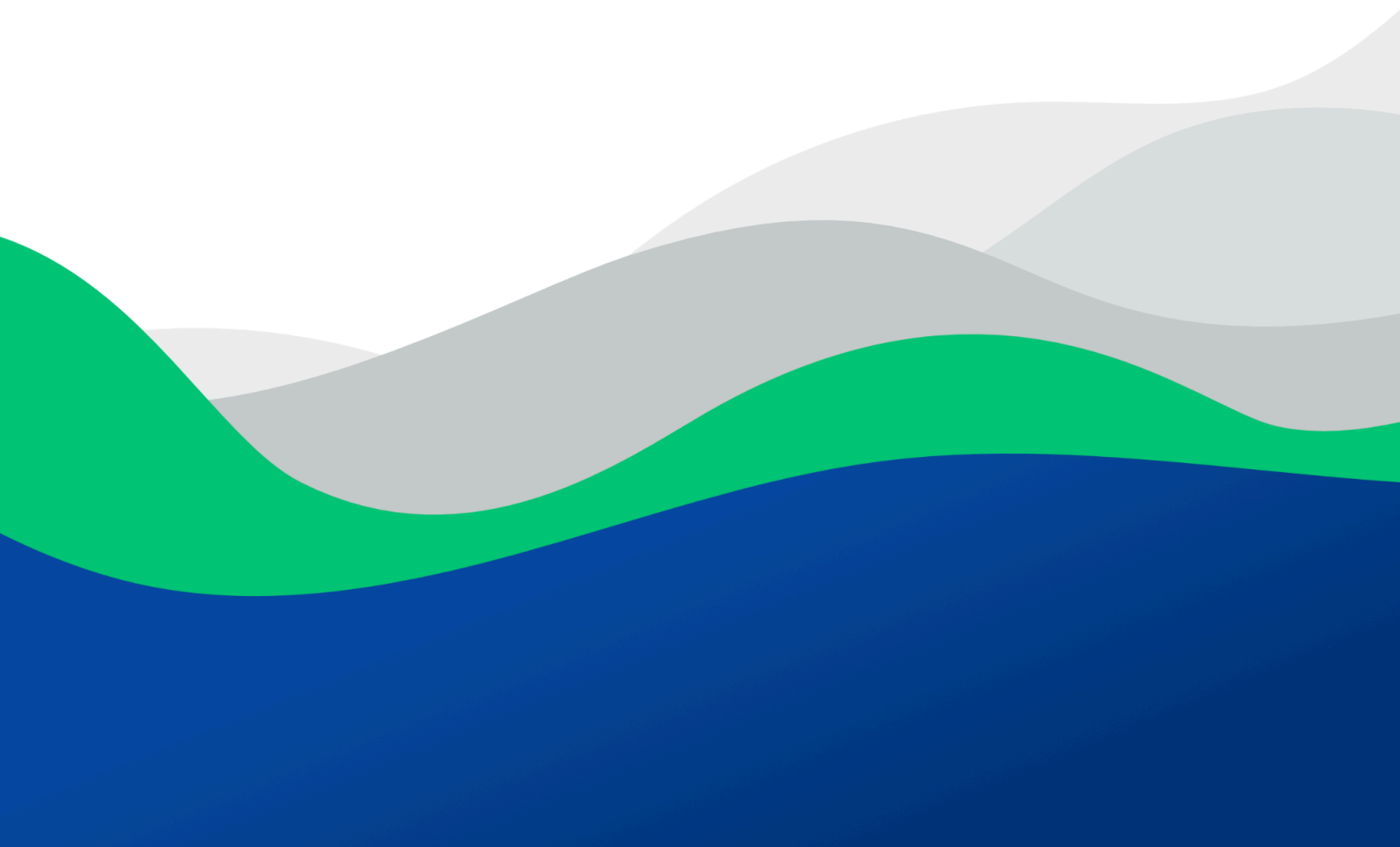
Rublon

Multi-Factor Authentication Buyer's Evaluation Guide



Rublon

Table of Contents

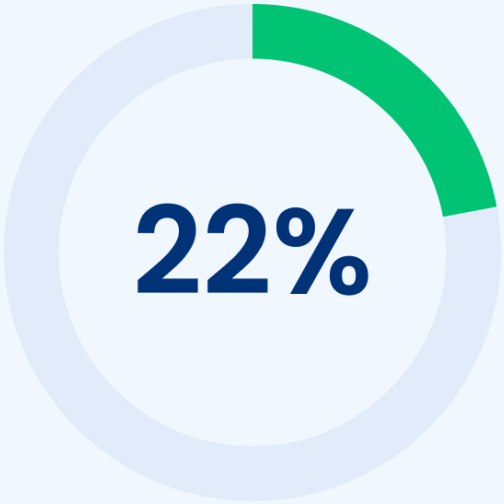
1. Introduction **3**
1.1. Why Choosing the Right MFA Solution Matters 3
1.2. How to Use This Guide 4
2. Define What Your Organization Needs **5**
2.1. Start With Business and Security Requirements 5
2.2. Inventory Users, Applications, and Access Paths 9
2.3. Avoiding Partial MFA Coverage 10
3. Evaluate Security Effectiveness **11**
3.1. Evaluate the Strength of Authentication Methods 11
3.2. Prioritize Phishing-Resistant MFA 12
3.3. Assess Protection Against MFA Fatigue and Social Engineering 13
3.4. Examine Access Policies and Enforcement 14
3.5. Verify Logging, Monitoring, and Incident Response Capabilities 15
4. Evaluate Infrastructure and Integration Coverage **16**
4.1. Protect Cloud, On-Premises, and Hybrid Environments 16
4.2. Evaluate Integration Breadth 18
4.3. Check Protocol and Identity Source Compatibility 20
4.4. Evaluate Custom and Legacy Application Support 21
5. Evaluate User and Administrator Experience **22**
5.1. Authentication Experience 22
5.2. Enrollment and Authenticator Lifecycle 24
5.3. Administrative Experience and Delegation 24
6. Evaluate Deployment and Operations **25**
6.1. Deployment Architecture 25
6.2. Time to Deploy and Time to Security 26
6.3. Availability, Resilience, and Recovery 27
6.4. Maintenance and Upgrade Requirements 28
7. Evaluate Commercial and Vendor Factors **29**
7.1. Calculate the Total Cost of Ownership 29
7.2. Security Assurance, Privacy, and Compliance 31
7.3. Evaluate the Vendor, Not Only the Product 32
8. Run a Meaningful Proof of Concept **33**
8.1. Build an MFA Proof-of-Concept Plan 33
8.1.1. Define Success Criteria 33
8.1.2. Use Representative Scenarios 33
8.1.3. Collect Evidence 34

8.2. MFA Vendor Evaluation Scorecard 34
8.3. Vendor Evaluation Worksheet 36
8.4. Warning Signs When Evaluating an MFA Vendor 36
9. How Rublon MFA Addresses the Evaluation Criteria 37
9.1. Rublon MFA at a Glance 37
9.2. Protecting the Organization's Critical Access Paths 38
9.3. Authentication Methods for Different Users and Risk Levels 39
9.4. Centralized Policies, Administration, and Visibility 40
9.5. Deployment Across Existing Infrastructure 41
9.6. Operational Value and Total Cost Considerations 42
9.7. Support, Documentation, and Security Assurance 42
10. Conclusion and Buyer Resources 43
10.1. Final MFA Buyer's Checklist 43
10.1.1. Scope and Coverage 43
10.1.2. Authentication and Policy 43
10.1.3. Operations and Resilience 44
10.1.4. Commercial and Vendor Review 44
10.1.5. Deployment Readiness 44
10.2. Next Steps 45

1. Introduction

1.1. Why Choosing the Right MFA Solution Matters

Deploying a multi-factor authentication (MFA) solution does not automatically create effective organization-wide protection. Security depends on which access paths are covered, which methods are permitted, how policies are enforced, and how the platform handles recovery, exceptions, and infrastructure failures.



Why is MFA important?

Credential theft remains a common route into business systems. Attackers obtain passwords through phishing, infostealer malware, credential stuffing, social engineering, and password reuse. Compromised credentials were an initial access vector in **22%** of the breaches reviewed in Verizon's 2025 Data Breach Investigations Report. Verizon's research also shows that credential abuse can represent a substantial share of daily authentication traffic.

Attackers increasingly target the authentication process itself. Adversary-in-the-middle phishing can relay passwords and one-time passcodes to a legitimate service. Push bombing attempts to pressure users into approving fraudulent requests. Stolen session tokens may allow attackers to reuse an authenticated session without completing MFA again.

The US National Institute of Standards and Technology (NIST) does not classify manually entered one-time codes as phishing-resistant because they can be captured and relayed. The Cybersecurity & Infrastructure Security Agency (CISA) recommends moving toward FIDO and WebAuthn authentication where possible.

© 2026 Rublon

A basic MFA deployment may satisfy a narrow audit requirement while leaving important gaps. Cloud applications may be protected while VPNs, privileged Windows logons, Linux servers, or legacy systems remain password-only. Weak recovery procedures and poorly controlled authenticator enrollment can further undermine otherwise strong authentication.

An unsuitable MFA platform also creates operational costs. Complex enrollment increases help desk demand. Limited integration support leads to parallel tools. Poorly matched methods cause resistance and workarounds. Hardware, telephony, professional services, infrastructure, maintenance, and internal labor can make a seemingly inexpensive product costly over time.

The right solution must balance security, coverage, usability, deployment effort, resilience, and cost. Its purpose is not merely to add another prompt, but to reduce account takeover risk without creating new blind spots or unnecessary friction.

1.2. How to Use This Guide

An MFA decision affects security architecture, infrastructure operations, compliance, user productivity, support workload, and long-term cost. Different stakeholders should evaluate the same platform from their own operational perspective:

- CISOs and security leaders can assess identity risk reduction.
- IT directors, infrastructure administrators, and network teams can focus on integration, deployment, resilience, and maintenance.
- IAM teams can examine identity sources, methods, policies, and authenticator lifecycle management.
- Compliance and risk teams should review auditability, data protection, and regulatory support.
- Procurement teams can compare licensing, implementation costs, support terms, and additional charges.
- Managed service providers should also consider scalability, delegated administration, and tenant separation.

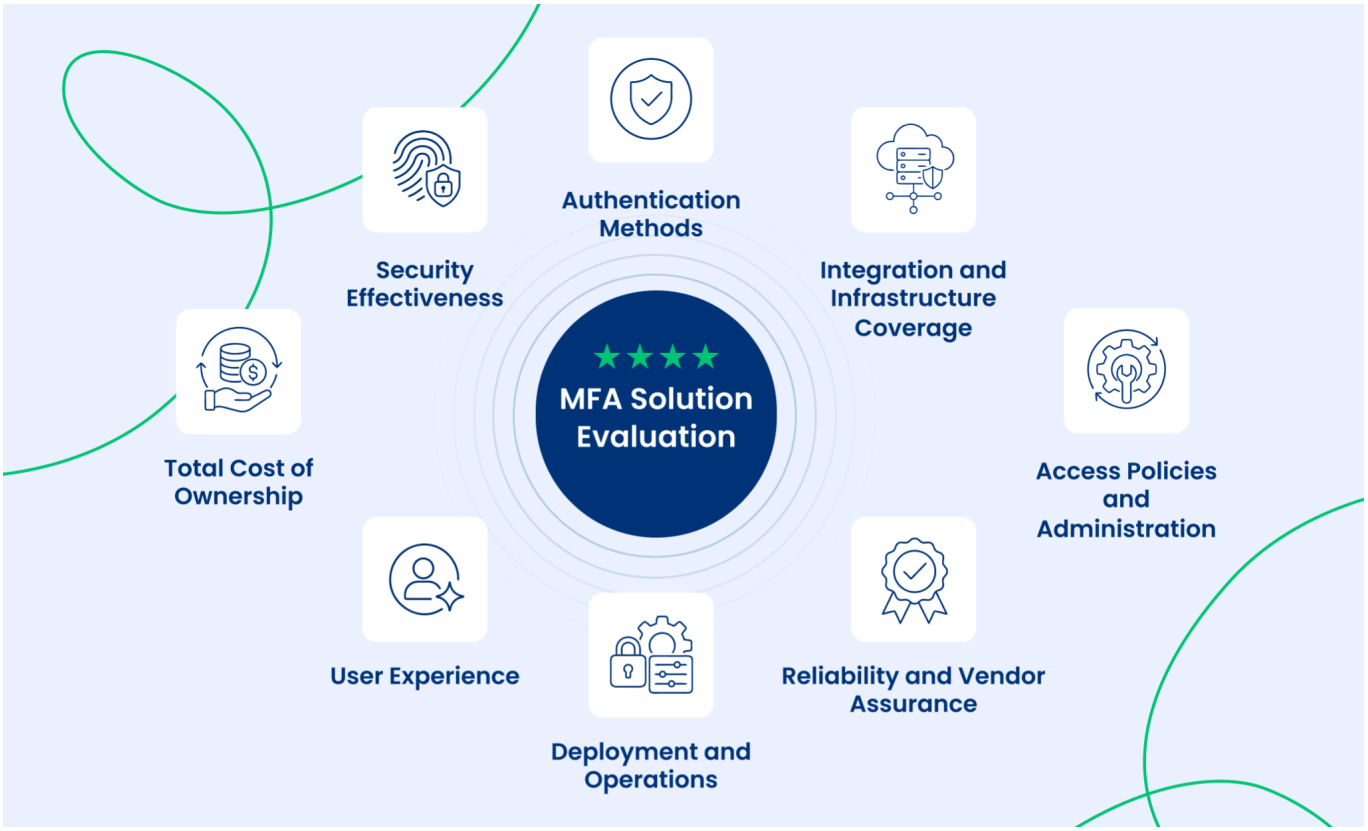
The evaluation covers eight connected areas:

- Security effectiveness
- Authentication methods
- Integration and infrastructure coverage
- Access policies and administration

© 2026 Rublon

- User experience
- Deployment and operations
- Reliability and vendor assurance
- Total cost of ownership

These areas should be assessed together. Strong authentication has limited value if important systems remain unprotected. Broad integration coverage is less attractive if administration becomes fragmented or users cannot complete authentication reliably.



MFA requirements should reflect the combined perspective of security, IT, compliance, and the organization's operational needs.

2. Define What Your Organization Needs

2.1. Start With Business and Security Requirements

Begin with the problem the organization needs to solve, not with a vendor feature list.

© 2026 Rublon

The primary objective may be reducing account takeover risk, meeting regulatory or cyber insurance requirements, fulfilling a customer contract, or addressing findings from a security assessment. Operational drivers may include remote work, privileged administration from multiple locations, or several incompatible MFA products deployed by different teams.

Scope the evaluation around the real environment. Cloud applications are only part of the picture. VPNs, Windows and Linux systems, network devices, Remote Desktop Services, administrative tools, and legacy applications may still depend on passwords.

Privileged identities require particular attention. Administrators, help desk personnel, infrastructure teams, and external service providers may be able to modify security settings, access backups, create accounts, or disable controls. Identify these users, the systems they can reach, and the authentication strength their privileges require.

The strategy should also support future change. Traditional methods may remain necessary during rollout, while FIDO2 security keys and passkeys are introduced first for high-risk users and expanded gradually.

Visibility is another core requirement. Security and compliance teams need reliable records of authentication attempts, policy decisions, authenticator changes, and administrative actions. The platform should show who attempted access, which method was used, and why access was allowed or denied.

A written requirements baseline gives security, IT, compliance, and procurement teams a shared definition of success.

© 2026 Rublon



Questions to Ask:

- Which workforce, administrative, service, contractor, partner, and external identities need protection?
- Which systems process sensitive, regulated, financial, or operationally critical data?
- Which applications and access paths still rely only on passwords?
- Which users require stronger authentication?
- Which systems cannot be migrated to a modern identity provider?
- Are remote workers, third-party support teams, and users without corporate devices included?
- Which regulatory, insurance, and contractual requirements apply?
- What evidence must be available for investigations and audits?

MFA Requirements Worksheet

Area	Requirement	Priority	Current State	Target State / Gaps
Users and Identities	Workforce users	**High / Medium / Low**		
	Administrators and	**High / Medium /**		

© 2026 Rublon

	privileged users	**Low**		
	Contractors and third parties	**High / Medium / Low**		
	Remote users	**High / Medium / Low**		
	Service and emergency accounts	**High / Medium / Low**		
Applications and Access Paths	Cloud applications	**High / Medium / Low**		
	VPN and remote access	**High / Medium / Low**		
	Windows workstations and servers	**High / Medium / Low**		
	Linux servers and SSH	**High / Medium / Low**		
	Network and infrastructure access	**High / Medium / Low**		
	Legacy and custom applications	**High / Medium / Low**		
Risk and Authenticatio n Strength	Phishing-resistant MFA for privileged access	**High / Medium / Low**		
	Stronger MFA for sensitive applications	**High / Medium / Low**		
	Methods for users without smartphones	**High / Medium / Low**		
	Backup and recovery methods	**High / Medium / Low**		
	Secure authenticator enrollment and replacement	**High / Medium / Low**		

© 2026 Rublon

Compliance and Auditability	Regulatory and contractual requirements	**High / Medium / Low**		
	Cyber insurance requirements	**High / Medium / Low**		
	Authentication and administrative logging	**High / Medium / Low**		
	Log retention and export	**High / Medium / Low**		
	Evidence required for audits	**High / Medium / Low**		

2.2. Inventory Users, Applications, and Access Paths

An MFA inventory should connect each identity type to the systems it can access, the route used, and the privilege involved. This exposes inconsistent controls and password-only paths.

Include workforce users, administrators, privileged accounts, contractors, partners, and support providers. Record both local and remote access. A VPN login differs from direct Windows access, while Remote Desktop Services may involve a web portal, gateway, and session host. Linux administration may use SSH, and network teams may authenticate to firewalls, switches, wireless infrastructure, and other appliances.

Group applications by architecture and integration model:

- Cloud and SAML applications
- External and internal web applications
- Windows workstations and servers
- Remote Desktop Services
- Linux servers
- VPNs and network devices
- Administrative consoles and infrastructure tools
- Custom and legacy applications

For each item, record the users, identity source, login method, network location, owners, data sensitivity, privilege level, and current authentication controls. Include dependencies and

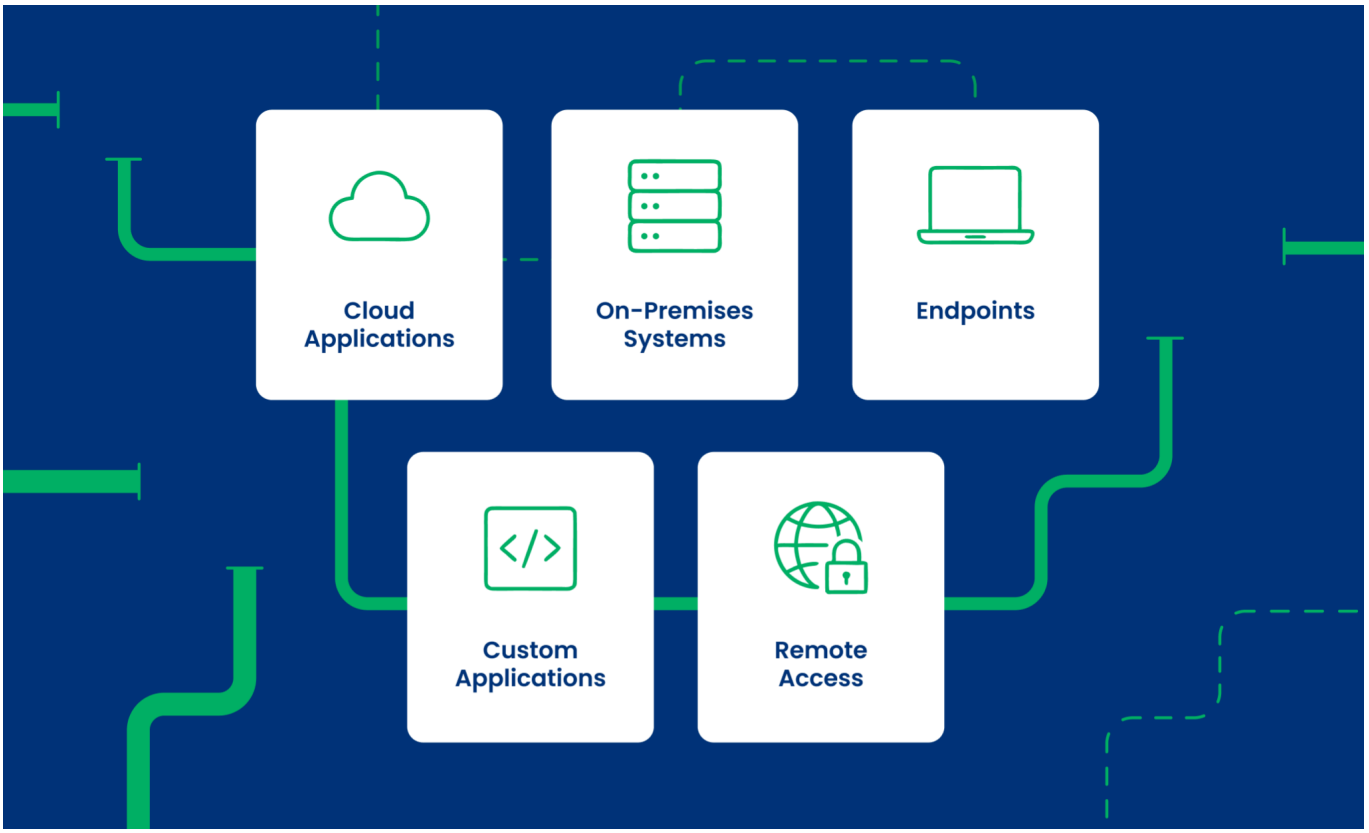
© 2026 Rublon

alternate entry points. An application may appear protected while its administrative interface, maintenance account, or direct server login remains outside MFA.

2.3. Avoiding Partial MFA Coverage

Partial coverage creates a false sense of security. Protecting SaaS applications does not secure local systems. MFA at the VPN does not protect a separate management interface. Remote Windows logons may be protected while local logons remain password-only. Modern applications may receive strong controls while older systems containing equally sensitive data are excluded.

Follow access from the user to the final resource. Every realistic path to a critical system should be identified and assessed.



Effective MFA coverage should extend across all major areas of the environment, not just cloud applications.

© 2026 Rublon

3. Evaluate Security Effectiveness

3.1. Evaluate the Strength of Authentication Methods

Assess each method by the factor it proves, how that factor is verified, and which attacks it resists.

The traditional factor categories are:

- **Knowledge:** something the user knows, such as a password or PIN
- **Possession:** something the user has, such as a registered phone or security key
- **Inherence:** a biometric characteristic, such as a fingerprint or facial feature

Two prompts do not necessarily create MFA. Two passwords, a password followed by a security question, or codes delivered through the same compromised channel may add friction without proving independent factors.

Common methods offer different security and operational characteristics:

- **Mobile Push** is fast and familiar. Its strength improves with number matching, contextual information, request limits, and protection against repeated prompts.
- **TOTP passcodes** work without mobile connectivity and support many applications. Because users enter them manually, attackers can capture and relay them.
- **FIDO2 security keys** use public-key cryptography and verify the legitimate service domain. They provide strong phishing resistance but require distribution, spare stock, and replacement procedures.
- **FIDO2 passkeys** use public-key cryptography and may be stored on a phone, computer, security key, or supported credential provider. They can be device-bound or synchronized.
- **Platform authenticators**, such as Windows Hello, authorize a passkey through a local PIN, fingerprint, or facial recognition.
- **Hardware OTP tokens** support users without smartphones and environments with limited connectivity, but their displayed codes remain phishable.
- **SMS, phone, and email methods** broaden accessibility and can support transition or recovery. Their security depends on external communication and account recovery channels.
- **RFID authentication** may suit shared workstations and workplaces that already issue employee cards. Cloning risk, PIN use, and rapid revocation should be assessed.
- **QR codes** allow for convenient scanning with the user's phone.
- **Recovery methods** such as bypass codes, temporary access, and break-glass accounts should be limited, logged, and tightly authorized.

© 2026 Rublon

No single method suits every user and system. A balanced portfolio combines phishing-resistant authentication for high-risk access, convenient methods for routine use, alternatives for users without smartphones, and controlled recovery options.

3.2. Prioritize Phishing-Resistant MFA

Traditional MFA can stop an attacker who has only a stolen password, but it may not stop real-time phishing. A fraudulent site can collect a password and one-time passcode, then relay both to the legitimate service. Approval-based methods can also be abused through misleading or repeated prompts.

Phishing-resistant authentication reduces reliance on user judgment. NIST defines phishing resistance as the ability to prevent authentication secrets and valid authenticator outputs from being disclosed to an impostor verifier. Manually entered one-time passcodes and push notifications do not meet this definition.

FIDO2 provides phishing resistance through public-key cryptography and WebAuthn. During registration, the authenticator creates a unique credential for a specific service. The private key remains with the authenticator, while the service stores the public key.

Origin binding prevents the credential from being used by a visually similar phishing site on another domain. The authenticator and service verify the relying party and request origin before completing authentication.

FIDO2 authenticators include:

- **Security keys:** separate physical authenticators using interfaces such as USB or NFC
- **Device-bound passkeys:** credentials retained on a specific device or authenticator
- **Synced passkeys:** credentials made available across devices through a protected synchronization service

A PIN or biometric can provide local user verification. This information remains on the authenticator and is not sent to the application as a reusable secret.

© 2026 Rublon



Recommended Use Cases

Phishing-resistant MFA should be prioritized for:

- Administrators and infrastructure teams
- Privileged and emergency accounts
- Remote access to internal systems
- Applications containing regulated or highly sensitive data
- Employees frequently targeted by phishing or impersonation
- Financial approvals and payment authorization

Migration can be gradual. Begin with high-risk users and critical systems, then expand as application support and authenticator availability improve. Weaker recovery methods must not undermine the stronger primary method.

3.3. Assess Protection Against MFA Fatigue and Social Engineering

Push bombing uses repeated authentication requests to confuse, pressure, or frustrate a user into approving one. A single accidental approval is usually sufficient for the attacker to gain access.

Number matching reduces this risk by requiring the user to enter or select a value shown on the login screen. Context such as the application, approximate location, device, and request time can make fraudulent attempts easier to recognize.

These controls should be combined with rate limits, temporary lockouts, and restrictions on repeated prompts. Administrators need a defined response after several failed, rejected, or expired requests. A denied request should not be replaced immediately by another identical prompt.

© 2026 Rublon

Authentication strength should also reflect application risk. Mobile Push may be appropriate for a routine service, while an administrative console, financial system, or production server may require a FIDO credential.

Logs should expose the full sequence of requests, denials, timeouts, method changes, and approvals. Several rejected attempts followed by one successful approval may indicate an active attack.

Users should be trained to reject and report unexpected prompts. The reporting process should identify who to contact and what action to take. Enrollment and replacement require similar protection. Identity verification, approval, logging, and user notification should prevent attackers from registering a new phone through social engineering.

Questions to Ask Vendors:

- How are repeated push attacks limited?
- Does Mobile Push support number matching and contextual information?
- Can methods be restricted by user, group, or application?
- Can stronger methods be required for sensitive systems?
- Are suspicious and rejected requests visible in logs?
- What happens after repeated failures or denials?
- How are new and replacement authenticators verified?

3.4. Examine Access Policies and Enforcement

Authentication policies should reflect application sensitivity, user privilege, access context, and operational risk. One organization-wide rule rarely provides appropriate protection for every scenario.

Per-application policies can require stronger authentication for payroll, administrative, financial, or production systems. User- and group-based policies can apply different controls to administrators, finance teams, executives, developers, contractors, and external users.

Method restrictions prevent weaker options from undermining stronger controls. A platform should be able to require FIDO2 for selected systems, disable telephony methods for privileged users, and reserve temporary codes for authorized recovery.

Step-up MFA requests stronger verification when risk increases, such as before changing security settings, accessing regulated data, or approving a financial transaction.

© 2026 Rublon

Remembered-device policies should define eligibility, duration, and revocation conditions. High-risk users and applications may need to be excluded. Network and location signals can influence authentication requirements, but should not be the sole basis for trust.

Emergency accounts should be limited, strongly protected, tested, and monitored. Any bypass must be temporary and audited.

Outage behavior must also be explicit:

- **Fail-open** preserves access but allows password-only authentication.
- **Fail-closed** maintains MFA enforcement but may block legitimate users.

The appropriate model depends on system criticality, business impact, and available recovery options.

Learn more in Rublon Guide to Business Continuity Preparedness.

Finally, verify policy consistency across cloud applications, VPNs, Windows logons, and legacy systems. A control visible in the console may not be available through every integration.

3.5. Verify Logging, Monitoring, and Incident Response Capabilities

Authentication logs should allow administrators to reconstruct each attempt. Essential fields include the user, protected resource, time, source IP address, method, policy result, and final status. The platform should distinguish successful authentication, user rejection, timeout, policy denial, and technical failure.

Integration context is equally important. A cloud application login, VPN connection, Windows logon, and Linux SSH attempt represent different access paths.

Authenticator events should record phone enrollment, security key registration, replacement, revocation, and temporary access. Records should identify who initiated the action and when. Users should be notified of significant changes where possible.

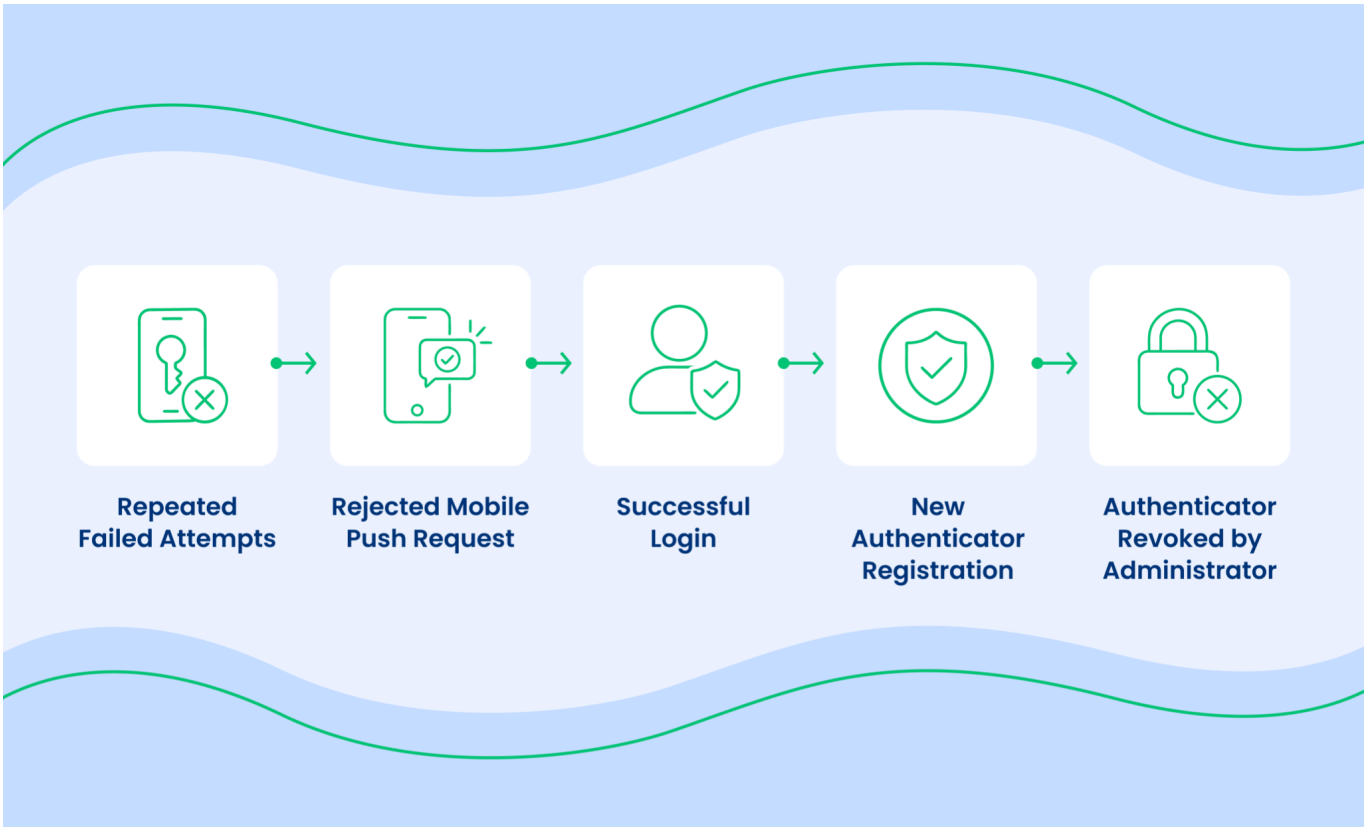
Administrative audit logs should capture changes to policies, applications, roles, user status, directory synchronization, and security settings. This helps determine whether a control failed, was misconfigured, or was deliberately changed.

Logs should be searchable, retained for an appropriate period, and exportable to a SIEM platform. During suspected compromise, administrators should be able to revoke authenticators, clear remembered-device trust, disable methods, and force secure re-enrollment. Response actions must also be audited.

© 2026 Rublon

Questions to Ask:

- What is captured for every attempt?
- Can rejection, timeout, policy denial, and technical failure be distinguished?
- Are enrollment, replacement, and revocation events logged?
- Are all administrative changes audited?
- How long are records retained?
- Can logs be exported automatically to a SIEM platform?
- Can authenticators and remembered devices be revoked quickly?
- Is the evidence sufficient for investigations and audits?



Complete MFA visibility requires logging both authentication attempts and events throughout the authenticator lifecycle.

4. Evaluate Infrastructure and Integration Coverage

4.1. Protect Cloud, On-Premises, and Hybrid Environments

Most organizations use several identity and authentication models. Cloud services may rely on federation, while internal applications still use Active Directory, LDAP, RADIUS, local accounts, or

© 2026 Rublon

older login flows. Remote users connect through VPNs and gateways, and administrators may sign in directly to Windows or Linux systems.

Cloud-only MFA can therefore leave local workstations, servers, network devices, and legacy applications outside the security boundary.

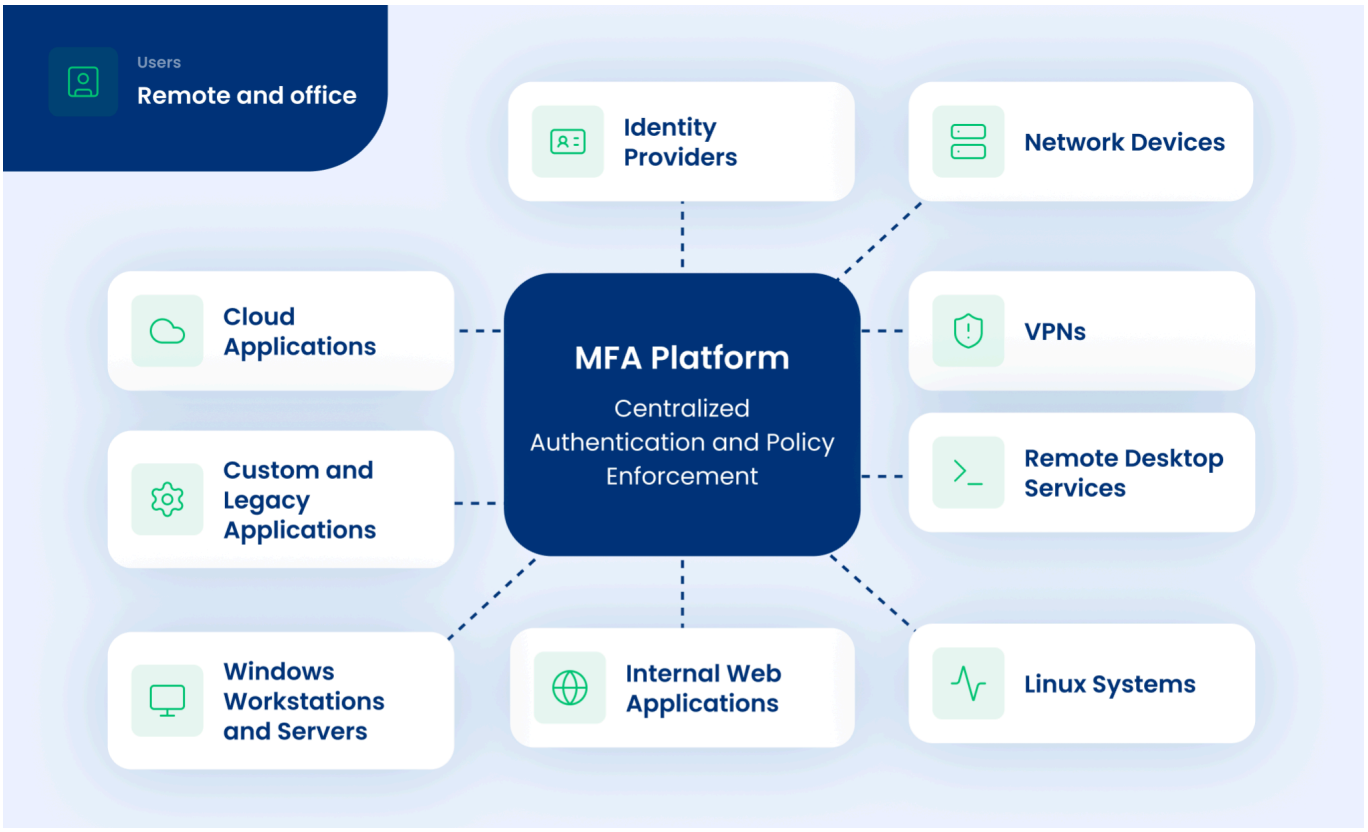
A suitable platform should provide centralized policy and visibility while supporting different enforcement mechanisms. It should coexist with existing directories and identity providers, use established protocols where appropriate, and offer dedicated connectors when deeper integration is required.

Modernization should be gradual. New applications may adopt SAML or OpenID Connect while older systems remain protected through LDAP, RADIUS, local connectors, or gateways. Supporting both states avoids duplicated administration and fragmented security tools.

Hybrid coverage also improves investigation. A shared view of cloud, VPN, server, and endpoint authentication makes it easier to follow activity across access paths and identify inconsistent enforcement.

Evaluate specific applications, operating systems, protocols, and login scenarios. Broad claims of cloud or on-premises support are not enough.

© 2026 Rublon



A centralized MFA platform can protect cloud, on-premises, and remote resources across a hybrid environment.

4.2. Evaluate Integration Breadth

Integration breadth should be measured against the organization's actual access paths, not the number of logos on a vendor page.

Remote access systems such as VPNs, firewalls, wireless controllers, and network appliances often use RADIUS or LDAP. Compatibility must be verified for the exact product version, authentication mode, and network architecture. Systems using the same protocol may handle challenges, timeouts, and secondary authentication differently.

Windows coverage should be assessed separately for:

- Local Windows Logon
- Remote Desktop Protocol
- Remote Desktop Gateway
- Remote Desktop Web Access and Web Client
- Other Microsoft services and administrative paths

© 2026 Rublon

For Linux, verify PAM and SSH support, including interactive sessions, privileged access, SSH keys, automation accounts, and recovery scenarios.

Cloud and web applications may use native connectors, SAML, OAuth 2.0, OpenID Connect, access gateways, reverse proxies, or other components. The selected approach should preserve application behavior and remain supportable after upgrades.

Custom applications may require APIs or SDKs. Review supported languages, examples, error handling, release policy, backward compatibility, and maintenance ownership.

Administrative consoles, virtualization platforms, backup systems, hypervisors, databases, and security tools should not be overlooked. Their privileges often make them higher-risk than ordinary workforce applications.

Identity compatibility may include Active Directory, Microsoft Entra ID, LDAP directories, RADIUS-based services, SAML identity providers, and local identities.



Evaluation Criteria

For each integration, determine:

- Whether it uses a native connector, protocol, federation, gateway, or custom code
- Supported application and operating system versions
- Available configuration and troubleshooting documentation
- Supported authentication methods
- Local component and certificate requirements
- Upgrade and backward-compatibility policies
- High-availability options
- Maintenance ownership
- User experience
- Known limitations and unsupported scenarios

Method support may differ by integration. FIDO2 may be available for a web application while a particular

© 2026 Rublon

VPN supports only Mobile Push or TOTP. Record support per access path.

4.3. Check Protocol and Identity Source Compatibility

An MFA platform should work with the existing identity architecture without requiring an immediate directory migration.

Active Directory support should cover user discovery, group mapping, account status changes, and domain environments. Microsoft Entra ID may operate as the main cloud identity source or coexist with Active Directory.

LDAP and LDAPS provide compatibility with enterprise directories. Verify schema requirements, search bases, attribute mapping, service account permissions, certificate validation, and directory outage behavior. LDAPS should be preferred where supported.

RADIUS is commonly used by VPNs, firewalls, wireless systems, and network devices. Because it is an authentication protocol rather than a directory, confirm how primary credentials are validated and how MFA challenges, timeouts, and failover are handled.

SAML supports federated authentication for many cloud and web applications. OAuth 2.0 is an authorization framework, while OpenID Connect adds authentication. Vendor claims should identify supported roles and flows.

Directory synchronization should import users and groups, apply changes promptly, and report failures. Multiple directories and domains require reliable identity matching. Duplicate usernames, changing email addresses, and mergers can otherwise create conflicts or incorrect account linking.

Provisioning and deprovisioning are equally important. Disabled or deleted users should lose access promptly.

Questions to Ask:

- Can existing identity sources be retained?
- Is a proprietary directory required?
- Can multiple directories, domains, and identity providers coexist?
- How are identity conflicts resolved?
- How quickly are changes synchronized?

© 2026 Rublon

- What happens when an account is disabled, deleted, or moved?
- Are synchronization failures reported?
- Which system remains authoritative?



A centralized MFA platform can work with existing identity sources while centralizing key identity management functions.

4.4. Evaluate Custom and Legacy Application Support

Older and internally developed legacy systems are often the hardest to protect and the easiest to omit.

First assess generic protocols such as RADIUS, LDAP, SAML, OAuth 2.0, and OpenID Connect. Standards can reduce development effort, but compatibility must be tested against the actual authentication flow.

APIs and SDKs can integrate MFA into custom sign-in processes, privileged actions, or transaction approvals. Evaluate supported languages, sample code, versioning, error handling, and the security responsibilities assigned to developers.

© 2026 Rublon

Reverse proxies and access gateways can protect web applications that cannot be modified. Review session handling, certificates, redirects, client IP preservation, availability, and upgrade compatibility.

If a system cannot be changed, MFA may need to be enforced at another layer, such as the VPN, operating system logon, remote access gateway, network boundary, or administrative interface.

Estimate custom development and maintenance before selection. Assign responsibility for building, testing, and updating the integration.

Standards-based options, documented APIs, and separable integration layers can reduce vendor lock-in, while native integrations may provide deeper control where justified.

Do not evaluate only the applications that are easy to protect. Evaluate the systems that would remain password-only after deployment.

5. Evaluate User and Administrator Experience

5.1. Authentication Experience

Authentication should be assessed by the number of user actions, completion time, clarity, and consistency across applications.

A routine login may justify a low-friction method, while privileged access may require stronger verification. Repeated prompts that add little security encourage users to rush or seek workarounds.

Experiences will differ across browsers, VPN clients, Windows logons, SSH sessions, and mobile devices, but users should always understand what is expected. Error messages should distinguish rejection, expiration, policy restriction, incorrect passcodes, service outages, and technical failures.

Test both mobile and desktop workflows. Some users will lack corporate smartphones, work in restricted areas, or decline to use personal devices. Suitable alternatives should be available without weakening protection for everyone.

Connectivity requirements must be understood. TOTP and some local authenticators work without mobile data, while other methods require access to a cloud service.

Shared workstations must keep identities, sessions, and remembered-device status separate. Accessibility should cover enrollment, authentication, errors, and recovery.

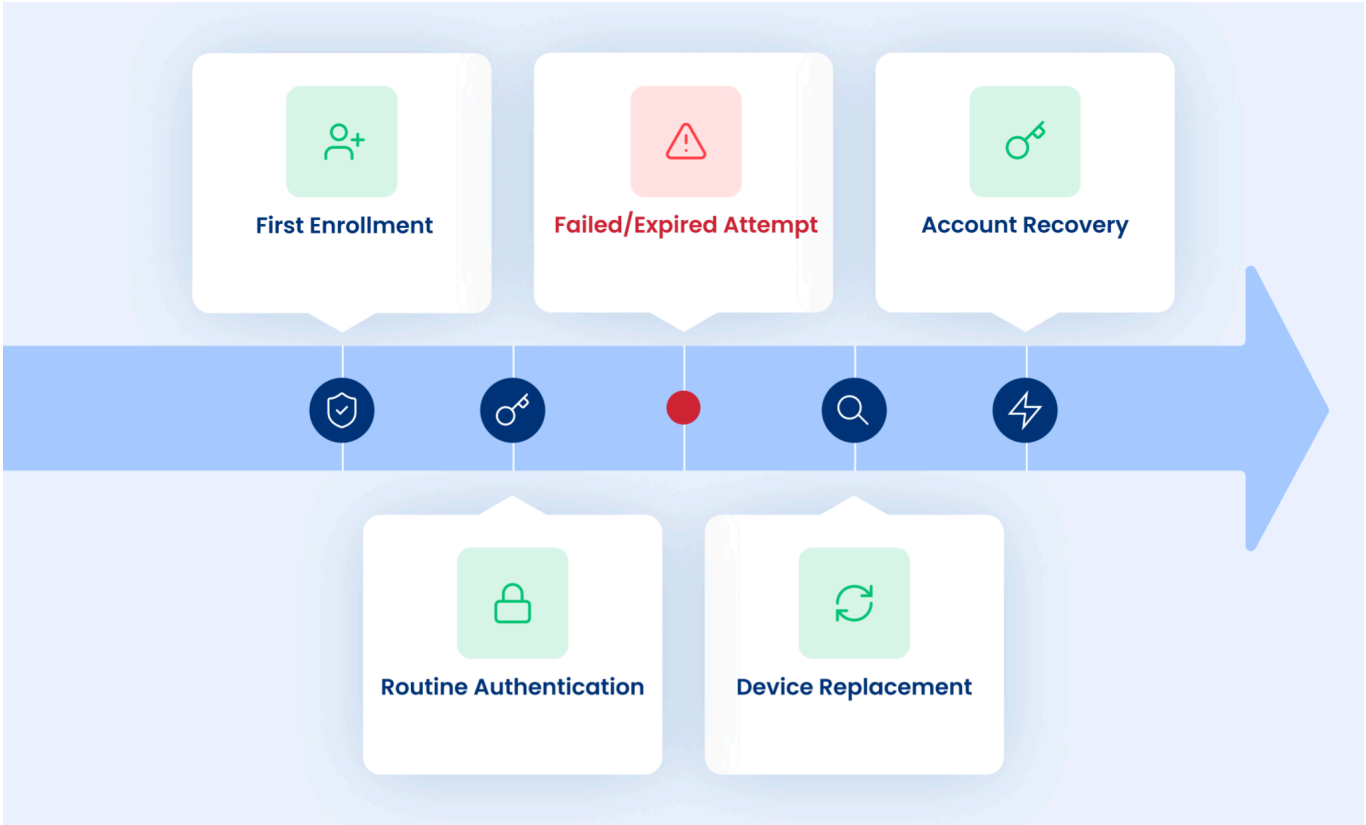
© 2026 Rublon

Multiple authenticators help improve resilience. A user may use a passkey daily, retain a security key as backup, and use another approved method for specific systems.

Test During the Proof of Concept:

- First authentication after enrollment
- Routine authentication across several applications
- User rejection
- Request expiration
- Loss of connectivity
- Migration to a new phone
- Loss of a security key
- A locked or disabled account
- The full recovery process

Record completion time, user confusion, support intervention, and differences between integrations.



The authentication journey extends from initial enrollment through routine use, failed attempts, device replacement, and account recovery.

© 2026 Rublon

5.2. Enrollment and Authenticator Lifecycle

Enrollment links an identity to an authenticator. Weak enrollment can assign a strong credential to the wrong person.

Administrator-led enrollment offers greater control for privileged users, security keys, hardware tokens, and regulated environments, but may not scale well. Self-enrollment reduces administrative work, provided invitations are protected, time-limited, and tied to reliable identity verification.

Bulk enrollment should accelerate deployment without automatically trusting unverified devices. Administrators need visibility into completed, pending, and exempt accounts.

Users may need multiple authenticators for resilience. Policies should define permitted types, registration limits, and backup requirements for high-risk users.

Replacement and recovery require strong verification. When a phone or key is lost, the old authenticator should be revoked, recent activity reviewed, and the user notified. Temporary access methods should expire automatically and remain fully audited.

Offboarding should revoke authenticators, remembered devices, exceptions, and recovery methods promptly. Directory synchronization can support this process, but delays and failure handling must be tested.

Controls against unauthorized registration should include limited administrative permissions, approval, notifications, and complete audit records.

5.3. Administrative Experience and Delegation

A centralized console should manage users, applications, policies, authenticators, and security events without requiring administrators to reconstruct information from multiple tools.

User records should show status, applications, groups, authenticators, policy assignments, and recent activity. Application owners need similar visibility into protected resources and assigned policies.

Administrative access should follow least privilege. Platform owners may require full control, while help desk personnel should be able to revoke a lost authenticator or issue temporary access without changing organization-wide policies.

Separation of duties may distinguish security, infrastructure, support, billing, and application management. Administrative units can further restrict access by department, region, subsidiary, customer, or tenant.

© 2026 Rublon

Bulk operations, search, and filtering should support large deployments. Administrators need to locate accounts by status, identity source, group, authenticator type, and enrollment state.

Troubleshooting should explain which policy applied, which method was attempted, and why authentication failed. Directory errors, connector failures, timeouts, rejections, and policy denials require different responses.

All administrative actions should be attributable and timestamped. Administrative accounts themselves should use strong MFA. Multi-tenant environments also require clear reporting boundaries and safeguards against changes to the wrong organization.

Questions to Ask:

- How many consoles are required?
- Can permissions be limited by role, task, and scope?
- Can help desk staff perform recovery without changing policies?
- Are bulk operations available?
- Can administrators identify the cause of a failed authentication?
- Are all administrative changes audited?
- Is MFA required for administrators?
- Are tenants and customer data isolated?

6. Evaluate Deployment and Operations

6.1. Deployment Architecture

Deployment architecture defines where credentials are processed, where authentication decisions are made, and how protected systems communicate with the MFA platform.

A cloud-managed platform can centralize configuration, policies, administration, and logs. Local connectors may still enforce MFA for VPNs, directories, Windows systems, Linux servers, and internal applications.

In a hybrid model, central policies are combined with components deployed near protected resources. The architecture should document each component's function, processed data, and behavior when connectivity is interrupted.

Review:

- Outbound and inbound network requirements
- Firewall rules and exposed ports
- Proxy compatibility

© 2026 Rublon

- TLS certificates and private key storage
- Certificate renewal and expiration monitoring
- Service account permissions and rotation
- Data flows, processing locations, retention, and residency
- Hosting providers, subprocessors, and backups
- Component hardening and patching

Outbound connections are generally easier to secure, while inbound access should be justified and minimized.

Service accounts should receive only the permissions required for their role. Directory access, integration, logging, and administration may require separate identities.

Local components should have unnecessary services disabled, restricted administrative access, monitored logs, and defined update procedures.

The management and authentication planes should also be separated. Loss of the administrative console should not automatically stop authentication, while compromise of one connector should not provide control over the entire platform.

Example: Read about Rublon MFA Architecture.

6.2. Time to Deploy and Time to Security

Time to security is the period between selecting the platform and achieving verified protection across important access paths.

Begin with discovery of users, identity sources, applications, protocols, dependencies, and exceptional scenarios. Incomplete discovery leads to delays and unsupported use cases during rollout.

The proof of concept should cover at least one critical system, one representative identity source, required methods, policies, logs, recovery, and outage behavior. A pilot should then include administrators, office users, remote workers, and users with unusual device or connectivity requirements.

Production deployment is safer in controlled phases. Each phase needs entry criteria, success measures, support ownership, and a tested rollback plan.

Communication should begin before enforcement. Users need enrollment dates, available methods, and support contacts. Administrators and support teams require training in troubleshooting, recovery, and escalation.

© 2026 Rublon

Measure adoption through enrollment rates, required method usage, protected applications, failed enrollments, exceptions, and password-only accounts.

Suggested Rollout Sequence:

1. Administrators and IT personnel
2. Remote access and VPN
3. Critical servers and privileged systems
4. Cloud and web applications
5. Remaining workforce applications
6. Legacy and exceptional use cases

Prioritize risk, not ease of deployment.



Effective MFA deployment progresses from discovery and proof of concept through production rollout and continuous optimization.

6.3. Availability, Resilience, and Recovery

MFA is part of the access path to critical systems. A disruption may either block legitimate users or reduce security by bypassing authentication.

© 2026 Rublon

Review the formal service level agreement, calculation method, exclusions, redundancy, geographic distribution, failover, backups, and recovery procedures.

Local proxies, gateways, connectors, and directory services may become single points of failure. Confirm whether redundant instances, load balancing, and configuration synchronization are supported.

Define authentication behavior during:

- Cloud service outages
- Connector failures
- Directory failures
- Network disruption
- Limited connectivity
- Planned maintenance

Offline authentication should specify supported users, methods, systems, validity periods, and audit requirements.

Monitoring should show whether components are reachable, correctly configured, and processing requests. Errors should distinguish directory, connector, network, and MFA service problems.

Fail-open and fail-closed decisions must be approved per system. Critical healthcare, safety, manufacturing, and emergency services may require different continuity models from ordinary applications.

Policies, connector configurations, certificates, service accounts, and secrets should be backed up and recoverable. Recovery ownership, objectives, dependencies, and testing frequency must be documented.

The vendor should maintain a status channel and provide timely incident updates, workarounds, and post-incident information where appropriate.

6.4. Maintenance and Upgrade Requirements

Identify which components the vendor maintains and which remain the customer's responsibility.

Cloud services may update automatically, while connectors, agents, gateways, and plugins require local maintenance. Automatic updates reduce routine work but may introduce compatibility risk. Manual updates provide control only when releases are tracked and installed promptly.

© 2026 Rublon

Confirm:

- Supported operating systems and application versions
- Support periods for older platforms
- Compatibility timelines for new releases
- Security patch and emergency update processes
- Backward compatibility
- Release and deprecation notices
- Maintenance window requirements

Upgrade testing should verify primary authentication, MFA, policies, logs, failover, and recovery. High-availability environments should support sequential updates where possible.

The responsibility model should cover cloud maintenance, local monitoring, connector updates, certificate renewal, protected application validation, and third-party dependencies.

7. Evaluate Commercial and Vendor Factors

7.1. Calculate the Total Cost of Ownership

The subscription price is only one component of MFA cost. Compare all expenses required to deploy, operate, support, and expand the platform.

Review how licenses are assigned, whether minimum commitments apply, and how contractors, seasonal users, service accounts, inactive users, and reassigned licenses are handled.

Separate method charges from the base subscription. SMS and phone authentication may generate usage costs based on frequency, destination country, and fallback behavior.

Hardware security keys and OTP tokens create purchasing, shipping, inventory, replacement, and spare-stock costs. Administrative time often exceeds the visible device price.

Additional cost categories include:

- Application and connector licenses
- Premium protocols or modules
- High-availability licensing
- Test and development environments
- Professional services
- Custom integration work
- Support plans
- Virtual machines and operating system licenses

© 2026 Rublon

- Databases, load balancers, certificates, monitoring, and backups
- Internal deployment and administration
- Training and communication
- Connector maintenance
- Help desk workload
- Replacement authenticators

Estimate support demand for lost devices, changed phones, forgotten PINs, failed enrollment, locked accounts, and temporary access.

The model should include growth through new users, offices, acquisitions, applications, identity sources, and authentication methods.

Three-Year TCO Model:

Use consistent assumptions for all vendors. Document uncertainty and test higher-usage and growth scenarios rather than relying on the lowest advertised price.

Cost Category	Year 1	Year 2	Year 3	Assumptions
Initial implementation				Architecture, integration, testing, migration
Annual subscription				Users, commitments, expected growth
Authentication delivery				SMS and phone use by country and volume
Hardware authenticators				Purchases, spares, shipping, replacement
Internal labor				Deployment, administration, training, support
Support and maintenance				Vendor support, infrastructure, updates, certificates
Expansion				Users, applications, components,

© 2026 Rublon

				services

7.2. Security Assurance, Privacy, and Compliance

Because an MFA provider sits in the authentication path and processes security-sensitive data, its own controls require evidence.

Independent certifications such as ISO/IEC 27001 and reports such as SOC 2 can provide assurance, but their scope matters. Verify the covered legal entity, services, audit period, auditor, and exclusions.

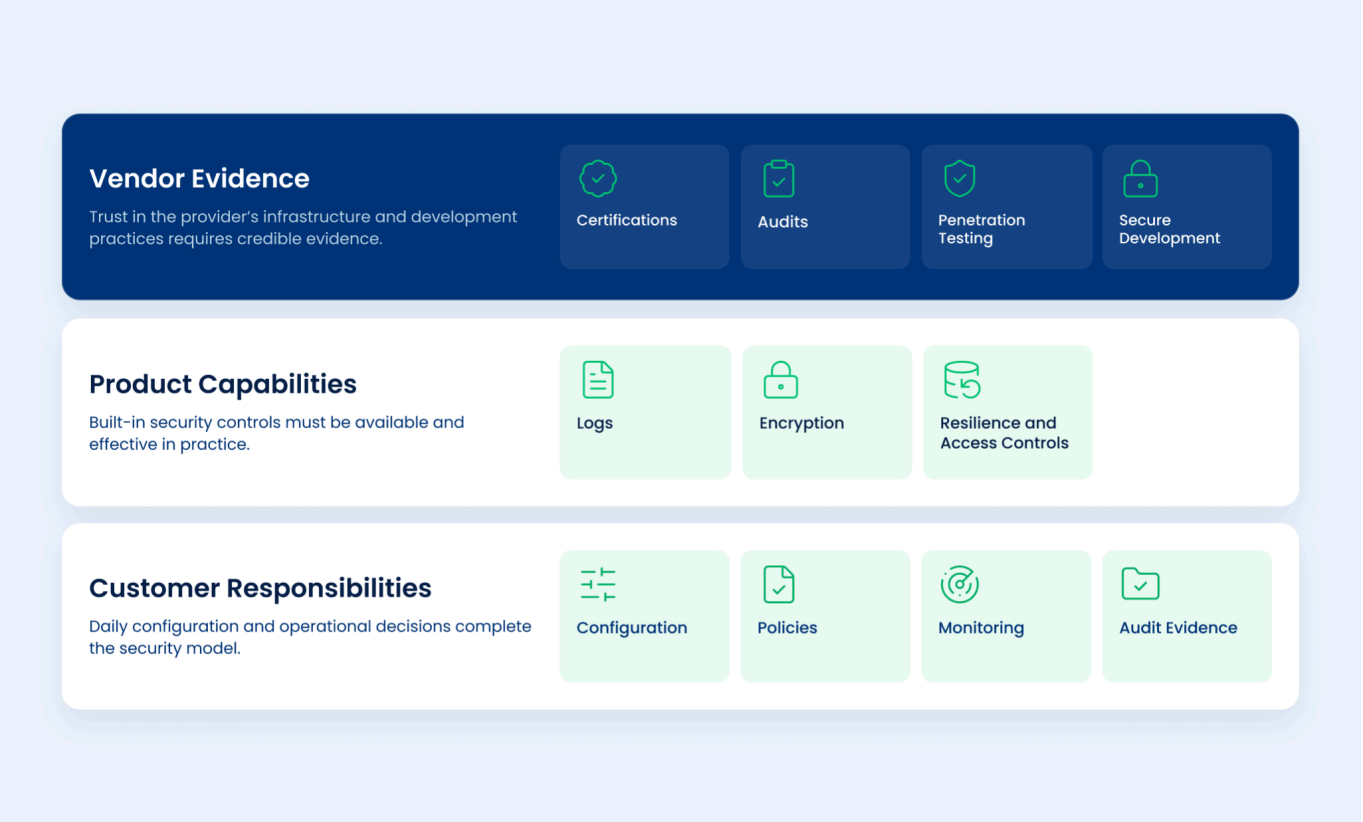
Review the vendor's:

- Information security management system
- Secure development lifecycle
- Source code and dependency review
- Internal and external penetration testing
- Vulnerability disclosure and remediation process
- Customer notification procedures
- Encryption in transit and at rest
- Key management
- Data retention and deletion
- Backup and recovery
- Business continuity

Data residency does not by itself establish compliance. Review processing locations, subprocessors, international transfers, support access, contractual safeguards, and breach notification terms.

Relevant requirements, regulations, standards, and guidance may include GDPR, NIS2, DORA, PCI DSS, HIPAA, the FTC Safeguards Rule, NIST guidance, cyber insurance conditions, and other compliance regulations and applicable obligations.

© 2026 Rublon



Effective MFA security assurance depends on vendor evidence, product capabilities, and sound customer practices working together.

7.3. Evaluate the Vendor, Not Only the Product

An MFA platform may remain in use for years while applications, operating systems, standards, and threats continue to change.

Assess the vendor's specialization in MFA and identity security, recent release history, product roadmap, and ability to explain integration and recovery risks.

Test support during the trial with representative technical questions. Evaluate response time, accuracy, escalation options, language and time-zone coverage, and whether support is delivered directly or through intermediaries.

Documentation should be current, versioned, detailed, and transparent about prerequisites and limitations.

Customer references are most useful when they resemble the planned deployment. Ask about implementation effort, upgrade experience, unexpected constraints, help desk impact, and post-sale support.

© 2026 Rublon

The partner ecosystem may provide local expertise, but responsibility for defects, compatibility, escalation, and long-term support must remain clear.

Transparent vendors document unsupported scenarios, architectural constraints, and incidents. A meaningful trial should include representative integrations, production-like functionality, and technical assistance.

8. Run a Meaningful Proof of Concept

8.1. Build an MFA Proof-of-Concept Plan

A proof of concept should test the conditions that could change the purchasing decision. It should not be limited to a polished demonstration or the easiest application.

Define the systems, users, identity sources, methods, policies, and failure scenarios in scope. Assign owners from security, infrastructure, IAM, support, and application teams.

8.1.1. Define Success Criteria

Measure whether:

- Required applications are protected through their intended access paths
- Users and groups synchronize correctly
- Required methods work with each integration
- Authentication time is acceptable
- Policies produce the expected result
- Logs provide sufficient context
- Recovery preserves identity assurance
- High availability and failover behave correctly
- Administrative effort is manageable
- User and help desk feedback is acceptable

Record the time required to configure an integration, assign a policy, enroll users, replace an authenticator, diagnose failure, and revoke access.

8.1.2. Use Representative Scenarios

Include:

- An office user
- A remote VPN user

© 2026 Rublon

- An administrator
- A contractor
- A user without a smartphone
- A user with a FIDO2 key or passkey
- A user replacing a phone
- A user with limited connectivity
- An unauthorized user
- Connector or directory failure

Test both successful and failed paths. Reject and expire requests. Disable accounts, remove groups, revoke authenticators, and attempt access again.

Follow the complete recovery journey from reporting a lost device through verification, temporary access, re-enrollment, notification, and revocation.

8.1.3. Collect Evidence

Capture:

- Configuration and prompt screenshots
- Authentication and audit logs
- Deployment time
- Firewall changes
- Local components
- Service accounts and certificates
- External dependencies
- Manual steps and configuration files
- User and administrator feedback
- Support response quality
- Known limitations and workarounds

The final result should show what production deployment would require, including unresolved assumptions and compromises.

8.2. MFA Vendor Evaluation Scorecard

A scorecard converts vendor claims and proof-of-concept results into a comparable decision record. Agree on categories and weights before reviewing final proposals.

© 2026 Rublon

Weights may change by environment. Healthcare may prioritize availability and emergency access. Financial organizations may emphasize phishing resistance and auditability. Legacy-heavy environments may give more weight to integration coverage.

Evaluation Category	Suggested Weight
Security Effectiveness	20%
Integration Coverage	20%
Authentication Methods	10%
Deployment and Operations	15%
User Experience	10%
Administration and Reporting	10%
Reliability and Vendor Assurance	10%
Total Cost of Ownership	5%
Total	**100%**

Use one scoring scale:

Score	Meaning
0	Not supported
1	Major gap or impractical workaround
2	Partially supported with material limitations
3	Meets the requirement
4	Exceeds the requirement
5	Strongly exceeds the requirement with verified evidence

Mandatory requirements must remain separate from the total score. Strength in low-risk categories should not offset the absence of a critical integration, method, or resilience control.

© 2026 Rublon

8.3. Vendor Evaluation Worksheet

Treat vendor responses as claims until supported by documentation, contractual language, or successful testing. Record limitations even when a requirement passes.

Requirement	Priority	Vendor Response	Evidence Provided	Proof-of-Concept Result	Score	Notes	Identified Limitation
	Mandatory / High / Medium / Low		**Documentation / demonstration / contract / test**	**Passed / Partially Passed / Failed / Not Tested**	**0-5**		

8.4. Warning Signs When Evaluating an MFA Vendor

Warning signs should be identified before the platform becomes a critical dependency.

Be cautious when:

- Only cloud applications are demonstrated
- VPNs, Windows, Linux, network devices, or legacy systems are excluded
- Features vary significantly between integrations without clear disclosure
- Important capabilities require unexpected modules, licenses, or services
- Trial functionality differs materially from production licensing
- Outage, failover, and recovery behavior cannot be explained
- No phishing-resistant method is available
- On-premises and legacy support is limited
- Logs cannot distinguish rejection, timeout, policy denial, and technical failure
- Administrative actions and authenticator changes are not audited
- Device replacement lacks reliable identity verification
- Data flows, processing locations, and subprocessors are unclear
- Limitations and version restrictions are undocumented

© 2026 Rublon

- Difficult scenarios are excluded from the proof of concept

A polished browser login does not prove enterprise-wide coverage. A transparent vendor should explain architectural constraints, method differences, dependencies, and recovery behavior before deployment.

9. How Rublon MFA Addresses the Evaluation Criteria

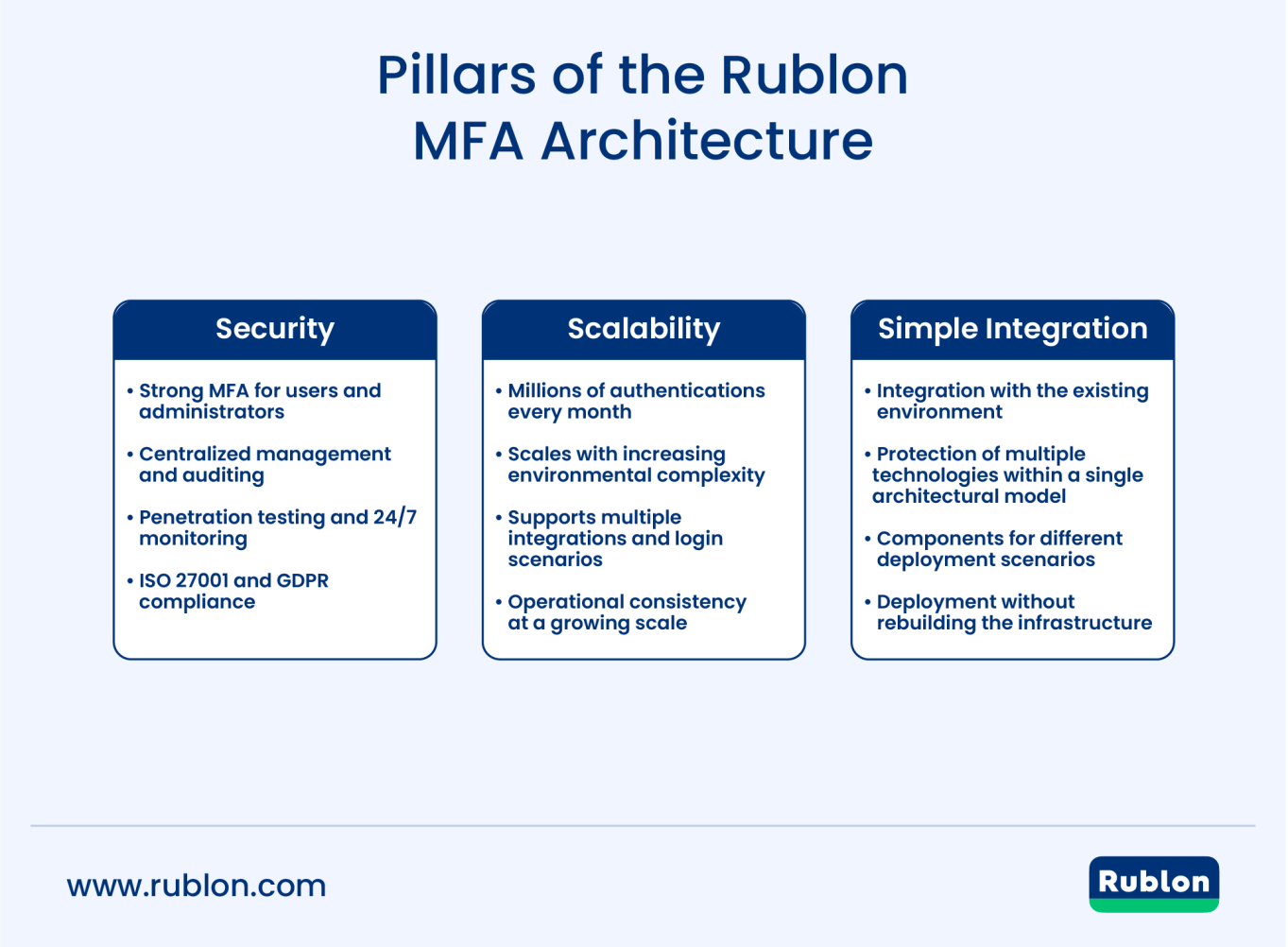
9.1. Rublon MFA at a Glance

Rublon MFA is a centrally managed enterprise MFA platform for cloud, on-premises, and hybrid environments. It adds multi-factor authentication to existing identity and authentication infrastructure without requiring organizations to replace established directories, identity providers, or login processes.

The Rublon Admin Console centralizes users, applications, groups, policies, authenticators, remembered devices, logs, and administrative activity.

Rublon MFA's architecture supports modern applications while retaining systems based on Active Directory, Microsoft Entra ID, LDAP, RADIUS, SAML, and application-specific login flows. Organizations can expand protection gradually according to risk and modernization priorities.

© 2026 Rublon



9.2. Protecting the Organization's Critical Access Paths

Rublon MFA places authentication at the point where users access systems.

- Rublon Authentication Proxy operates as an on-premises RADIUS or LDAP proxy for compatible VPNs, firewalls, network appliances, and access services. It can connect to identity sources including Active Directory, Microsoft Entra ID, OpenLDAP, FreeIPA, FreeRADIUS, and Microsoft NPS. Multiple proxy instances can support high availability.
- Rublon MFA for Windows Logon and RDP protects supported local Windows logons and Remote Desktop Protocol sessions.
- Remote Desktop Services coverage includes Remote Desktop Gateway, Remote Desktop Web Access, RD Web Client, and RD Web Feed. MFA can be applied at the gateway, web access layer, session host, or several points in the access path.
- Rublon MFA for Linux SSH integrates through PAM and protects supported local and remote SSH authentication, including password-based access and configured SSH key scenarios.

- Rublon MFA for macOS integrates with the macOS sign-in mechanism and protects logins by adding a second authentication step after the user's password is verified.
- Rublon Access Gateway provides SAML 2.0 integration for cloud and federated applications.
- Rublon Identity Bridge supports applications based on OAuth 2.0 and OpenID Connect through a centralized authorization layer.
- Rublon App Shield protects self-hosted web applications without changing their source code. It can cover internal portals, administrative panels, legacy web systems, and other browser-based applications.
- Rublon MFA for AD FS adds MFA to Active Directory Federation Services, protecting federated access to applications and services that rely on AD FS.
- Rublon MFA for Outlook Web App (OWA) adds MFA to Outlook Web App and Exchange Control Panel logins on supported Microsoft Exchange deployments.
- Rublon MFA for Roundcube adds MFA to Roundcube webmail logins through a dedicated connector, requiring an additional authentication method after the user enters their credentials.
- The Rublon REST API and SDK libraries add MFA to internally developed applications, including graphical and promptless flows.
- Authentication method availability and user experience vary by integration, so each critical path should be validated through current documentation and proof-of-concept testing.

9.3. Authentication Methods for Different Users and Risk Levels

Rublon MFA supports software and hardware methods for different users, applications, and risk levels. Authentication Methods policies control which methods are available to applications and groups.

The Rublon Authenticator mobile app supports:

- Mobile Push
- TOTP-based Passcode without internet connectivity (RFC 6238)
- QR Code authentication
- Registration on multiple mobile devices

For higher-risk access, Rublon MFA supports FIDO U2F and FIDO2 security keys, as well as FIDO2 passkeys stored in Windows Hello, compatible password managers, mobile devices, or physical keys.

Compatible third-party applications, including Google Authenticator and Microsoft Authenticator, generate TOTP passcodes. The Passcode method also supports SHA-1 OATH

© 2026 Rublon

hardware tokens. YubiKey OTP provides another hardware option, although it is not phishing-resistant like FIDO2.

SMS Passcode and Phone Call can support users who cannot use an authenticator application or reliable internet connection. SMS Link and Email Link provide additional alternatives but require network access to open the verification link. SMS Passcode, SMS Link, and Phone Call consume Phone Credits and are generally better suited to accessibility, transition, or recovery than privileged access.

Supported Windows Logon and RDP deployments can also use compatible RFID cards, badges, key fobs, and other RFID authenticators.

Recovery may combine backup authenticators with administrator-issued Bypass Codes that have configurable expiration and usage limits.

9.4. Centralized Policies, Administration, and Visibility

The Rublon Admin Console centralizes applications, users, groups, policies, authenticators, remembered devices, directory synchronization, administrative access, and reporting.

Administrator roles allow support staff to perform selected recovery and authenticator tasks without receiving full policy or billing access.

Directory Sync imports users and groups from Microsoft Entra ID. Active Directory and OpenLDAP synchronization is available through Rublon Authentication Proxy.

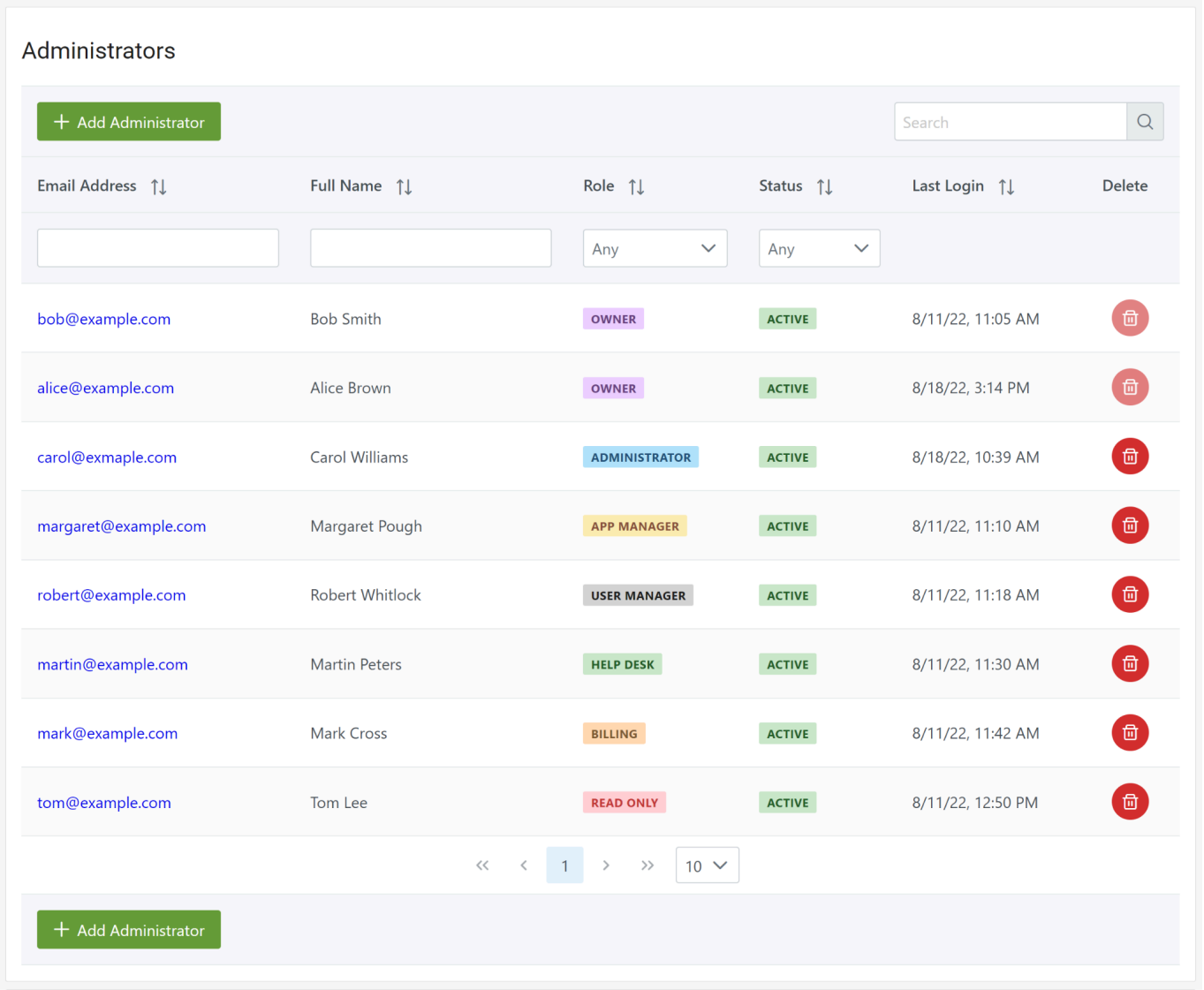
Operational records include:

- **Authentication Logs:** user, application, device, method, result, time, IP address, browser, operating system, and available location data
- **Activity Logs:** authenticator changes made through self-service workflows
- **Audit Logs:** administrative actions
- **Phone Logs:** delivery status and Phone Credit usage

Rublon Log Sync can continuously retrieve these logs from the Rublon Admin API and forward them to a compatible SIEM system for centralized monitoring and analysis.

Remembered devices are centrally visible and governed by policy. They can be restricted or disabled for privileged users and critical applications.

© 2026 Rublon



9.5. Deployment Across Existing Infrastructure

Rublon MFA adds authentication to existing environments without replacing their identity foundation.

It supports identities from Active Directory, Microsoft Entra ID, and compatible LDAP directories. Directory synchronization imports users and groups for application assignment and policy enforcement.

RADIUS and LDAP systems can integrate through Rublon Authentication Proxy. SAML applications can use Rublon Access Gateway. Dedicated connectors cover Windows Logon, Remote Desktop Services, Linux SSH, Microsoft services, and selected cloud applications.

Systems without a native connector may use generic protocols, Rublon App Shield, Rublon Identity Bridge, the Rublon REST API, or SDK libraries.

Application-specific documentation describes prerequisites, configuration, authentication flows, and relevant limitations.

A staged deployment can begin with a staging environment and a small pilot. Teams can validate enrollment, policies, methods, failures, logs, and support procedures before wider enforcement.

Rollout may proceed by group, application, or access path. Administrators and VPN access can be protected first, followed by servers, cloud services, web applications, workforce systems, and more complex legacy applications.

This allows modern and older systems to remain under one MFA platform while modernization progresses at a manageable pace.

Read more: Rublon Guide to Deployment Best Practices.

9.6. Operational Value and Total Cost Considerations

Rublon MFA uses per-user subscriptions. Published pricing includes unlimited application integrations. Adding another supported VPN, server, cloud service, or web application does not create a separate application integration charge.

SMS Passcode, SMS Link, and Phone Call consume separately purchased Phone Credits. Usage should be estimated by user count, frequency, destination country, and recovery needs.

Some integrations require customer-managed hosts, certificates, monitoring, updates, and high-availability arrangements. Internal engineering and help desk effort should also be included in TCO.

Unlimited supported application integrations allow the platform to expand without a new license category for each resource. Consolidating separate MFA tools may also reduce cost, depending on contracts, migration effort, and technical requirements.

9.7. Support, Documentation, and Security Assurance

Rublon provides technical documentation, integration guides, downloads, architecture materials, release notes, deployment guidance, and user communication resources.

Rublon Support assists customers in English, German, and Polish. Current service information is available through the Rublon status page.

A 30-day Rublon MFA Trial provides access to platform features without requiring a payment card, allowing organizations to test representative integrations and workflows.

Rublon operates a certified PN-EN ISO/IEC 27001:2023 information security management system. Security and legal documentation covers data protection, privacy, processing terms, and subprocessors. The Rublon Service Level Agreement commits to at least 99.9% monthly availability for covered paid use, subject to its terms and exclusions.

Customer stories and independent reviews provide further evidence about deployment, usability, and support in different industries.

10. Conclusion and Buyer Resources

10.1. Final MFA Buyer's Checklist

10.1.1. Scope and Coverage

☐ Applications, systems, and access paths are inventoried.

☐ Workforce, administrative, contractor, partner, and external identities are identified.

☐ Privileged and emergency accounts are classified separately.

☐ Cloud, on-premises, remote, local, and hybrid access is included.

☐ Legacy and custom applications are assessed.

☐ Remaining password-only systems are documented and risk accepted.

10.1.2. Authentication and Policy

☐ Required methods work through the intended integrations.

☐ FIDO2 security keys or passkeys are tested for high-risk access.

☐ Policies for users, administrators, contractors, and emergency accounts are validated.

☐ Weaker fallback methods cannot bypass stronger requirements without authorization.

© 2026 Rublon

☐ Remembered-device, network, location, and recovery policies behave as expected.

☐ Enrollment, replacement, and re-enrollment controls are tested.

10.1.3. Operations and Resilience

☐ Authentication and audit logs provide sufficient context.

☐ Retention, export, and SIEM requirements are confirmed.

☐ Help desk permissions follow least privilege.

☐ Offline and limited-connectivity scenarios are tested where required.

☐ Connector, directory, network, and cloud failure behavior is documented.

☐ Fail-open and fail-closed decisions are approved.

☐ High availability and configuration recovery are tested.

☐ Monitoring, maintenance, and upgrade ownership are assigned.

10.1.4. Commercial and Vendor Review

☐ Three-year TCO has been calculated consistently.

☐ Subscription, usage, hardware, infrastructure, support, and labor are included.

☐ Growth in users and applications is modeled.

☐ Certifications, privacy documents, subprocessors, and service commitments are reviewed.

☐ Support has been tested with representative questions.

☐ Customer references match the intended deployment.

☐ Limitations, workarounds, and unsupported scenarios are recorded.

☐ Trial functionality is compared with production licensing.

10.1.5. Deployment Readiness

☐ Mandatory proof-of-concept requirements are satisfied.

© 2026 Rublon

☐ User and administrator feedback is reviewed.

☐ Expected help desk impact is understood.

☐ Pilot users and applications are selected.

☐ Communication, training, escalation, and rollback plans are ready.

☐ Success measures and adoption reporting are defined.

☐ The phased rollout is approved.

10.2. Next Steps

Validate Rublon MFA against the identities, applications, and access paths that matter to your organization.

Use the 30-day trial to connect representative systems, test required methods and policies, inspect logs, and complete recovery and outage scenarios. Review the integration documentation for VPNs, Windows and Linux systems, Remote Desktop Services, cloud services, web applications, and other infrastructure.

For unsupported or custom applications, contact Rublon to discuss the authentication architecture and available integration options. An architecture consultation can also help define local components, directory connections, resilience requirements, proof-of-concept scope, and rollout sequence.

Start evaluating Rublon MFA in your real access environment:

- Start Your Free Trial
- Review Supported Integrations
- Contact Rublon

Rublon sp. z o.o.
ul. Stanislawa Wyspianskiego 11
65-036 Zielona Gora
Poland

www.rublon.com

© 2026 Rublon