Autenticación de Aplicaciones

1.1 La autenticación como control organizacional

La **autenticación** (*Authentication*) es el proceso mediante el cual una aplicación verifica que una persona, sistema o servicio es quien afirma ser. En organizaciones, ingresar a una aplicación no es un acto menor: cada acceso puede permitir consultar datos, aprobar pagos, modificar proveedores, emitir comprobantes, descargar reportes, crear usuarios o cambiar parámetros críticos.

Por eso, la autenticación debe analizarse como parte del **control interno**, la **seguridad de la información** y la **gestión de riesgos**, no como una pantalla técnica de ingreso.

1.2 Identificación, autenticación y autorización — Tres conceptos distintos

Un error frecuente en las organizaciones es tratar estos tres pasos como uno solo. Son funciones distintas con controles distintos:

Concepto	Pregunta que responde	Ejemplo
Identificación (*Identification*)	¿Quién dice ser?	El usuario escribe su nombre de usuario o correo corporativo
Autenticación (*Authentication*)	¿Puede probarlo?	El usuario ingresa su contraseña y confirma un código temporal
Autorización (*Authorization*)	¿Qué puede hacer?	El sistema le permite cargar solicitudes, pero no aprobar órdenes superiores a $500.000

Un usuario puede autenticarse correctamente y, aun así, no tener permiso para aprobar pagos, modificar precios o consultar datos confidenciales. La autenticación permite ingresar; no define por sí sola el alcance del permiso. Por eso debe combinarse con **perfiles**, **roles**, **segregación de funciones**, **límites de aprobación** y **registros de auditoría**.

1.3 Factores de autenticación

Los métodos de autenticación se organizan en **factores**: tipos de prueba usados para validar identidad.

Factor	Nombre en inglés	Descripción	Ejemplo
Algo que se sabe	*Something you know*	Información conocida por el usuario	Contraseña o PIN
Algo que se tiene	*Something you have*	Elemento físico o digital bajo control del usuario	Código temporal, token o certificado
Algo que se es	*Something you are*	Rasgo biométrico	Huella digital o reconocimiento facial
Algo que se hace	*Something you do*	Patrón de conducta	Forma de escribir o firmar digitalmente
Lugar o contexto	*Somewhere you are*	Condición de ubicación o entorno	Acceso permitido solo desde red corporativa

La **autenticación multifactor (MFA)** utiliza dos o más factores de distintas categorías. Si un atacante obtiene la contraseña, todavía necesita otro elemento para ingresar.

1.4 Métodos de autenticación

1.4.1 Contraseñas — El método clásico

La autenticación por **contraseña** (*Password Authentication*) es el método más tradicional. Combina una identidad declarada con una clave secreta. Su ventaja es la simplicidad. Su debilidad principal es ampliamente conocida: las contraseñas pueden ser débiles, repetidas, robadas, compartidas, anotadas o capturadas mediante engaños.

Políticas recomendadas en organizaciones:

Política	Justificación
Longitud mínima de 12 caracteres	Las claves cortas son más fáciles de descifrar por fuerza bruta
Bloqueo ante intentos fallidos	Dificulta ataques automatizados
Prohibición de claves comunes o filtradas	Evita contraseñas que ya están en bases de datos de atacantes
MFA para sistemas críticos	Reduce el daño si la contraseña es comprometida
Monitoreo de accesos anómalos	Permite detectar usos indebidos aunque la autenticación sea correcta

Obligar a cambiar la contraseña cada 30 días puede llevar a patrones predecibles. En muchos casos, resulta más efectivo exigir contraseñas robustas, impedir claves filtradas, activar MFA y monitorear accesos anómalos.

Caso ilustrativo: si 300 empleados usan una aplicación interna y 45 repiten la misma contraseña en servicios externos, una filtración ajena a la organización puede derivar en accesos indebidos internos. Las contraseñas deben tratarse como un riesgo operativo, no solo como una decisión técnica.

1.4.2 MFA — Autenticación multifactor

La **autenticación multifactor (MFA)** agrega capas de verificación. Puede combinar contraseña con:

- Código temporal generado por una aplicación autenticadora.
- Notificación push que el usuario aprueba en su dispositivo.
- Llave física (*hardware token*).
- Certificado digital.
- Factor biométrico.

MFA reduce significativamente el riesgo de robo de credenciales, phishing y accesos indebidos. No lo elimina: existen ataques que intentan engañar al usuario para que apruebe una solicitud no iniciada por él (*MFA fatigue*). La capacitación es parte indispensable del control.

Autenticación adaptativa: la organización puede aplicar MFA con distinto nivel de exigencia según el riesgo de la situación.

Situación	Nivel de control sugerido
Cuenta administrativa, desde cualquier ubicación	MFA obligatoria en todo acceso
Cuenta de consulta, desde red corporativa	Sin MFA adicional
Cuenta de consulta, desde red externa	MFA requerida
Operación de alto impacto (transferencia, aprobación)	MFA más segregación de funciones

1.4.3 OTP — Contraseña de un solo uso

OTP (*One-Time Password*) es un código válido por una sola vez o por un período breve. Puede enviarse por mensaje, correo o generarse en una aplicación autenticadora.

TOTP (*Time-Based One-Time Password*) es una variante que cambia cada 30 o 60 segundos y se genera localmente en el dispositivo del usuario.

Variante	Canal	Riesgo principal
OTP por SMS	Red telefónica	Puede ser interceptado o redirigido (*SIM swapping*)
OTP por correo	Email del usuario	Depende de la seguridad del correo
TOTP por aplicación	Aplicación autenticadora	Más resistente al phishing; no depende de red
OTP por llave física (*hardware token*)	Dispositivo físico	Muy robusto; requiere gestión del inventario

En sistemas sensibles conviene preferir aplicaciones de autenticación o llaves físicas en lugar de OTP por SMS, que puede ser capturado por engaños o interceptado.

1.4.4 Tokens, JWT y OAuth

Un **token** es una credencial digital que representa una autorización o una sesión. En aplicaciones modernas, los tokens permiten que un sistema reconozca a un usuario o servicio sin pedir la contraseña en cada operación.

Concepto	Descripción	Uso típico
Token	Credencial digital de acceso temporal	Mantener sesión autenticada en aplicaciones web o móviles

Concepto	Descripción	Uso típico
JWT (*JSON Web Token*)	Formato estándar para transportar identidad, permisos y vencimiento	Autenticación entre servicios y aplicaciones web modernas
OAuth (*Open Authorization*)	Protocolo que permite conceder acceso limitado sin compartir la contraseña	Aplicaciones que necesitan operar con recursos de otra plataforma en nombre del usuario

Desde la administración, los tokens deben:

Requisito	Por qué es importante
Tener alcance limitado (*scope*)	Un token de reportes no debería poder crear usuarios ni modificar precios
Tener tiempo de vida razonable	Los tokens de larga duración aumentan el riesgo si son robados
Poder revocarse	Ante un incidente, debe poder cancelarse sin esperar su vencimiento

Si una aplicación de reportes recibe un token para leer datos de ventas, no debería poder modificar precios ni crear usuarios. El alcance debe coincidir con la función necesaria.

1.4.5 Biometría

La **autenticación biométrica** (*Biometric Authentication*) verifica rasgos físicos o conductuales: huella digital, rostro, iris, voz o patrones de escritura. Su ventaja es la comodidad: el usuario no necesita recordar una contraseña. Su riesgo principal es la sensibilidad del dato.

Diferencia crítica: una contraseña comprometida puede cambiarse. Una huella digital no.

Aspecto	Consideración para la organización
Finalidad	Debe estar claramente definida y limitada al uso previsto
Resguardo del dato	El dato biométrico debe almacenarse con controles de seguridad estrictos
Consentimiento	Debe evaluarse según el marco normativo aplicable
Alternativas	Debe existir un método alternativo para quienes no puedan usar biometría
Tiempo de conservación	No debe conservarse más allá de lo necesario

En aplicaciones empresariales, la biometría suele funcionar como factor adicional en dispositivos, no como método único de acceso.

1.4.6 SSO — Inicio de sesión único

El **inicio de sesión único** (*Single Sign-On*, SSO) permite acceder a varias aplicaciones usando una sola autenticación centralizada.

Ventaja	Riesgo
Reduce contraseñas repetidas y fragmentadas	Si la cuenta central es comprometida, varias aplicaciones quedan expuestas
Facilita la baja centralizada al desvincularse un empleado	Concentra el riesgo en un único punto de falla
Mejora la experiencia del usuario y reduce tickets de soporte	Requiere protección especialmente robusta de la identidad central

Ventaja	Riesgo
Facilita la administración de permisos	Exige monitoreo activo de accesos y sesiones

Conclusión sobre SSO: debe combinarse obligatoriamente con MFA, monitoreo, políticas de sesión, revisión de permisos y alertas por accesos inusuales. La identidad central se convierte en un activo crítico.

1.4.7 Certificados digitales

La **autenticación por certificados** (*Certificate-Based Authentication*) utiliza certificados digitales para verificar identidad de usuarios, dispositivos, servidores o aplicaciones. Se basa en criptografía de clave pública (*Public Key Cryptography*): existe una clave pública y una clave privada.

Los certificados son especialmente útiles para: - Autenticar comunicaciones entre sistemas sin usar contraseñas compartidas. - Validar la identidad de servidores. - Firmar digitalmente documentos o transacciones.

Gestión del ciclo de vida — puntos críticos:

Riesgo	Consecuencia	Control
Certificado vencido	Interrupción del servicio o comunicación	Monitorear con al menos 30 días de anticipación
Clave privada filtrada	Posibilidad de suplantación	Resguardar en bóveda de claves; revocar inmediatamente
Certificado no revocado	Acceso indebido tras la desvinculación	Incorporar al proceso de baja de usuarios

1.4.8 Claves de API

Una **clave de API** (*API Key*) es una credencial que permite que una aplicación se autentique frente a otra. Las APIs (*Application Programming Interfaces*) son los mecanismos por los cuales sistemas distintos intercambian información o servicios.

Riesgo: una clave de API expuesta puede permitir consumo indebido de servicios, acceso a datos o generación de costos no previstos.

Formas de exposición frecuentes: - Incluida en código fuente que se publica en repositorios. - Compartida por correo o mensajería informal. - Guardada en planillas o documentos de acceso abierto. - Presente en equipos personales sin control.

Buenas prácticas para la gestión de API Keys:

Práctica	Finalidad
Limitar permisos al mínimo necesario	Una clave de lectura no debe poder escribir ni eliminar
Restringir por dirección IP cuando sea posible	Solo los sistemas autorizados pueden usarla
Rotar claves periódicamente	Reduce el impacto de una filtración no detectada
Guardar en bóvedas de secretos	No deben estar en código, planillas ni correos
Registrar uso	Detectar consumos anómalos o no autorizados
Establecer cuotas de uso	Limitar el daño ante una clave comprometida
Revocar claves no utilizadas	Reducir superficie de ataque

Las claves de API deben protegerse con el mismo criterio que cualquier credencial crítica. Nunca deben compartirse por canales informales.

1.4.9 Cookies y sesiones

Una **cookie** es un pequeño dato guardado por el navegador. Una **sesión** (*Session*) representa el estado de autenticación de un usuario durante un período. Después de ingresar correctamente, la aplicación mantiene la sesión activa para no pedir credenciales en cada pantalla.

Este mecanismo es necesario para la operación, pero genera riesgos:

Riesgo	Descripción
Sesión abierta en equipo compartido	Otra persona puede operar como si fuera el usuario autenticado
Cookie de sesión robada	Puede usarse para suplantar al usuario durante el tiempo de validez
Sesión sin vencimiento	Una sesión que nunca cierra equivale a un acceso permanente

Controles recomendados:

Control	Aplicación
Cierre automático por inactividad	15 minutos en sistemas financieros y administrativos
Vencimiento razonable de sesión	Proporcional a la sensibilidad del sistema
Cierre de sesión manual visible	El usuario debe poder cerrarla activamente
Protección de cookies	Configuración que impide lectura por scripts o terceros

Control	Aplicación
Cierre total al cambiar contraseña	Invalida todas las sesiones activas
Bloqueo ante cambio de dispositivo o ubicación	Detecta posibles robos de sesión

1.5 Criterios para elegir el método de autenticación

La elección del método debe basarse en el riesgo del sistema, no en la preferencia tecnológica.

Tipo de aplicación	Riesgo principal	Método recomendado
Correo corporativo	Robo de información y suplantación	Contraseña robusta, MFA y alertas de acceso
Sistema contable	Modificación de registros y pagos indebidos	MFA, roles, auditoría y sesiones cortas
Portal de empleados	Datos personales y recibos	Contraseña robusta, MFA según sensibilidad, registro de accesos
Aplicación de clientes	Acceso a cuentas y datos	MFA opcional u obligatoria según tipo de operación
Integración entre sistemas	Uso indebido de servicios	API Keys con permisos limitados, certificados o tokens de corto plazo
Administración de usuarios	Creación o baja de accesos críticos	MFA obligatoria, doble control y registro de cambios

La administración debe participar en esta decisión porque el nivel de autenticación afecta costos, tiempos, experiencia del usuario y riesgo. Un exceso de controles puede generar atajos inseguros. Un defecto puede generar incidentes graves.

1.6 Controles administrativos complementarios

La autenticación técnica debe acompañarse con controles de gestión que aseguren su efectividad en el tiempo:

Control	Descripción
Política de accesos documentada	Define quién solicita un usuario, quién lo aprueba, qué datos se registran, cómo se otorgan permisos y cuándo se revisan
Revisión periódica de accesos	Mensual en áreas críticas, trimestral en áreas de riesgo medio; identifica cuentas sin uso, permisos acumulados y accesos incompatibles
Baja inmediata	Al desvincularse o cambiar de función, los accesos deben ajustarse sin demora; 5 días de retraso pueden generar exposición real
Registro de cambios sensibles	Alta, cambio de rol, modificación de permisos, restablecimiento de contraseña y desactivación de MFA deben quedar documentados con usuario, fecha y motivo
Capacitación de usuarios	La autenticación falla si las personas comparten códigos, aprueban notificaciones no iniciadas, guardan

Control	Descripción
	contraseñas en lugares visibles o responden solicitudes engañosas

1.7 Indicadores para la gestión de la autenticación

Los indicadores permiten medir el estado real de la autenticación y detectar brechas antes de que deriven en incidentes.

Indicador	Qué mide	Criterio de seguimiento
Porcentaje de cuentas con MFA	Nivel de protección adicional	Objetivo cercano al 100% en sistemas críticos
Cuentas inactivas	Usuarios sin uso reciente	Revisar y desactivar según política; riesgo de acceso indebido o de exempleados
Intentos fallidos de acceso	Posibles ataques o errores	Analizar aumentos por usuario, horario o ubicación
Restablecimientos de contraseña	Frecuencia de soporte y riesgo	Identificar áreas con alta repetición; puede indicar phishing o hábitos inseguros
Cuentas compartidas	Falta de trazabilidad	Deben eliminarse o justificarse formalmente con responsable asignado

Indicador	Qué mide	Criterio de seguimiento
Accesos fuera de horario	Operaciones anómalas	Revisar especialmente en sistemas financieros y administrativos
Tokens o API Keys sin vencimiento	Exposición técnica acumulada	Definir expiración y rotación periódica
Certificados próximos a vencer	Riesgo de interrupción de servicio	Controlar con al menos 30 días de anticipación

Un tablero de autenticación puede presentarse trimestralmente a la dirección. En áreas críticas, la revisión mensual es más adecuada.

1.8 Ideas clave

- La autenticación no es una pantalla de ingreso: es un **control organizacional** que determina quién puede operar sobre datos, pagos, proveedores y procesos críticos. Su diseño es una decisión de gestión, no solo técnica.

- **Identificación, autenticación y autorización son funciones distintas**. Confundirlas lleva a errores de diseño: un usuario autenticado no necesariamente tiene permiso para todo, y un sistema que no distingue estos tres pasos carece de control granular sobre sus accesos.

- La **autenticación multifactor (MFA)** es el control individual más efectivo para reducir el riesgo de robo de credenciales. Debe ser obligatoria en sistemas críticos y combinarse con capacitación para evitar ataques de *MFA fatigue*.

- Los **tokens y las API Keys** son credenciales que deben gestionarse con el mismo rigor que las contraseñas. Un token sin vencimiento o una clave de API sin restricciones equivalen a un acceso abierto e incontrolado.

- El **SSO** mejora la experiencia del usuario y facilita la administración, pero concentra el riesgo en un único punto. Requiere protección especialmente robusta de la identidad central.
- Los **datos biométricos** no pueden cambiarse como una contraseña. Su uso requiere evaluación de finalidad, resguardo, alternativas y cumplimiento normativo.
- La **baja oportuna de usuarios** es tan crítica como el alta correcta. Una cuenta activa de un exempleado o de un tercero sin vigencia representa un riesgo real aunque la persona ya no tenga relación con la organización.
- Los **indicadores de autenticación** convierten una función técnica en un proceso gestionable y auditable. Sin medición, no es posible saber si los controles funcionan ni justificar las inversiones ante la dirección.

1.9 Preguntas de evaluación

1. ¿Qué diferencia existe entre identificación, autenticación y autorización? Proporcione un ejemplo concreto de los tres pasos en un sistema de pagos.
2. ¿Por qué la autenticación es relevante para la administración y no solo para el área de TI? Identifique al menos tres procesos organizacionales donde una falla de autenticación puede generar consecuencias operativas.
3. ¿Qué ventajas y riesgos tiene la autenticación por contraseña como método único? ¿Qué políticas permiten reducir su exposición?
4. ¿Por qué la autenticación multifactor reduce el riesgo de robo de credenciales? ¿Qué riesgo específico permanece incluso con MFA activa?
5. ¿Qué es un token y por qué debe tener alcance y vencimiento definidos? Proporcione un ejemplo de daño potencial si un token no tiene restricciones.
6. ¿Qué riesgos administrativos aparecen cuando una organización implementa inicio de sesión único (SSO) sin controles complementarios?

7. ¿Por qué las claves de API deben protegerse como credenciales críticas? ¿Cuáles son las formas más frecuentes de exposición inadvertida?
8. ¿Qué controles deberían aplicarse sobre cookies y sesiones en aplicaciones administrativas? ¿Por qué el tiempo de inactividad es relevante?
9. ¿Qué indicadores pueden usarse para medir la calidad de la gestión de autenticación y cómo se presentarían ante la dirección?
10. ¿Por qué la baja oportuna de usuarios es tan importante como el alta correcta? Describa el riesgo concreto que genera una cuenta activa de un exempleado.

1.10 Glosario

Término	Traducción / Explicación
API	*Application Programming Interface*. Interfaz de Programación de Aplicaciones. Mecanismo que permite que sistemas distintos intercambien información o servicios de forma estandarizada.
API Key	Clave de API. Credencial que permite que una aplicación se autentique frente a otra. Debe gestionarse con el mismo rigor que cualquier contraseña crítica.
Authentication	Autenticación. Proceso mediante el cual una aplicación verifica que una persona, sistema o servicio es quien afirma ser, mediante una prueba.
Authorization	Autorización. Función que determina qué acciones puede realizar una identidad ya autenticada dentro de un sistema. Distinta de la autenticación.

Término	Traducción / Explicación
Biometric Authentication	Autenticación biométrica. Verificación de identidad mediante rasgos físicos o conductuales: huella digital, rostro, iris, voz o patrones de escritura.
Certificate-Based Authentication	Autenticación por certificados. Método que utiliza certificados digitales para verificar identidad de usuarios, dispositivos, servidores o aplicaciones.
Cookie	Pequeño dato guardado por el navegador del usuario que permite a la aplicación mantener el estado de autenticación sin pedir credenciales en cada pantalla.
Hardware Token	Llave de seguridad física. Dispositivo físico que genera o almacena códigos de autenticación. Muy resistente al phishing.
Identification	Identificación. Primer paso del proceso de acceso: el usuario declara su identidad (nombre de usuario, correo). No implica verificación.
JSON	*JavaScript Object Notation*. Formato de datos estructurados usado para intercambiar información entre sistemas de forma legible y compacta.
JWT	*JSON Web Token*. Token Web en formato JSON. Formato estándar que transporta afirmaciones sobre una identidad: usuario, vencimiento y permisos.
MFA	*Multi-Factor Authentication*. Autenticación multifactor. Mecanismo que utiliza dos o más factores de autenticación de distintas categorías para verificar identidad.

Término	Traducción / Explicación
MFA Fatigue	Fatiga de MFA. Ataque en el que el atacante envía múltiples solicitudes de aprobación push hasta que el usuario, por cansancio o distracción, aprueba una de forma involuntaria.
OAuth	*Open Authorization*. Protocolo que permite conceder acceso limitado a recursos sin compartir la contraseña principal. Usado para que aplicaciones operen con recursos de otras plataformas en nombre del usuario.
OTP	*One-Time Password*. Contraseña de un solo uso. Código válido por una sola vez o por un período breve, generado por SMS, correo o aplicación autenticadora.
PIN	*Personal Identification Number*. Número de identificación personal. Código numérico usado como factor de autenticación.
Public Key Cryptography	Criptografía de clave pública. Sistema criptográfico que utiliza dos claves matemáticamente relacionadas: una pública (para verificar) y una privada (para firmar o descifrar).
Session	Sesión. Estado de autenticación de un usuario durante un período de tiempo dentro de una aplicación, mantenido sin necesidad de pedir credenciales en cada interacción.
SIM Swapping	Duplicación de SIM. Ataque en que el atacante transfiere el número telefónico de la víctima a una SIM bajo su control, interceptando los OTP enviados por SMS.
Scope	Alcance. En tokens y OAuth, define exactamente qué operaciones puede realizar la aplicación autorizada con los recursos concedidos.

Término	Traducción / Explicación
SSO	*Single Sign-On*. Inicio de sesión único. Mecanismo que permite acceder a varias aplicaciones con una sola autenticación centralizada.
TI	Tecnologías de la Información. Conjunto de recursos tecnológicos utilizados para procesar, almacenar, transmitir y proteger información.
TOTP	*Time-Based One-Time Password*. Contraseña de un solo uso basada en tiempo. Variante de OTP que cambia cada 30 o 60 segundos y se genera localmente en el dispositivo del usuario.