K

Kerberos Authentication: Tickets, Security, and Troubleshooting

Kerberos is a ticket-based network authentication protocol. Learn how TGTs and service tickets work, what can go wrong, and how to secure deployments.

PUBLISHER
Gridinsoft Support

UPDATED
Oct 8, 2026

READ TIME
5 min

CONTENTS

In this guide

01 What Kerberos does
02 Kerberos terms
03 How Kerberos authentication works
04 Authentication is not authorization
05 Common security risks
06 Hardening Kerberos
07 Kerberos troubleshooting workflow
08 Frequently asked questions

Quick answer: Kerberos is a network authentication protocol that uses time-limited tickets so a user or service can prove identity without sending a password to every application. A trusted Key Distribution Center (KDC) issues a ticket-granting ticket (TGT)

and service tickets. Kerberos authenticates identities; it does not by itself decide every authorization permission or encrypt all application data.

SAML is another single-sign-on technology, but its identity assertions and browser-based federation flow differ from Kerberos tickets. Identify the protocol in use before applying a troubleshooting or trust-configuration procedure.

01 What Kerberos does

Kerberos provides single sign-on and mutual authentication in many enterprise environments. In an Active Directory domain, the KDC runs as part of the domain controller and uses the directory's account secrets. Microsoft's Kerberos authentication overview explains the Windows implementation and supported features.

The protocol reduces repeated password transmission, but tickets are credentials. Anyone who steals or forges a usable ticket may impersonate its identity until the ticket expires or defensive actions invalidate its usefulness. Protecting endpoints, domain controllers, keys, and privileged sessions remains essential.

02 Kerberos terms

- **Principal:** a user, computer, or service identity known to the Kerberos realm.
- **KDC:** the trusted service that issues tickets. It includes the Authentication Service (AS) and Ticket-Granting Service (TGS).
- **TGT:** a ticket presented to the TGS to request tickets for individual services.
- **Service ticket:** a ticket intended for a specific service, such as a file share or web application.
- **SPN:** a Service Principal Name that maps a service instance to the account under which it runs.
- **Realm:** the administrative authentication boundary; in Active Directory this generally corresponds to a domain.

03 How Kerberos authentication works

1. **Initial authentication:** the client contacts the AS. After validating the account, the KDC returns a TGT protected so only the appropriate parties can use it.
2. **Ticket request:** when the user accesses a service, the client presents the TGT to the TGS and asks for a ticket for that service's SPN.
3. **Service access:** the client presents the service ticket to the target server. The service validates it using its account key.
4. **Mutual authentication:** the service can prove its identity back to the client, helping prevent connection to an impostor.

The user's password is not sent to the application server. Ticket contents and authenticators provide proof, and timestamps help prevent replay. Clocks therefore need reasonable synchronization.

04 Authentication is not authorization

Kerberos answers "which identity presented a valid ticket?" The application, operating system, or directory then decides what that identity may do using group memberships, access-control lists, roles, and policy. A successful ticket does not automatically grant access to every resource. Likewise, a Kerberos error may be caused by the service identity or SPN rather than a user's permission.

05 Common security risks

- **Ticket theft:** malware with sufficient access may extract tickets from a session and reuse them.
- **Kerberoasting:** an authenticated user requests service tickets and attempts offline password guessing against service-account material. Strong managed service-account secrets reduce risk.

- **AS-REP roasting:** accounts configured without preauthentication may expose material for offline guessing.
- **Forged tickets:** compromise of high-value domain keys can enable powerful persistence such as golden or silver ticket techniques.
- **Unsafe delegation:** overly broad delegation can allow a compromised service to impersonate users to other services.
- **Legacy encryption:** older algorithms and weak account passwords increase cracking and compatibility risk.

These are not flaws that make Kerberos inherently unsafe. They show why identity tiering, strong keys, controlled delegation, endpoint security, and domain-controller protection matter.

06 Hardening Kerberos

Use long, automatically managed service-account passwords where possible, remove unnecessary SPNs and inactive accounts, and require preauthentication. Prefer currently supported encryption types and plan migration away from legacy dependencies rather than disabling compatibility blindly. Restrict delegation, separate privileged administration from ordinary workstations, minimize who can log on to domain controllers, and monitor changes to sensitive accounts and policies.

Alert on unusual volumes of service-ticket requests, rare encryption types, suspicious ticket lifetimes, access from unexpected hosts, and changes to SPNs or delegation. Baseline normal applications first because backups, scanners, and service farms can generate high ticket volumes legitimately.

07 Kerberos troubleshooting workflow

1. **Confirm scope:** identify the user, client, service, server, time, and exact error rather than treating all login failures alike.
2. **Check prerequisites:** verify DNS resolution, domain-controller discovery, network reachability, and time synchronization.

3. **Validate the SPN:** ensure the requested SPN exists on the correct account and is not duplicated.
4. **Inspect tickets and logs:** review client, server, and domain-controller events; confirm which ticket was requested and which encryption type was negotiated.
5. **Check identities:** look for locked, disabled, expired, recently changed, or mismatched user, computer, and service-account secrets.
6. **Test a narrow correction:** fix the underlying DNS, time, SPN, account, delegation, or crypto issue, then retest and document it.

Avoid “fixes” that disable Kerberos, turn off preauthentication, enable broad delegation, or restore weak encryption across the domain. Microsoft publishes a detailed Kerberos troubleshooting guide for Windows environments.

08 Frequently asked questions

Does Kerberos send my password to every server?

No. The client uses tickets issued by the KDC rather than sending the password to each application service.

Why does time matter?

Kerberos uses timestamps to limit replay. Excessive clock difference can cause authentication failures.

Does Kerberos encrypt all network traffic?

No. It authenticates and can provide session-key material, but the application protocol and configuration determine protection of the actual data.

NEED MORE HELP?

We are here when the guide is not enough.

Visit the Gridinsoft Support Center for current product guidance or submit a support request with the details of your case.

Open this guide online at support.gridinsoft.com