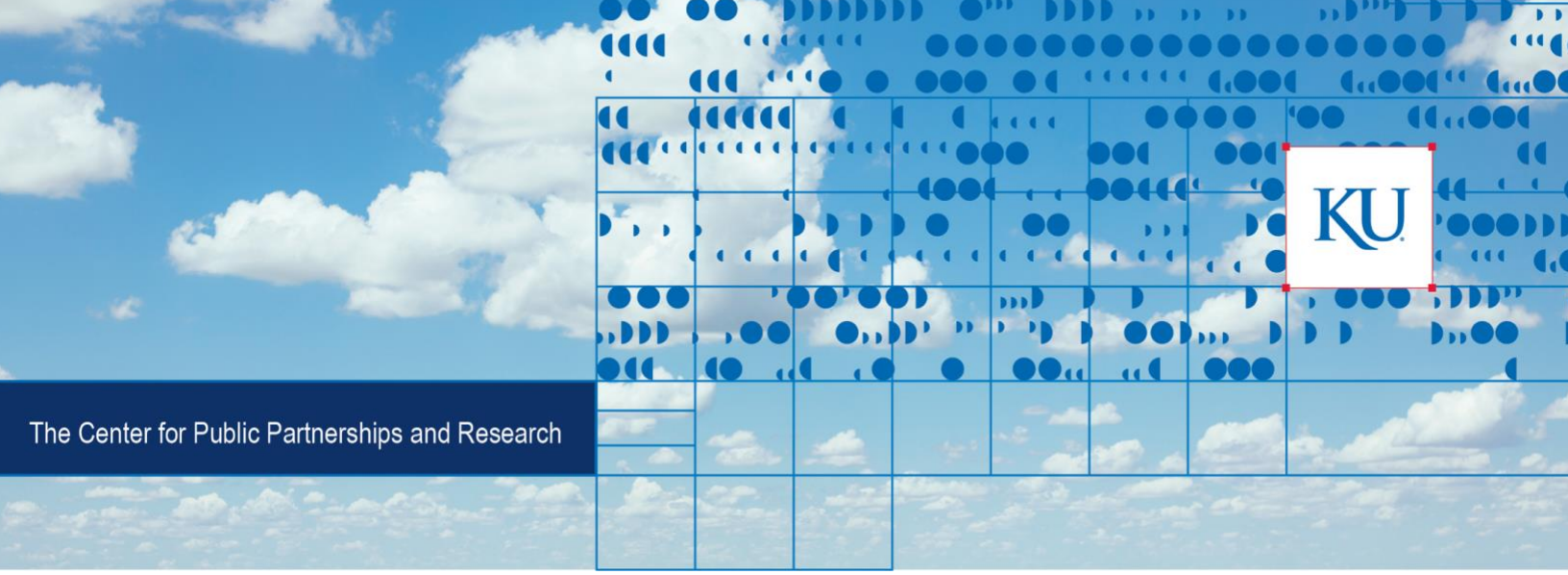

CPPR Multifactor Authentication (MFA) Guidelines For CPPR-Developed Applications

October 2026

Table of Contents

Purpose 1

Guiding Principle 1

Guidelines 2

Risk-Based Expectations 9

Review 9

Purpose

The Center for Public Partnerships and Research (CPPR) develops and maintains technology solutions used by CPPR staff, state and community partners, and other authorized users.

Multifactor authentication (MFA) is an important safeguard against unauthorized access resulting from compromised passwords or credentials. These guidelines establish CPPR's expectations and recommended practices for incorporating MFA into CPPR-developed applications and platforms.

Because CPPR-developed systems vary in purpose, users, technical architecture, contractual requirements, and data sensitivity, these guidelines are intended to establish security guardrails and best practices.

Guiding Principle

CPPR-developed applications should support and use multifactor authentication whenever reasonably possible within customer, technical, operational, contractual, and other legitimate business constraints.

The need for MFA increases with the sensitivity of the information contained within a system and the level of access an account provides.

MFA should be considered a standard security control rather than an optional feature when an application:

- stores, processes, transmits, or provides access to confidential, sensitive, regulated, or personally identifiable information;
- contains individual-level program, participant, client, patient, family, or other protected information;
- allows users to download or export sensitive information;
- provides administrative, configuration, security, reporting, data-management, or user-management privileges;
- integrates with other systems containing sensitive information.

The absence of MFA should receive greater scrutiny as the potential impact of unauthorized access increases.

Guidelines

1. MFA should be considered during application design and development.

MFA should be considered early in the design, architecture, and development of new CPPR applications rather than addressed only after implementation.

For new applications involving sensitive information, MFA should generally be included in the application's security requirements unless there is a documented business, technical, or customer reason not to do so.

2. Authentication decisions should reflect data sensitivity and risk.

The appropriate authentication approach should be based in part on the nature of the information and functionality available through the application.

Factors to consider include:

- the classification and sensitivity of information contained within the application;
- the potential consequences of unauthorized access;
- contractual, legal, regulatory, sponsor, or customer requirements;
- whether users can view, modify, download, transmit, or delete information;
- the number and type of users accessing the system;
- whether users are internal CPPR/KU users or external users;
- integrations with other systems or data sources; and
- the privileges associated with individual accounts.

Applications containing confidential or sensitive information should generally have stronger authentication protections than applications containing only public or low-risk information.

3. MFA should be enabled whenever feasible for sensitive systems.

For applications that contain confidential, sensitive, regulated, or otherwise protected information, MFA should be implemented whenever technically and operationally feasible.

When MFA is not implemented for such a system, the project or product team should document:

- the reason MFA is not feasible;
- the nature and sensitivity of the information involved;
- the risks associated with unauthorized account access;
- alternative or compensating security controls;
- customer expectations and contractual requirements; and
- whether MFA could reasonably be added in a future development cycle.

The intent is not to prohibit legitimate applications that cannot currently support MFA, but to ensure that the decision is intentional and proportionate to the risk.

4. Administrative and privileged accounts should receive heightened protection.

MFA should be strongly prioritized for accounts with administrative or elevated privileges.

This includes accounts that can:

- create, modify, disable, or delete user accounts;
- assign permissions or security roles;
- access data across organizations, customers, programs, or user groups;
- modify application configuration;
- administer integrations or APIs;
- export large amounts of data;
- modify authentication or security settings; or
- otherwise exercise elevated control over the application.

Where technically feasible, privileged users should authenticate through individual accounts rather than shared administrative credentials.

5. Authentication methods should follow current security practices.

When CPPR controls the authentication design, teams should favor established and well-supported authentication methods rather than developing custom MFA mechanisms.

Depending on the application and user population, appropriate approaches may include:

- authentication through an approved identity provider or single sign-on environment;
- passkeys or security keys;
- authenticator applications;
- time-based one-time passwords;
- secure push-based authentication; or
- other widely accepted authentication technologies.

SMS or email verification may be appropriate when stronger methods are not practical for the intended user population, but teams should consider the relative security limitations of different authentication methods when selecting an approach.

Security questions or knowledge-based questions should not generally be treated as a strong second authentication factor.

6. User needs and context should be considered.

CPPR applications frequently operate within state and community-based environments, and serve diverse populations. MFA implementation should therefore consider feasibility for users and user needs in addition to technical security.

Relevant considerations may include:

- user identity-management capabilities;
- contractual or procurement requirements;

- user access to smartphones or other authentication devices;
- accessibility;
- digital literacy;
- availability of organizational email accounts;
- shared-device environments;
- rural or low-connectivity environments;
- implementation and support burden; and
- interoperability with customer systems.

Where these considerations make a particular MFA method impractical, teams should evaluate alternative authentication methods or compensating controls rather than automatically eliminating MFA.

7. Account recovery should be designed as part of authentication security.

Account recovery mechanisms should not undermine MFA protections.

Application teams should consider how users will regain access when they:

- lose or replace an authentication device;
- change phone numbers or email addresses;
- lose access to an authenticator application;
- leave an organization;
- change roles; or
- otherwise lose access to a registered authentication method.

Recovery processes should include appropriate identity verification and should avoid relying on easily obtainable information.

8. Authentication requirements should be documented.

For applications involving sensitive information or significant organizational risk, the project or product team should document the authentication approach as part of the application's security or technical documentation.

Documentation should identify, as appropriate:

- whether MFA is available;
- which user roles require or may use MFA;
- the MFA methods supported;
- significant customer or technical constraints;
- any decision not to implement MFA where it would otherwise be expected; and
- compensating controls used to reduce authentication risk.

Documentation need not be extensive; the goal is to ensure that important security decisions are intentional and understandable.

9. MFA should be revisited as applications evolve.

Teams should reconsider MFA when:

- an application begins collecting more sensitive information;
- new functionality significantly increases user privileges;
- the application expands to new customers or populations;
- authentication technology within the application changes;
- customer or contractual requirements change;
- significant application modernization occurs; or
- security standards and accepted practices evolve.

Applications that cannot reasonably support MFA today should, where appropriate, avoid architectural decisions that unnecessarily prevent MFA from being added later.

Risk-Based Expectations

As a general guide:

Higher expectation for MFA

Applications containing sensitive, confidential, regulated, or individually identifiable information; administrative interfaces; systems allowing significant data export or modification; and accounts with elevated privileges.

Moderate expectation for MFA

Applications containing internal or moderately sensitive information where unauthorized access could create meaningful operational, financial, reputational, or customer impact.

Lower expectation for MFA

Public-facing applications, public information resources, or systems where authenticated users have access only to low-risk information and functionality.

These categories are intended as decision-making guidance rather than rigid classifications.

Review

CPPR should periodically review these guidelines as authentication technologies, University requirements, security practices, customer expectations, and CPPR's application portfolio evolve.