

Routing

- [Flood Routing – Verteilung über das Mesh](#)
- [Path Hashes – kompakte Repeaterkennungen](#)
- [Regions und Scopes](#)
- [Hops und Grenzen](#)
- [Direct Routing und bekannte Pfade](#)
- [Routing-Grundlagen](#)
- [Routing in der praktischen Netzplanung](#)

Flood Routing - Verteilung über das Mesh

Flood Routing verteilt dafür vorgesehene Pakete über erreichbare Repeater. Es ist ein grundlegender Mechanismus für Situationen, in denen kein einzelner bekannter Zielpfad verwendet wird.

Ablauf

Ein Sender strahlt ein Flood-Paket aus. Geeignete Repeater, die es empfangen und nach ihren Regeln weiterleiten dürfen, können es erneut senden. Weitere Repeater können diese Aussendung wiederum empfangen.

Damit kann ein Paket Bereiche erreichen, zu denen der ursprüngliche Sender keine direkte Funkverbindung besitzt.

Vorteil

Flooding benötigt keine vollständige zentrale Netzkarte. Das ist für ein dezentrales Funknetz attraktiv: Ein Knoten muss nicht vorher jeden möglichen Weg durch die gesamte Region kennen.

Kosten

Jede Weiterleitung ist eine eigene Funkübertragung. Dadurch steigt die gesamte Airtime. In einem kleinen Netz ist das oft unkritisch; in einem dichten oder weit verbundenen Netz kann ungebremstes Flooding jedoch viel unnötigen Verkehr erzeugen.

Nicht jedes Paket wird endlos weitergetragen

MeshCore besitzt Begrenzungs- und Filtermechanismen. Dazu gehören Pfadlängenbegrenzungen und bei aktuellen Versionen Regions/Scopes beziehungsweise Transport Codes.

Zusätzlich entscheidet die jeweilige Firmwarerolle, ob sie ein Paket überhaupt weiterleitet. Ein normaler Companion ist nicht automatisch ein Repeater.

Flood-Adverts

Flood-Adverts sind ein leicht verständliches Beispiel. Ein Companion kann seine Bekanntmachung über Repeater verbreiten. Dadurch werden entfernte Teilnehmer auf ihn aufmerksam.

Würde jeder Teilnehmer in einem großen Netz ständig Flood-Adverts senden, entstünde jedoch erhebliche zusätzliche Funklast. Intervalle und manuelle Nutzung sollten deshalb zur Netzgröße passen.

Gruppenkommunikation

Auch Gruppenkommunikation kann Flood-Mechanismen verwenden. Der Kanal selbst definiert dabei die Teilnehmer über den gemeinsamen Schlüssel; der Scope kann zusätzlich begrenzen, in welchem Regionsbereich die Nachricht transportiert wird.

Gute Praxis

Flooding sollte dort eingesetzt werden, wo seine Robustheit und Reichweite gebraucht werden. Ein bereits bekannter stabiler Direktpfad ist für gezielte Kommunikation häufig effizienter.

Für MeshCore Hessen bedeutet das: regionale Netze sollen nicht einfach möglichst weit miteinander „zusammengeflutet“ werden. Regions und Scopes bieten einen Ansatz, lokalen Verkehr lokal zu halten.

Quellen, Rechte und redaktionelle Prüfung

- [MeshCore – offizielle Dokumentation](#)
- [meshcore-cli – offizielles Projekt](#)
- [MeshCore Wiki DE – ergänzende Communityquelle](#)

Redaktioneller Hinweis: Diese Seite ist eine eigenständige Darstellung für MeshCore Hessen. Sachinformationen wurden anhand der oben genannten Quellen geprüft. Texte, Bilder und die Gliederung des MeshCore Wiki DE wurden nicht übernommen; die Seite wird nur als ergänzende Communityquelle zur Faktenkontrolle genannt. Für technische Details hat die aktuelle offizielle MeshCore-Dokumentation Vorrang. Deutschlandbezogene Funkrechtsangaben werden an den Veröffentlichungen der Bundesnetzagentur geprüft.

Urheberrecht: Technische Begriffe, Protokollnamen, Befehlsnamen, Zahlenwerte und andere notwendige Sachangaben werden unverändert verwendet, soweit dies für fachliche Richtigkeit erforderlich ist. Es werden keine längeren fremden Textpassagen oder Bilder reproduziert.

Prüfstand: 18. August 2026

Path Hashes - kompakte Repeaterkennungen

Ein vollständiger öffentlicher Schlüssel wäre für jeden Repeater in einem Funkpfad zu groß. MeshCore verwendet deshalb kompakte Hash-Werte, um Zwischenstationen innerhalb eines Pfades zu kennzeichnen.

Aktuelles Paketformat

Im aktuellen Packet Format kodiert das Path-Length-Byte zwei Dinge gleichzeitig:

- die Anzahl der Path-Hash-Einträge in den unteren sechs Bits
- die Hash-Größe in den oberen zwei Bits

Dokumentiert sind derzeit 1, 2 und 3 Byte pro Path Hash. Der vierte mögliche Code ist reserviert beziehungsweise ungültig.

Warum mehrere Größen?

Kurze Hashes sparen Funkdaten und damit Airtime. Mit wachsender Knotenzahl steigt jedoch die Wahrscheinlichkeit, dass kurze Kennungen nicht mehr eindeutig genug sind. Größere Hashes reduzieren dieses Kollisionsrisiko, benötigen aber mehr Platz im Paket.

Das ist ein klassischer Zielkonflikt zwischen Effizienz und Eindeutigkeit.

Kompatibilität

Ältere Firmwarestände unterstützten nur das frühere 1-Byte-Verhalten. In einem gemischten Netz ist deshalb nicht automatisch die größtmögliche Hash-Größe die beste Wahl.

Vor Änderungen an produktiver Community-Infrastruktur muss geprüft werden, welche Firmwarestände tatsächlich im Netz eingesetzt werden.

Maximale Pfaddaten

Das aktuelle Format begrenzt den Pfad auf eine maximale Bytezahl. Dadurch hängt die praktisch mögliche Anzahl von Pfadeinträgen auch von der gewählten Hash-Größe ab.

Eine größere Hash-Größe verbessert also die Kennungsqualität, reduziert aber die Zahl der Einträge, die innerhalb derselben maximalen Pfaddatenmenge Platz finden können.

Für Anwender

Normale Nutzer müssen Path Hashes nicht manuell berechnen. Das Thema ist vor allem für Repeaterbetreiber, Netzdiagnose, CLI-Nutzer und Entwickler relevant.

Quellen, Rechte und redaktionelle Prüfung

- [MeshCore – offizielle Dokumentation](#)
- [meshcore-cli – offizielles Projekt](#)
- [MeshCore Wiki DE – ergänzende Communityquelle](#)

Redaktioneller Hinweis: Diese Seite ist eine eigenständige Darstellung für MeshCore Hessen. Sachinformationen wurden anhand der oben genannten Quellen geprüft. Texte, Bilder und die Gliederung des MeshCore Wiki DE wurden nicht übernommen; die Seite wird nur als ergänzende Communityquelle zur Faktenkontrolle genannt. Für technische Details hat die aktuelle offizielle MeshCore-Dokumentation Vorrang. Deutschlandbezogene Funkrechtsangaben werden an den Veröffentlichungen der Bundesnetzagentur geprüft.

Urheberrecht: Technische Begriffe, Protokollnamen, Befehlsnamen, Zahlenwerte und andere notwendige Sachangaben werden unverändert verwendet, soweit dies für fachliche Richtigkeit erforderlich ist. Es werden keine längeren fremden Textpassagen oder Bilder reproduziert.

Prüfstand: 18. August 2026

Regions und Scopes

Regions und Scopes sind Mechanismen zur Begrenzung und Strukturierung von Flood-Verkehr. Sie werden besonders wichtig, wenn aus vielen lokalen MeshCore-Inseln ein größeres zusammenhängendes Netz entsteht.

Grundidee

Eine **Region** wird auf Repeater-Infrastruktur definiert. Ein **Scope** wird beim Senden verwendet, um den gewünschten räumlich beziehungsweise hierarchisch definierten Bereich anzugeben.

Auf Paketebene werden dafür Transport Codes verwendet. Repeater können anhand ihrer Regionskonfiguration entscheiden, ob entsprechender Flood-Verkehr weitergetragen werden darf.

Warum das nötig ist

Ohne Begrenzung könnte eine lokale Nachricht durch eine immer größer werdende Repeaterlandschaft weit über ihren sinnvollen Zielbereich hinaus weitergetragen werden.

Ein Gespräch innerhalb eines Landkreises muss nicht automatisch sämtliche erreichbaren Repeater in anderen Teilen Hessens beschäftigen.

Hierarchie

Regions können hierarchisch organisiert werden. Dadurch lassen sich lokale Bereiche in größere Bereiche einordnen.

Ein mögliches **Planungsbeispiel** für MeshCore Hessen wäre etwa Hessen ? Mittelhessen ? Landkreis Gießen ? lokaler Bereich. Das ist hier ausdrücklich ein Strukturbeispiel und keine bereits verbindlich festgelegte MeshCore-Hessen-Konfiguration.

Scope beim Client

Aktuelle Clients beziehungsweise Werkzeuge können einen Flood Scope setzen. Die CLI dokumentiert beispielsweise einen temporären Scope sowie die Möglichkeit, unscoped zu senden.

Der Scope verändert nicht den Verschlüsselungsstatus eines Kanals. Er steuert die Transportreichweite des dafür vorgesehenen Verkehrs.

Repeaterberechtigungen

Die Repeater-CLI bietet Region-Kommandos zum Anlegen hierarchischer Regionsdefinitionen und zum Erlauben oder Verweigern von Flood-Verkehr für bestimmte Regionen.

Damit kann Infrastruktur bewusst steuern, welche regionalen Bereiche sie transportiert.

Auto-Discovery und Versionsstand

Regions/Scopes werden aktiv weiterentwickelt. Clients können Funktionen zur automatischen Erkennung verfügbarer Regionsinformationen anbieten. Konkrete Mindestversionen und Bedienwege ändern sich schneller als das Grundprinzip und werden deshalb in versionsbezogenen Anleitungen gepflegt.

Planung für Hessen

Bevor eine verbindliche Regionsstruktur für MeshCore Hessen veröffentlicht wird, sollte sie mit realer Infrastruktur und benachbarten Communities abgestimmt werden. Ein einmal weit verbreitetes Namensschema lässt sich später nur mit Aufwand ändern.

Die Hierarchie sollte verständlich, stabil und nicht unnötig tief sein. Politische Verwaltungsgrenzen können hilfreich sein, müssen aber nicht jede reale Funkstruktur perfekt abbilden.

Quellen, Rechte und redaktionelle Prüfung

- [MeshCore – offizielle Dokumentation](#)
- [meshcore-cli – offizielles Projekt](#)
- [MeshCore Wiki DE – ergänzende Communityquelle](#)

Redaktioneller Hinweis: Diese Seite ist eine eigenständige Darstellung für MeshCore Hessen. Sachinformationen wurden anhand der oben genannten Quellen geprüft. Texte, Bilder und die Gliederung des MeshCore Wiki DE wurden nicht übernommen; die Seite wird nur als ergänzende Communityquelle zur Faktenkontrolle genannt. Für technische Details hat die aktuelle offizielle

MeshCore-Dokumentation Vorrang. Deutschlandbezogene Funkrechtsangaben werden an den Veröffentlichungen der Bundesnetzagentur geprüft.

Urheberrecht: Technische Begriffe, Protokollnamen, Befehlsnamen, Zahlenwerte und andere notwendige Sachangaben werden unverändert verwendet, soweit dies für fachliche Richtigkeit erforderlich ist. Es werden keine längeren fremden Textpassagen oder Bilder reproduziert.

Prüfstand: 18. August 2026

Hops und Grenzen

Ein Hop ist eine Funkstrecke innerhalb eines mehrstufigen Übertragungswegs. Geht ein Paket vom Companion zu Repeater A, weiter zu Repeater B und anschließend zum Ziel, besteht der Weg aus mehreren Funkabschnitten.

Kein Meshtastic-Limit übernehmen

MeshCore und Meshtastic sind unterschiedliche Systeme. Die bekannte Meshtastic-Aussage über maximal sieben Hops darf nicht auf MeshCore übertragen werden.

Das aktuelle MeshCore-Paketformat verwendet sechs Bits für die Anzahl der Path-Hash-Einträge und dokumentiert Werte bis 63 in diesem Feld; die Firmwaredokumentation beschreibt darüber hinaus interne Grenzen im Bereich von 64 Hops beziehungsweise Pfadeinträgen. Entscheidend ist: **MeshCore hat kein pauschales Sieben-Hop-Limit.**

Technische Grenze ist keine Planungsempfehlung

Nur weil ein Protokoll eine hohe Zahl darstellen kann, sollte ein reales Netz nicht auf extrem lange Wege ausgelegt werden.

Jeder zusätzliche Hop benötigt Zeit. Bei Flooding entstehen zusätzliche Aussendungen; bei direkten Pfaden wächst der Pfadanteil des Pakets. Außerdem steigt die Wahrscheinlichkeit, dass sich ein Zwischenknoten oder Funkweg verändert.

Latenz

Mehrere Funkstrecken bedeuten zusätzliche Verzögerung. MeshCore ist ohnehin kein Echtzeit-Sprachsystem. Lange Wege können die Verzögerung weiter erhöhen.

Zuverlässigkeit

Ein Pfad ist nur so verfügbar wie seine beteiligten Zwischenstationen. Fällt einer von vielen zwingend benötigten Repeatern aus, kann der gespeicherte Weg unbrauchbar werden.

Netzplanung

Für Hessen sollte das Ziel nicht lauten, einen Rekord für möglichst viele Hops zu erreichen. Sinnvoller ist ein Backbone aus guten Standorten, das Regionen mit möglichst wenigen zuverlässigen Übergängen verbindet.

Regions und Scopes können zusätzlich verhindern, dass lokaler Flood-Verkehr unnötig durch ein sehr großes zusammenhängendes Netz läuft.

Quellen, Rechte und redaktionelle Prüfung

- [MeshCore – offizielle Dokumentation](#)
- [meshcore-cli – offizielles Projekt](#)
- [MeshCore Wiki DE – ergänzende Communityquelle](#)

Redaktioneller Hinweis: Diese Seite ist eine eigenständige Darstellung für MeshCore Hessen. Sachinformationen wurden anhand der oben genannten Quellen geprüft. Texte, Bilder und die Gliederung des MeshCore Wiki DE wurden nicht übernommen; die Seite wird nur als ergänzende Communityquelle zur Faktenkontrolle genannt. Für technische Details hat die aktuelle offizielle MeshCore-Dokumentation Vorrang. Deutschlandbezogene Funkrechtsangaben werden an den Veröffentlichungen der Bundesnetzagentur geprüft.

Urheberrecht: Technische Begriffe, Protokollnamen, Befehlsnamen, Zahlenwerte und andere notwendige Sachangaben werden unverändert verwendet, soweit dies für fachliche Richtigkeit erforderlich ist. Es werden keine längeren fremden Textpassagen oder Bilder reproduziert.

Prüfstand: 18. August 2026

Direct Routing und bekannte Pfade

Direct Routing verwendet einen konkreten Pfad durch das Mesh. Statt ein Paket breit über alle geeigneten Repeater zu verteilen, beschreibt der Pfad die vorgesehenen Zwischenstationen.

Warum direkte Pfade sinnvoll sind

Nach erfolgreicher Pfadfindung ist bekannt, über welche Repeater ein Ziel erreicht werden kann. Folgende Nachrichten können diesen Weg nutzen. Das reduziert gegenüber wiederholtem Flooding die Zahl unnötiger Aussendungen.

Pfad im Paket

Die aktuelle Packet-Format-Dokumentation speichert den Pfad als Folge kompakter Hash-Werte. Zusätzlich kodiert das Path-Length-Feld sowohl die Zahl der Hashes als auch deren Größe.

Das Paketformat unterstützt aktuell 1-, 2- und 3-Byte-Path-Hashes; ein weiterer Code ist reserviert. Die gesamte Pfaddatenmenge ist zusätzlich begrenzt.

Rückwege

Die offizielle Payload-Spezifikation kennt Returned-Path-Pakete. Sie können dem ursprünglichen Absender einen Rückweg beschreiben und zusätzliche Informationen wie eine Bestätigung bündeln.

Pfade können veralten

Ein direkter Pfad ist keine dauerhafte Garantie. Wird ein Repeater abgeschaltet oder bewegt sich ein beteiligter Knoten, kann der gespeicherte Weg nicht mehr funktionieren.

Deshalb ist mobile Repeater-Infrastruktur problematisch: Andere Teilnehmer können einen Pfad lernen, dessen Zwischenstation später an einem ganz anderen Ort steht.

Kürzer ist nicht immer ausschließlich nach Hopzahl besser

Ein Pfad mit weniger Hops kann eine schwache Einzelstrecke enthalten, während ein etwas längerer Weg über stabile Sichtverbindungen zuverlässiger sein kann. Routingqualität lässt sich deshalb nicht allein aus der Anzahl der Zwischenstationen ableiten.

Diagnose

Bei wiederkehrenden Zustellproblemen sollte geprüft werden, ob eine direkte Funkstrecke funktioniert, ob ein alter Pfad verwendet wird und ob die beteiligten Repeater noch erreichbar sind. Änderungen sollten einzeln vorgenommen werden, damit die Ursache nachvollziehbar bleibt.

Quellen, Rechte und redaktionelle Prüfung

- [MeshCore – offizielle Dokumentation](#)
- [meshcore-cli – offizielles Projekt](#)
- [MeshCore Wiki DE – ergänzende Communityquelle](#)

Redaktioneller Hinweis: Diese Seite ist eine eigenständige Darstellung für MeshCore Hessen. Sachinformationen wurden anhand der oben genannten Quellen geprüft. Texte, Bilder und die Gliederung des MeshCore Wiki DE wurden nicht übernommen; die Seite wird nur als ergänzende Communityquelle zur Faktenkontrolle genannt. Für technische Details hat die aktuelle offizielle MeshCore-Dokumentation Vorrang. Deutschlandbezogene Funkrechtsangaben werden an den Veröffentlichungen der Bundesnetzagentur geprüft.

Urheberrecht: Technische Begriffe, Protokollnamen, Befehlsnamen, Zahlenwerte und andere notwendige Sachangaben werden unverändert verwendet, soweit dies für fachliche Richtigkeit erforderlich ist. Es werden keine längeren fremden Textpassagen oder Bilder reproduziert.

Prüfstand: 18. August 2026

Routing-Grundlagen

Routing beantwortet die Frage, wie ein MeshCore-Paket vom Sender zum Ziel beziehungsweise in den gewünschten Netzbereich gelangt. Die aktuelle Paketformat-Dokumentation unterscheidet Flood Routing und Direct Routing sowie Varianten mit Transport Codes.

Flood Routing

Bei Flood Routing wird kein einzelner vollständiger Zielpfad vorausgesetzt. Geeignete Repeater können ein Paket weitertragen. Das ist robust für Discovery und Verteilung, erzeugt aber mit jedem weiterleitenden Repeater zusätzliche Airtime.

Direct Routing

Bei Direct Routing enthält das Paket einen Pfad. Die Repeaterkennungen dieses Pfads werden kompakt als Path Hashes gespeichert. Ein Paket soll dadurch entlang eines bekannten Weges weitergegeben werden, statt überall geflutet zu werden.

Transport Codes

Aktuelle Paketformate kennen zusätzlich Transport Codes. Sie werden bei den Transport-Flood- und Transport-Direct-Routentypen verwendet und bilden die technische Grundlage für Regions/Scopes.

Pfadlänge

Das aktuelle Paketformat kodiert in einem Byte sowohl die Zahl der Path-Hash-Einträge als auch deren Größe. Die unteren sechs Bits enthalten die Hash-Anzahl; die oberen Bits kodieren die Hash-Größe.

Damit ist eine Aussage wie „MeshCore hat immer genau sieben Hops“ falsch. Die technische Obergrenze des aktuellen Formats liegt wesentlich höher, während die praktisch sinnvolle Pfadlänge von Netz, Airtime und Zuverlässigkeit abhängt.

Routing ist kein Selbstzweck

Ein Netz ist nicht besser, nur weil eine Nachricht möglichst viele Repeater durchläuft. Jeder zusätzliche Hop benötigt Funkzeit und schafft eine weitere Stelle, an der sich Funkbedingungen ändern können.

Gute Netzplanung versucht deshalb, wenige stabile und sinnvoll gelegene Infrastrukturpunkte zu verwenden.

Dynamische Funkwelt

Ein bekannter Pfad ist eine Momentaufnahme. Mobile Knoten, abgeschaltete Repeater oder veränderte Funkbedingungen können einen Weg ungültig machen. MeshCore benötigt daher Mechanismen zur Pfadfindung und Aktualisierung.

Die folgenden Seiten trennen Flooding, direkte Pfade, Path Hashes, Hops und Regions/Scopes bewusst voneinander.

Quellen, Rechte und redaktionelle Prüfung

- [MeshCore – offizielle Dokumentation](#)
- [meshcore-cli – offizielles Projekt](#)
- [MeshCore Wiki DE – ergänzende Communityquelle](#)

Redaktioneller Hinweis: Diese Seite ist eine eigenständige Darstellung für MeshCore Hessen. Sachinformationen wurden anhand der oben genannten Quellen geprüft. Texte, Bilder und die Gliederung des MeshCore Wiki DE wurden nicht übernommen; die Seite wird nur als ergänzende Communityquelle zur Faktenkontrolle genannt. Für technische Details hat die aktuelle offizielle MeshCore-Dokumentation Vorrang. Deutschlandbezogene Funkrechtsangaben werden an den Veröffentlichungen der Bundesnetzagentur geprüft.

Urheberrecht: Technische Begriffe, Protokollnamen, Befehlsnamen, Zahlenwerte und andere notwendige Sachangaben werden unverändert verwendet, soweit dies für fachliche Richtigkeit erforderlich ist. Es werden keine längeren fremden Textpassagen oder Bilder reproduziert.

Prüfstand: 18. August 2026

Routing in der praktischen Netzplanung

Routingtechnik und Standortplanung gehören zusammen. Software kann keinen schlechten Funkweg wegkonfigurieren.

Ziel zuerst definieren

Vor einem neuen Repeater sollte klar sein, welche Aufgabe er erfüllen soll:

- einen Ort lokal versorgen
- zwei bisher getrennte Bereiche verbinden
- eine Geländeabschattung überbrücken
- einen vorhandenen Backbone robuster machen

Ohne konkretes Ziel lässt sich später kaum bewerten, ob der Repeater tatsächlich einen Nutzen bringt.

Wenige gute Standorte

Ein hoch gelegener Standort mit guten Sichtbeziehungen kann mehrere schlecht positionierte Repeater ersetzen. Umgekehrt ist „hoch“ allein keine Garantie: Antenne, lokale Abschattung, Gebäudestruktur und Störungen bleiben relevant.

Flood-Domänen begrenzen

Mit wachsender Infrastruktur sollte nicht jeder lokale Flood automatisch das gesamte erreichbare Netz durchlaufen. Regions/Scopes ermöglichen eine logische Begrenzung zusätzlich zur physischen Funkstruktur.

Redundanz

Ein einziges perfektes Relais kann einen großen Bereich verbinden, bildet aber zugleich einen Single Point of Failure. Sinnvolle Redundanz bedeutet nicht, jeden Standort mehrfach zu kopieren, sondern alternative belastbare Wege für wichtige Netzbereiche zu schaffen.

Mobile Repeater vermeiden

Ein Repeater, der regelmäßig seinen Standort wechselt, erschwert die Verwendung stabiler Pfade. Persönliche mobile Geräte sollten daher nicht ohne klare Notwendigkeit als Infrastruktur betrachtet werden.

Dokumentation

Für Community-Infrastruktur sollten mindestens dokumentiert werden:

- Knotenname
- Aufgabe beziehungsweise Versorgungsziel
- grober Standortbereich
- Hardware
- Firmwarestand
- Antennenkonzept
- Regionskonfiguration
- Datum wesentlicher Änderungen

Geheime Administrationsdaten und private Schlüssel gehören nicht in die öffentliche Dokumentation.

Messen statt schätzen

Geländemodelle und Sichtlinienberechnungen helfen bei der Vorauswahl. Eine reale Funkmessung mit der vorgesehenen Antenne und Konfiguration bleibt jedoch entscheidend.

Reichweitenwerte anderer Standorte sind Vergleichswerte, keine Garantie.

Quellen, Rechte und redaktionelle Prüfung

- [MeshCore – offizielle Dokumentation](#)
- [meshcore-cli – offizielles Projekt](#)
- [MeshCore Wiki DE – ergänzende Communityquelle](#)

Redaktioneller Hinweis: Diese Seite ist eine eigenständige Darstellung für MeshCore Hessen. Sachinformationen wurden anhand der oben genannten Quellen geprüft. Texte, Bilder und die Gliederung des MeshCore Wiki DE wurden nicht übernommen; die Seite wird nur als ergänzende Communityquelle zur Faktenkontrolle genannt. Für technische Details hat die aktuelle offizielle MeshCore-Dokumentation Vorrang. Deutschlandbezogene Funkrechtsangaben werden an den Veröffentlichungen der Bundesnetzagentur geprüft.

Urheberrecht: Technische Begriffe, Protokollnamen, Befehlsnamen, Zahlenwerte und andere notwendige Sachangaben werden unverändert verwendet, soweit dies für fachliche Richtigkeit erforderlich ist. Es werden keine längeren fremden Textpassagen oder Bilder reproduziert.

Prüfstand: 18. August 2026