarding sessions and send setup instructions to the employee's personal email address, before they have access to their corporate email.

8. Phase your deployment, be prepared to review and revise

It's rare that complex deployments and policies are a perfect fit the first time. With a process change that will affect all employees, it's always a good idea to track the effectiveness of an MFA solution as it is being deployed and used and be able to refine policies based on observations. Ideally, you will be able to phase your deployment in a manner where IT/Security will start using MFA first. From there, you can expand to additional user groups. Get comfortable with the auditing functionality early in the process and it will be invaluable for troubleshooting and adjusting policy configuration. Once you've deployed MFA to users, use auditing tools to spot check adoption and use. A mechanism that allows user feedback to be reported can also be a good idea. And while users may not always take the time to provide written feedback, an audit trail gives you some visibility into what they experienced. Did it take them three tries to enter their OTP? Did they give up? Problems like this could indicate a misconfiguration, a gap in user education, or simply a scenario that wasn't considered in the initial rollout plan. Using audit tools and encouraging employee feedback assures all stakeholders that the system is working as intended and new security policies are being successfully adopted.

Bonus: Consider adaptive MFA

These tips are a great start, and step-up MFA can even allow fine-grained control over how and when MFA is applied, but it requires careful consideration to configure. In some cases, even for well-defined policies and criteria, you may want to be able to make decisions on-the-fly based on changes to user or device context. To take advantage of the ability to make dynamic changes, check out Okta's Adaptive MFA solution. Adaptive MFA works by noting access patterns and then adapting the policy around each user or group. For example, an employee who routinely travels and checks email from overseas may only periodically require a second authentication factor, but an employee who never travels would immediately receive an MFA challenge should they do so. Risk-based policies, like prompting for a step-up authentication challenge when trying to access resources through an unauthorized proxy or automatically blocking access from known malicious IPs can also kick in when triggered by suspicious events. Adaptive MFA is a powerful tool to automatically derive dynamic policies over time—ones that are tight enough to give you the security your organization requires, but flexible enough to treat your users as individuals.

Building Secure Multi-Factor Authentication

Three best practices for engineering and product leaders

Introduction

Volumes have been written about how to design secure authentication for electronic systems. In this brief, we provide some practical advice for people building multi-factor authentication for their applications, based on our observations working with engineering and product teams.

We explore three ways to increase the security of your MFA feature:

1. Understand and manage the vulnerability of your account recovery flow
2. Protect your login flow from brute force attacks
3. Design to manage tradeoffs between risk, usability, and cost Throughout this brief we assume that the password has been compromised, and examine the second-factor through this lens

Understand and manage the vulnerability of your account recovery flow

Multi-factor authentication is only as secure as its account recovery flows. In many highly publicized recent cases, attackers have been able to exploit vulnerabilities in the account recovery process to gain control of an account. For example, Acme's web application provides for MFA based on a soft token app installed on a user's phone and allows the user to enroll a phone number to receive a backup second-factor for account recovery in the event that the user is unable to access their soft token. The strength of Acme's second-factor now depends on the strength of the telecom provider's processes for authenticating the customer and forwarding calls or SMS. Will the attacker be able to impersonate the user and convince or pressure a customer service rep to route calls or SMS to a number she controls? Every second-factor will need a method for replacement, and so this begs the question of how to develop secure recovery flows. Here are some tips for designing a secure recovery flow for your second-factor, noting that different approaches will suit different circumstances:

- **Independence of primary and secondary factors**
  Separate the recovery of the second-factor from the recovery of the primary factor. Should an attacker gain access to the primary authentication factor, the second-factor becomes immaterial if it can be reset with possession of just the password. Further, the recovery flow for the second-factor should be completely separate from the recovery flow for the password. For example, if an email message is the method for recovering the password, make sure to recover the second-factor through an altogether separate channel.
- **Involve an administrator**
  An administrator can in many scenarios implement a sophisticated high assurance authentication method.

In enterprise scenarios, companies will be in the best position to authenticate members of their organization through shared secrets derived from the content of the employee's work or profile, the company, and human relationships. One notable approach is to ask an employee's manager to authenticate the user and then authorize IT to execute the MFA reset.

In consumer scenarios, an administrator will be able to interrogate a user across a large set of shared secrets. For example, upon onboarding, consumer banking applications will collect a large set of obscure personal details that become shared secrets for account recovery. Recent events in the person's history with the application or company can also constitute viable shared secrets. The evaluation of a set of shared secrets can be automated via web or voice and can in many cases provide better assurance than a human through lower vulnerability to social engineering.

- **Provide a backup second-factor**
  Many scenarios require an automated method for recovering the second-factor (for example, products serving large numbers of users where 1:1 support is prohibitively expensive, or there is a need to reduce operational costs). Enrolling the user in more than one second factor at the time of onboarding allows the user to recover a second-factor by completing authentication through a backup second-factor. One notable, simple, and low-cost example is to provide users with a card (either physical or printable) with a set of codes that can be used only once, and that can be used as a backup second-factor.

Protect login flows from brute force attacks

As the availability of inexpensive computing resources increases, so does the vulnerability of authentication systems to brute force guessing attacks. However, several simple techniques can be used to significantly improve the security of your multi-factor authentication in the circumstance where the password has been compromised:

- **Login flow sequence, rate limits, and account locking**
  Placing the challenge for the second-factor on a page beneath the login page has two benefits. First, it protects your user from an attack aimed at locking them out of their account once a failed login attempt limit is reached (with rate limits applied to the primary factor). Second, obscuring the second-factor provides an attacker with less visibility into another layer of security. Implement a rate limit and lock policy on the second factor. The probability that a user enters their token incorrectly multiple times is low. As such, your suspicion of attack should grow with each failed attempt. Response times should grow with each subsequent attempt to decrease the aggregate number of attempts possible per unit time, with a complete account lockout (where feasible) upon several consecutive failed attempts. For time-based second-factors, manage rate limits according to the life of the token.

- **Logs and alerts**
  Collect and analyze unsuccessful second-factor attempts. In the event of several failed second-factor challenges, alert the user or an administrator of this suspicious behavior, and prompt the user to enroll a new token.

- **Use an out-of-band token**
  A second-factor that is verified through a channel separate from the primary factor adds extra protection against brute force attacks (and phishing). For example, a popular new factor sends the user a push notification on a mobile phone with details about the authentication request and a prompt to accept or deny the request. This channel is inaccessible to a traditional brute force guessing approach.

Design to manage risk, usability, and cost

The design of a multi-factor authentication feature will have significant implications on security, usability, and cost in any context. A higher assurance second-factor can in some cases present the burden of increased hassle for end users and administrators, which can impact the adoption of MFA for your product, and thereby decrease security. Here are some best practices for balancing risk, usability, and cost:

- **Offer a spectrum of options to serve diverse user populations**
  Different user populations present different levels of risk and hence, warrant different levels of assurance. For example, an administrator can have a larger scope of access than an individual user. As such, you may wish to provide relatively stronger second-factors for administrators, while offering more convenient options for users. In consumer scenarios, different users will have different preferences on the level of security they wold like on their account. In some cases, a lower assurance, more convenient option (such as SMS) that is actually used may provide more security than a high assurance option that lacks adoption.

- **Support federated authentication**
  In enterprise scenarios, many companies are implementing authentication and MFA locally for identities they manage, and federating to resources. This approach allows product development teams to outsource the administration of policy and security processes to customers. Enabling customers to implement MFA independently allows them to optimize across the aforementioned considerations according to their specific circumstances and constraints. For example, a customer can design the administration of account recovery to suit their specific IT function. This outsourced approach has the added advantage of allowing users to use one token for access to all resources.

Conclusion

Roadmap to MFA Success

To recap, multi-factor authentication is a compelling method for application developers to increase the security of access to their applications. Many steps must be taken to ensure the security of an MFA feature, including analyzing the second factor recovery flow, designing against brute force attacks, and balancing security, usability, and cost.

A modern, automated approach to multi-factor authentication helps take control of credentials to drastically reduce the risk of a data breach. Where should organizations start?

We recommend you focus on these key milestones:

1. Eliminate passwords wherever possible
2. Enable strong, unique passwords everywhere else
3. Secure account recovery flows with independent primary and secondary factors
4. Harden critical applications with step-up authentication
5. Apply unified policy to on-premises, cloud, and mobile applications
6. Automate provisioning with accurate entitlements
7. Deprovision at scale, and enable visibility and reporting
8. Roll out centralized, real-time reporting and alerts for all authentication events
9. Integrate your identity management strategy with existing security tools
10. Extend identity and multi-factor authentication to include partners, suppliers, and contractors

Why Okta for MFA?

Okta's modern approach to identity management is uniquely positioned to help businesses take control of both identity and multi-factor authentication to reduce data breaches. With Okta's multi-factor authentication, you can:

Quickly enable MFA for your workforce and customers

- Deploy MFA quickly and easily with 5,000 out of the box connections on the Okta application Network
- Extend coverage to on-premises applications via support for RADIUS, RDP, ADFS, and LDAP, as well as header-based auth and Kerberos via Okta Access Gateway
- Facilitate intelligent, contextual access decisions based on device and connection attributes

But, to protect against data breaches in a comprehensive way, you need more than strong authentication. With Okta it's easy to:

Centralize identity

- Reduce account management complexity
- Unify access for users to eliminate passwords while simplifying access
- Mitigate risk and reduce identity sprawl by restricting access to services via intelligent SAML connections

Reduce the attack surface

- Automated provisioning and deprovisioning accelerates consistent onboarding while eliminating orphan accounts
- Extensible for custom applications via SCIM, SDK, and Okta's API
- Complete lifecycle management ensures the right level of access to the right applications with access request workflows

Enable rapid response to compromise

- Automated provisioning and deprovisioning accelerates consistent onboarding while eliminating orphan accounts
- Extensible for custom applications via SCIM, SDK, and Okta's API
- Complete lifecycle management ensures the right level of access to the right applications with access request workflows

To see how easy it is to administer Okta's Adaptive multi-factor authentication solution and pilot the authentication process, watch this demo.

About Okta

Okta is the leading provider of identity for the enterprise. The Okta Identity Cloud connects and protects employees of many of the world's largest enterprises. It also securely connects enterprises to their partners, suppliers and customers. With deep integrations to over 6,000 applications, the Okta Identity Cloud enables simple and secure access from any device. Thousands of customers, including Experian, 20th Century Fox, LinkedIn, Flex, News Corp, Dish Networks and Adobe trust Okta to work faster, boost revenue and stay secure. Okta helps customers fulfill their missions faster by making it safe and easy to use the technologies they need to do their most significant work. For more information, visit us at www.okta.com or follow us on www.okta.com/blog.