WLAN Pros Toolbox · Field Manual

Compiled 2026-07-02 · Field & Trade Reference added 2026-07-05 · covers 173 tools · app v1.7.0

This field manual documents every tool in the WLAN Pros Toolbox, drawn directly from the help text that ships inside the app. Each entry states what the tool does, why it is in the kit, how to drive it, the inputs it takes, the formula or method behind it where one applies, a worked example where one helps, and the field notes that keep you out of trouble. Tools are grouped and ordered the same way they appear in the app, so you can navigate the manual and the Toolbox the same way. Every figure and method is the one the app actually runs.

Contents

  - Codes & Safety (6)
  - AEC & Documentation (4)
  - Compliance & Governance (2)
  - Wireless Landscape (1)
  - Verticals (4)
  - Vendor & Hardware (2)
  - Calculators (1)
- **Field conveniences** (5 handy tools: reference, not curriculum)

Test Network (6 tools)

Live Wi-Fi and internet diagnostics. These tools read the device's real connection, namely the associated AP, link rates, signal, and throughput, and answer the everyday question of whether a slowdown is the Wi-Fi or the internet.

Test My Connection

Answer the everyday question "is the slowdown my Wi-Fi or my internet?" in plain English, and tell the user what to say to support.

Why it's here. The consumer-facing front door. Same backends and verdict engine as the pro Wi-Fi vs Internet tool, re-skinned for a non-technical user. Reach for it when you want a one-tap answer plus vetted self-help steps, not engineer numbers.

How to use

1. Open the tool and tap Run.
2. On iOS, install the companion Shortcut first if prompted; the Wi-Fi link read comes from it (see Wi-Fi Information). Without it, the tool still measures the internet and degrades honestly.
3. On macOS, granting Location is optional; it only affects whether the SSID name shows. The verdict never needs Location (it is rate-based). On macOS the app can proactively surface the native "Allow Location" prompt so the name resolves without a settings trip.
4. Read the two status chips (Wi-Fi / Internet) and the headline.

Formula or method. It runs two things in one pass and feeds both into the shared verdict engine: a connected-link read via WifiInfoSourceResolver → MacWifiInfoAdapter (macOS CoreWLAN) or WiFiDetailsBridge → ConnectedAp.fromWifiDetails (iOS Shortcut); a full net_quality run via OwnEngineQualityClient.forHost('one.one.one.one'). It translates the net_quality grades into the engine's internet-health flag via the shared ConnectionCheck.internetHealth: GOOD only when download AND upload AND latency AND loss all grade good or excellent; otherwise marginal. The engine's five engineer verdicts are collapsed into four consumer outcomes plus a two-axis chip status (Wi-Fi / Internet each "Fine", "Slow", or "Couldn't check") by ConsumerVerdictMapper. Outcomes: A "Looks like your Wi-Fi"; A-lead "Mostly your Wi-Fi"; B "Looks like your Internet"; C "Both look fine"; D1 "Couldn't check everything" (internet measured, Wi-Fi not); D2 "Couldn't complete the check" (neither measured).

Field notes

- Platform differences: macOS reads the link via CoreWLAN (Tx rate, RSSI/SNR, no Rx). iOS reads it via the Shortcut (Tx and Rx). If the link can't be read (wired, or iOS without the Shortcut), it falls to the

D1/D2 honest path; it does not guess a side. D1 keeps the real internet figure it did measure; D2 admits neither side was read.

- The headline is a hedge by design ("Looks like…", not "your Wi-Fi is broken"). The two chips teach the model that Wi-Fi and internet are two separate things. "Both look fine" is the most useful real-world answer; it points the user at the specific app/site instead of the connection.
- Cloud Apps reachability panel: the result column also mounts a Cloud Apps panel that TCP-connects (port 443) to the recurated kCloudApps named-service list and reports each as reachable or "unreachable" (a word + glyph, never color alone). It proves the service edge/CDN is reachable and times that hop; it is NOT a measure of in-app call or stream quality, and the caption says so. It reuses the same shared ReachabilityProbe the Network Quality tool runs.
- Analyze Results sits beside Copy on the result screen: Copy saves the raw report, Analyze explains the verdict and recommends next steps, entirely on-device (see Analyze Results).
- The verdict copy is deliberately non-diagnostic for a layperson. The underlying numbers and bands are the same as Wi-Fi vs Internet; if you want the numbers, use that tool.
- How it picks a side: it compares how much internet speed you're actually getting against how much your Wi-Fi link could realistically carry. When the internet measures fine on its own, it says both look fine; otherwise it leans toward Wi-Fi or internet depending on which one has the headroom to spare.
- D2 ("Make sure you're on Wi-Fi and try again") is the honest answer when nothing measured, never a fake zero.

Analyze Results

Turn a finished Test My Connection result into a plain-language, conclusion-first report: what your connection is telling you and what to do about it, ordered by importance.

Why it's here. Test My Connection gives the headline verdict and the raw numbers; Analyze Results explains them. Reach for it when you want the why behind the verdict and the recommended next steps, evaluated entirely on the device. It sits alongside Copy on the result screen: Copy saves the raw report for support, Analyze explains it.

How to use

1. Run Test My Connection first so there is a measured result.
2. Tap "Analyze my results" beneath the verdict.
3. Read the findings top to bottom: the most important (the verdict, an open network, packet loss, a dead path) lead, then signal/band/internet-quality context. Each carries a severity word (ACTION / WORTH A LOOK / CONTEXT) plus a color, never color alone.
4. Use the Copy action (top right) to save the whole analysis as plain text.

Formula or method. A pure, local rule engine (AnalyzeEngine) evaluates a structured rule library against the same result data already on screen, read directly from the in-memory models (ConnectedAp, the net_quality result, the Wi-Fi-vs-Internet verdict, the DNS probe, and the cloud reachability tally), never by re-parsing the copied report text. Every threshold is imported from the app constants, not duplicated: RSSI/SNR from WifiGradingBands,

latency/jitter/loss/responsiveness/download from QualityScoring, the verdict thresholds from the Wi-Fi-vs-Internet engine, and the security labels from WifiSecurity. Fired rules are sorted by severity (P1 critical, P2 important, P3 context) then by the rule library's declaration order, so the verdict always leads. Context-only reassurances ("no problem here") are suppressed unless a real finding also fired, so the report never narrates a non-issue. Runs entirely on the device: no network call, nothing stored, nothing leaves the device.

Field notes

- Doctrine guardrails: the "you're on 2.4 GHz" (R-20) and "narrow channel width" (R-23) rules deliberately carry the honest trade-off and never tell a user to blanket-switch bands or force 160 MHz: wider is not automatically better, and 2.4 GHz is the right choice for range and IoT.
- Null discipline: a field the platform did not measure fires no rule. The honesty rules (channel width not captured on iPhone; Wi-Fi signal details not captured) explain the gap instead of guessing.

Network Quality

A one-shot transport-quality measurement covering latency, jitter, loss, download, upload, and responsiveness, plus a reachability check against a set of popular cloud apps. Each dimension is graded on its own; there is deliberately no single composite "score".

Why it's here. When you want to characterize a connection's transport behavior the way Apple's networkQuality or an Orb does, measured by the app's own engine. Reach for it to see whether the internet path is healthy across multiple axes, not just "how many Mbps".

How to use

1. Open the tool. A live latency trend starts sampling immediately (every 30s) while the screen is open.
2. Tap "Run test" for the full one-shot measurement (download/upload/responsiveness run only on a full run).
3. Read the six graded rows and the popular-sites reachability table.

Formula or method. All this app's own engine. Latency / jitter / loss: 10 sequential TCP-connect RTTs to one.one.one.one:443 (not ICMP, the sandbox blocks raw sockets). Jitter is RFC-3550-style mean deviation between consecutive samples; loss is failed-connects ÷ attempts × 100. With zero successful samples, latency and jitter report "Unavailable" but loss is a real 100%. Download: parallel-summed, multi-server. Several concurrent streams (5 by default) share one window of about 15 seconds, each against a different independent public server or CDN from a diverse pool (Cloudflare, OVH, Hetzner, Cachefly, ThinkBroadband). The first few seconds of TCP slow-start ramp are discarded, so the number is sustained steady-state throughput, not the ramp. Each server's own rate is measured; any server that comes back below roughly half the median (a throttled or slow outlier) is dropped, and the survivors' rates are summed into the aggregate download figure, so one slow server cannot drag the result down. If every stream fails, it raises an honest "couldn't measure", never a fake 0 Mbps. Upload: single stream with multi-CDN fallback (only Cloudflare __up is a verified large-POST sink, honest single-stream, not faked parallelism). A non-2xx or empty transfer is an honest failure, not 0.

Responsiveness (RPM): a simplified single-flow loaded-latency estimate inspired by RFC 9097 / Apple networkQuality, NOT the full multi-flow RPM standard. It samples loaded RTT while a download flow runs, then RPM = 60000 / loadedAvgMs. Reachability: TCP-connect (port 443) to 14 well-known cloud-app hosts, a mix the public and a WLAN pro both recognize: social/consumer (Facebook, Instagram, TikTok, YouTube, Netflix) plus pro/infra (Google, iCloud, Microsoft 365, Cloudflare, AWS, Zoom, Slack, GitHub) and a Cloudflare DNS anchor. Grade bands: Latency ms: Excellent <20, Good <50, Fair <100, Poor ≥100 (grounded in ITU-T G.114, our cut points). Jitter ms: <5/<15/<30. Loss %: 0/<1/<2.5. Responsiveness RPM: ≥1000/≥500/≥100. Download Mbps: ≥100/≥25/≥5 (explicitly a heuristic). Upload Mbps: ≥20/≥5/≥1 (heuristic).

Field notes

- Platform differences: runs on macOS, Windows, Linux, Android, iOS over dart:io sockets/HTTP. On web it routes to the download-the-app fallback (no sockets).
- Read each grade word on its own; a connection can be Excellent on latency and Poor on upload at the same time, which is the point. The download/upload Mbps grades are "good enough for a household" heuristics, not standards. RPM is directional only.
- Latency uses TCP handshake RTT, so it includes the full SYN/SYN-ACK round trip to a real host (a faithful proxy, slightly higher than ICMP).
- The on-screen footnote states plainly: "these are this app's own measurements, not an Orb or Ookla score." Download is summed-parallel (so it reflects aggregate link capacity, not a single flow). A failed measurement is "Unavailable" with a note, never 0.

Wi-Fi Information

Show the live connected-AP link details: SSID, BSSID, RSSI, noise, SNR, Tx/Rx rate, channel, width, band, 802.11 standard.

Why it's here. The "what is my radio actually doing right now" read. Reach for it to confirm band/channel/width, check signal and SNR against design targets, or sanity-check a client's negotiated rate.

How to use

1. macOS: open the tool; it pulls a CoreWLAN snapshot. Tap Refresh to re-read. SSID and BSSID are gated behind macOS Location Services; tap Grant (or open the Location settings pane) to reveal them. The radio metrics do not need Location.
2. iOS: this is live streaming only. Install the combined "WLAN Pros Live" companion Shortcut (one-time, via the iCloud link), then tap Start. The Shortcut loops, harvesting the connected AP each cycle via the stock "Get Network Details" action and handing JSON back to the app; Stop freezes the last values. There is no one-tap snapshot on iOS.

Formula or method. All sources normalize into one ConnectedAp model. macOS comes from CoreWLAN via the com.wlanpros.toolbox/wifi_info native channel; iOS comes from the Shortcut JSON parsed by WiFiDetails (case-insensitive keys, tolerant numeric parse). In iOS Live mode, RSSI and SNR

get hard grades against the bands below; Tx/Rx rates get a trend (Rising/Falling/Steady) rather than a hard grade, because a "good" data rate is entirely relative to band/width/MCS. RSSI bands (dBm): Excellent ≥-59 (>-60), Good ≥-67, Fair ≥-72, Poor below. SNR bands (dB): Excellent ≥36 (>35), Good ≥25, Fair ≥15, Poor below.

Field notes

- macOS (CoreWLAN): exposes SSID/BSSID (Location-gated), RSSI, noise, SNR (reported directly), Tx rate, PHY mode → standard label, channel, channel width, band (reported), country code, interface name, hardware MAC. It does NOT expose the Rx rate or Tx power (public CoreWLAN limitation); those render "Not exposed by macOS CoreWLAN", never estimated.
- iOS (Shortcut): exposes SSID, BSSID, channel, RSSI, noise, standard, Rx rate AND Tx rate. SNR is derived app-side (rssi – noise) and band is derived from the channel number (both labeled "derived"). The harvest does NOT return channel width, so width renders "Not reported by iOS", never fabricated. Channel→band derivation: 1 to 14 → 2.4 GHz, 36 to 177 → 5 GHz, 181 to 233 → 6 GHz; the ambiguous low 6 GHz range (1 to 93) is read as 2.4/5 GHz and never silently claimed as 6 GHz.
- Android / Windows: honest "coming in a later update" state (clean seam, not built). Web: download-the-app fallback.
- RSSI and SNR carry hard grades; read the rate as a trend, not a pass/fail. The standard label combines the 802.11 designation and Wi-Fi generation (e.g. "802.11be (Wi-Fi 7)"; 802.11ax on 6 GHz shows Wi-Fi 6E). Anything marked "derived" was computed, not read from the radio.
- macOS Tx-only and iOS no-width are real platform ceilings, not bugs. On macOS, if the SSID is blank, it's almost always a missing Location grant, not a hidden network. The macOS read has a 5s hang-safety: a stalled CoreWLAN read surfaces an honest "No Wi-Fi reading" rather than freezing.

Cellular Information

Show the iPhone's mobile-network details: carrier, radio technology, signal bars, country code, roaming.

Why it's here. A quick read of what the cellular radio is doing, useful when comparing Wi-Fi offload behavior or confirming a device fell back to LTE/5G.

How to use

1. iOS only. Install the same combined "WLAN Pros Live" companion Shortcut, then Start; the Shortcut harvests cellular details via "Get Network Details" and hands them over the App Group bridge.

Formula or method. Parses the Shortcut JSON into CellularInfo (case-insensitive keys, with whitespace-trim tolerance after a real on-device bug). Fields: carrier name, radio technology (mapped from raw CTRadioAccessTechnology* constants to friendly labels like "5G (NSA)", "LTE"; unknown values pass through, never blanked), signal bars (coarse 0–4 status-bar indicator), country code, roaming bool.

Field notes

- Platform differences: iOS is the only source. There is deliberately no native CoreTelephony path: CTCarrier is deprecated since iOS 16.4 and returns placeholder junk, and cellular signal strength (RSRP/RSRQ/dBm) is private-API-only and an App Store rejection. So data comes only via the Shortcut. macOS (no radio), Android, Windows show an honest "not available on this platform"; web shows the download fallback.
- Signal bars are bars (0 to 4), never relabeled dBm/RSRP/RSRQ; the app does not have a raw signal value and must not imply it does. Bars are clamped to 0 to 4; an out-of-range value is never trusted.
- This is the one tool where iOS exposes more than a native app could (the Shortcut runs in Apple's privacy context). A missing field renders "Unavailable", never a fabricated value.

Roaming Log

Record each time your device roams from one access point (BSSID) to another on the same network (SSID) during an open monitoring session, with the timestamp, the from→to BSSID pair, and the signal at the roam.

Why it's here. A roam is a BSSID change while the SSID stays the same. Watching the live Wi-Fi signal shows a roam happen, but nothing recorded it. Reach for this when you want a timestamped list of roams from a walk-around, e.g. to see where the device clings to a far AP or hops too often.

How to use

1. macOS: opens recording automatically. It polls the Wi-Fi link continuously while the screen is open. Walk your space; each roam is logged as it happens.
2. iOS: install the "WLAN Pros Live" companion Shortcut (same one Wi-Fi Information uses), tap Start, then walk with the screen open. Firing the Shortcut switches to the Shortcuts app, so it waits for your deliberate Start tap rather than auto-firing.
3. Read the list newest-first: time · network, the from→to BSSID pair, and the signal (RSSI/SNR) read at the roam.

Formula or method. The shared WifiSignalSampler feeds every fresh connected-AP sample to a pure RoamDetector. The detector records a roam only when the current BSSID differs from the prior non-null BSSID AND the SSID is unchanged. A null/blank BSSID breaks the chain without fabricating a roam; the first known BSSID seeds the anchor (no roam); a same-BSSID sample is ignored; a changed SSID is a network switch, not a roam, and is excluded. On macOS the detector is fed on every poll (before the sparkline's unchanged-RF guard) so a roam at identical signal still registers; on iOS it is fed from each new streamed payload and reset on a fresh Stop→Start.

Field notes

- Foreground session only on iOS. There is no public iOS API for background Wi-Fi or BSSID-change callbacks, so no app can log roams with the app closed or the phone in a pocket: the same ceiling Wi-Fi Check shares. macOS records continuously while the screen is open.
- BSSID is the AP radio's MAC address. A roam between two radios in the SAME physical AP (e.g. 2.4 GHz → 5 GHz) is still a BSSID change and is logged; that is correct: it is a real roam.

- The signal shown is the reading at the moment the new BSSID first appeared, not an average. "Signal unavailable" prints honestly when the platform omitted RSSI for that sample, never a fake value.
- Stopping and starting (iOS) clears the session log so a new walk does not inherit the prior walk's roams.
- Copy-to-save: the §8.16 Copy action in the app bar exports the whole recorded roam session as paste-ready plain text (each roam with its time, network, from→to BSSID pair, and signal), built by the pure buildRoamLogCopyText so a walk can be saved to a ticket or note.

How the Toolbox Measures Throughput

The Toolbox reports two throughput numbers, and they answer two different questions. One measures your path to the internet. The other measures the Wi-Fi hop between your device and the Access Point. Read each as its own answer and both make sense. Compare them to each other, or to a headline speed test, and you will only confuse yourself.

Keep them straight from the start.

Two numbers, two questions

- **Internet throughput** is shown by *Test My Connection* and *Network Quality*. The headline figure is your download, what people mean by internet speed, measured against the wider internet right now. Upload is measured too and shown as its own separate number.
- **Usable Wi-Fi capacity** is shown by *Wi-Fi vs Internet*. This is roughly how much throughput the Wi-Fi hop itself can carry, based on the rate your radios negotiated.

Same app, two measurements, two purposes. Everything below explains what each one actually does and why a number you see somewhere else may not match.

How the internet number is measured

The internet download figure comes from an aggregate-capacity measurement, the same class of test the well-known speed tests run. It is built to be comparable to them, not built to read low.

Here is what the app actually does:

- **Download opens several streams at once, to a diverse set of independent public servers.** Five streams by default, each against a different provider or CDN, never a single provider. Running parallel streams across independent networks is how you fill a fast connection, and how you keep one provider's rate-limiting from collapsing the whole measurement.
- **The window runs about 15 seconds, and the ramp-up is thrown away.** The first few seconds are TCP slow-start, when the connection has not yet reached full speed. The app discards them and reports the sustained, steady-state rate, not the ramp.
- **Each server's own rate is measured, and slow outliers are dropped.** Any server that comes back far below the pack, below roughly half the median rate, is treated as throttled and excluded. The

surviving servers' rates are summed into the aggregate download figure, so one bad or throttled server cannot drag the result down.
- **The reported internet number is the download.** That is what people mean by internet speed. Upload is measured too, over its own window, and shown as its own separate number.
- **When it cannot get a clean measurement, it says so.** You get an honest "couldn't measure," never a fake 0. A zero on a working connection would be a lie, and the app will not tell it.

Responsiveness, the RPM figure, is a single-flow loaded-latency reading taken while a download runs. It holds up well across servers, so it stays steady even when the throughput number moves.

Privacy stays simple. These are anonymous downloads from public servers. Nothing about you is sent or published, and the whole measurement runs on your device.

Why your speed test may read differently

No single number is "the" speed of a connection. Throughput is a measurement, and every measurement depends on how it was taken. Run two honest tests on the same connection and they can land on different numbers for reasons that have nothing to do with either one being broken.

What moves the number:

- **Which server you hit, and how well it is peered to your ISP.** A single nearby edge cache that your provider peers with directly has a shorter, cleaner path than a spread of independent public servers out on the wider internet.
- **How many streams the test opens, and to how many destinations.** More streams across more independent networks measure real aggregate capacity; a single stream to one close cache often posts a higher-looking peak.
- **Whether the ramp-up counts.** A test that includes the first seconds of TCP slow-start can report a different number than one that throws the ramp away and measures only the sustained rate, the way this app does.
- **Time of day and congestion.** The same connection is not the same connection at 3 a.m. and at 8 p.m.

Put those together and the pattern is easy to see. A one-number tool that saturates the single nearest edge cache can post a higher figure than a steady-state measurement summed across several independent public servers. That gap is a difference in method. Both tests measured something real. This app measures the honest, sustained throughput to real internet destinations, which is the number that matches what your connection actually delivers day to day.

What the Wi-Fi number is, and what it is not

The *Wi-Fi vs Internet* tool shows a different figure entirely: usable Wi-Fi capacity, which the app estimates at about 55% of the negotiated Wi-Fi link rate.

Two things to understand about that 55%.

First, it is deliberately below the headline link rate, and that is correct. The rate your device and the Access Point negotiate is a ceiling, not a delivery. Real-world throughput over Wi-Fi runs roughly 50 to 60% of that negotiated rate once you account for protocol overhead, acknowledgments, contention, and retries. Taking 55% of the link rate is a reasonable estimate of what the hop can actually carry.

Second, this number describes the Wi-Fi hop only. It is the road between your device and the Access Point, and nothing past it. It says nothing about your internet speed.

Do not compare the Wi-Fi capacity number to an internet speed test. Different road, different question. One tells you how much the local Wi-Fi hop can move; the other tells you how much your connection to the internet can move. The whole point of putting them side by side in the tool is to show you which of the two is the limiting factor, not to make them agree.

What our internet number represents

When the Toolbox shows you an internet download figure, it is real, sustained throughput to a diverse set of neutral public servers, measured over about a 15-second window with the slow-start ramp discarded and throttled outliers dropped. It is not the highest number the app could have coaxed out of a single hand-picked nearby cache, and it is not padded to look good next to anyone else's result.

If another test on the same connection reads higher, that is the method talking: a single nearby edge cache, or a run that counts its ramp-up. If the Toolbox cannot get a clean read, it tells you it couldn't rather than inventing a number.

Two throughput numbers, two questions. The internet figure is your honest path to the wider internet. The Wi-Fi figure is the capacity of the hop to your Access Point. Keep them straight and each one tells you exactly what it means.

Networking Tools (25 tools)

Socket, lookup, and scan utilities for working a network from the device in hand. Ping, traceroute, port and host discovery, DNS and registry lookups, and packet-level senders and inspectors.

Device Info

Show the device's own system facts: model (marketing name plus the raw identifier), total physical memory (RAM), system uptime since the last boot, and the cellular IP address where the device has a cellular interface.

Why it's here. The companion to Interface Information: that tool answers "what's my IP, gateway, and Wi-Fi link"; this one answers "what device is this, how much RAM, how long since boot, and what's my cellular IP". Reach for it to confirm the hardware model or to read the cellular-side address that the Wi-Fi-centric interface view doesn't surface.

How to use

1. Open the tool; it reads a snapshot. Each field a platform can't provide renders its honest unavailable state rather than a fabricated value.
2. Tap Refresh in the top bar to re-read (uptime advances; the rest is stable).
3. Use Copy to put the whole snapshot on the clipboard as labeled text.

Formula or method. Model and total memory come from the device_info_plus package (BSD-3): on iOS, modelName (the package maps utsname.machine, e.g. iPhone16,2, to a marketing name) plus physicalRamSize; on macOS, modelName/model plus memorySize. Uptime comes from a tiny native MethodChannel (com.wlanpros.toolbox/system_info → systemUptime) reading ProcessInfo.processInfo.systemUptime: no package exposes it. The cellular IP comes from dart:io NetworkInterface.list, matching the conventional iOS cellular interface name pdp_ip0.

Field notes

- Cellular IP uses a heuristic: Apple does not treat interface names as a stable API, so detection matches the conventional iOS cellular name pdp_ip0. "No cellular interface" is the normal, honest result on a Wi-Fi-only iPhone, in airplane mode, or on a Mac (which has no cellular interface).
- Total memory is shown in binary units (an 8 GiB device reads "8 GB") to match how RAM is physically sized; the label stays the familiar GB/MB.
- Uptime is seconds since boot, formatted as "3d 4h 12m"; it always shows at least minutes (a just-booted device reads "0m").
- Nulls are honest "not available", never 0 or "". On Android, model is shown but total RAM is not surfaced by the package, so it reads unavailable.

Inspector (HTTP Header)

Issue a HEAD or GET, follow and record the redirect chain hop-by-hop, and return the final status plus all response headers.

Why it's here. Shows the full 301→302→200 story, not just the destination. Reach for it to debug redirects, security headers, caching, or CDN behavior.

How to use

1. Enter a URL (bare host assumes https://), pick HEAD (default) or GET, inspect. Each hop is shown with its status, Location, and headers.

Formula or method. Sets followRedirects = false and follows the chain itself, recording one hop per response until a non-redirect status or the 10-redirect cap. HEAD→GET fallback: if HEAD returns 405 (and the caller didn't demand GET), it transparently retries that hop with GET and notes the fallback. Relative Location values are resolved per RFC 7231.

Field notes

- Platform differences: iOS App Transport Security blocks cleartext http://, and the app deliberately does not add a blanket ATS exception (that would weaken every request). So an http:// target fails at the socket layer on iOS, and the tool surfaces a specific message ("On iOS, cleartext HTTP is blocked by ATS, try the https:// URL") rather than a generic failure. Otherwise identical on native; gated off on web.
- The hop chain reads top (first request) to bottom (final response). headFellBackToGet and redirectLimitHit flags are surfaced. Header names are title-cased and sorted.
- Bodies are drained and discarded; this is a header inspector, not a fetcher. On iOS, use the https:// URL.

Inspector (SSL/TLS)

Connect to host:port over TLS and report the server certificate as inspectable data, including expired, self-signed, and name-mismatch certs.

Why it's here. A field cert inspector. Reach for it to read validity, SANs, fingerprints, key size, and the issuer on any TLS service, especially broken ones.

How to use

1. Enter a host (URLs are accepted and stripped to the host), optional port (443 default), inspect.

Formula or method. SecureSocket.connect with onBadCertificate: (cert) => true: it accepts any certificate at the socket layer so an expired/self-signed/mismatched cert is still captured and shown; the validity verdict is computed from the cert dates, not thrown as an error. A bad cert is a successful inspection, not a failure. Field coverage is split: dart:io X509Certificate gives PEM/DER, subject/issuer

DN, notBefore/notAfter, SHA-1; basic_utils X509Utils re-parses the PEM to recover structured DN, SAN list, serial, signature algorithm, public-key algorithm + bits, and SHA-256. Validity is computed against "now": valid / expired / not-yet-valid + days-to-expiry.

Field notes

- Platform differences: same on every native platform. Gated off on web.
- The validity state is an icon + text (not color alone). Two honest limits are stated on-screen, not faked: (1) ALPN ≠ TLS version/cipher, since dart:io exposes only the ALPN result, not the negotiated TLS version or cipher suite, so those are reported "not exposed by the platform", never invented. (2) Leaf only, since dart:io hands over only the leaf certificate, not the intermediate/root chain it validated against.
- Connection problems (DNS, refused, timeout) are failures; an invalid cert is not. Don't read the ALPN line as the TLS version; the tool deliberately separates them.

Interface Information

Show the device's own network state: per-interface IPv4/IPv6 addresses, interface name and inferred type, plus gateway, DNS, Wi-Fi SSID/BSSID, and the device's primary IP where the platform exposes them.

Why it's here. The "what's my IP, gateway, and link" foundation. Reach for it first when you need the device's own address (e.g. before a ping/sweep) or to confirm which interface is active.

How to use

1. Open the tool; it reads a snapshot. Each field that a platform can't provide returns null and renders "Not available on this platform".

Formula or method. Built on dart:io NetworkInterface.list for the address/interface table, plus network_info_plus for Wi-Fi-link details (SSID, BSSID, gateway, subnet mask, Wi-Fi IPv4/IPv6) that dart:io doesn't expose. Each sub-read is independently guarded so one denied call (e.g. SSID) never blanks the whole screen. Interface kind (Wi-Fi/Ethernet/Cellular/Loopback/VPN/Other) is a heuristic from the OS interface name, e.g. macOS en0 is guessed Wi-Fi, en1+ Ethernet; explicitly conservative. primaryIPv4 prefers the Wi-Fi link IP, else the first non-loopback IPv4.

Field notes

- Platform differences: SSID/BSSID/gateway depend on network_info_plus and OS permission state. iOS needs the wifi-info entitlement + Location for SSID/BSSID; macOS/Android vary. Web is gated off. SSID is cleaned of wrapping quotes and the placeholder.
- The interface "type" is a name-based guess; on macOS the en0/en1 split is heuristic, so trust the addresses over the label on unusual hardware.
- Nulls are honest "not available", never 0 or "". Gateway/DNS exposure is platform-and-permission dependent.

IP Geolocation

Country, region, city, coordinates, timezone, ISP/org, and ASN for an IP (or your own public IP).

Why it's here. Quick geo + ownership context for an address. Reach for it to see where an IP resolves and who runs it.

How to use

1. Enter an IP or hostname, or leave blank to locate your own public IP, look up. A copyable "lat,long" and an OpenStreetMap URL are offered.

Formula or method. Queries ipwho.is (free, no key, HTTPS, chosen because it's keyless AND returns ASN + timezone in one response; ip-api.com is HTTP-only and would trip iOS ATS, so it's explicitly not used). An empty query hits https://ipwho.is/ (your IP); otherwise https://ipwho.is/{ip}. A cheap client-side sanity check rejects obvious junk before spending a round-trip; ipwho.is's in-band {"success": false} is mapped to a precise "check your input" or rate-limited state, not read as OK.

Field notes

- Platform differences: JsonHttpClient → dart:io, native-only; web gated to the download fallback (ipwho.is CORS unverified).
- Coordinates are shown as selectable mono data plus a copyable pair and an external maps link (no embedded interactive map; that's future). Every field is nullable → missing data renders "Not available".
- Geolocation accuracy is the provider's, especially for mobile/CGNAT IPs. A rate-limit response is surfaced explicitly ("wait a minute and try again").

Current Location

Reads the device's GPS location on open and shows latitude, longitude, altitude, and horizontal accuracy directly, no conversion step.

Why it's here. The fast "where am I right now" answer. The same fix is available inside the Lat / Long converter, but people expect a dedicated location tool rather than a coordinate calculator.

How to use

1. Open the tool; it requests Location permission (if needed) and reads one fix automatically.
2. Read latitude / longitude (decimal degrees), altitude (meters), and horizontal accuracy (meters).
3. Tap Update to read a fresh fix. Grant Location or open Settings if prompted.

Formula or method. Reuses the DeviceLocationService seam behind the Lat / Long tool (geolocator, LocationAccuracy.best, 15 s cap). Resolves Location permission/services first and renders the matching sealed state (granted -> fix; needs-permission -> request; blocked -> Settings deep-link; no-fix ->

honest unavailable). A genuine no-fix on a granted/enabled device falls back to an IP-derived approximate location, clearly labeled approximate (never presented as GPS). Coordinates render in Roboto Mono as identifier values.

Example. Granted + outdoors: 40.712800, -74.006000, altitude 12 m, accuracy 5 m.

Field notes

- Altitude and accuracy can read "Not reported" on hardware without GPS (e.g. a Mac on a coarse Wi-Fi fix).
- An approximate fix derived from your public IP is labeled as such and is city-level, not GPS-precise.
- No coordinates are transmitted; the IP-approximate fallback derives location from the public IP only.

IP Subnetting (IPv4)

Computes the full IPv4 subnet breakdown (network, broadcast, netmask, wildcard, first/last usable host, total addresses, and usable host count) from an address plus a CIDR prefix or a dotted mask.

Why it's here. CIDR math you'd otherwise do on paper or in your head at a whiteboard. Reach for it to confirm a network/broadcast boundary, a usable-host range, or a mask↔prefix conversion.

How to use

1. Enter an IPv4 address (e.g. 10.20.0.0). You can append the prefix inline as 10.20.0.0/22; an inline /prefix wins and the second field is ignored.
2. Otherwise enter the prefix or mask in the second field: a CIDR prefix (22 or /22) or a dotted mask (255.255.252.0).
3. The breakdown recomputes live on every valid keystroke. A host inside the subnet (e.g. 10.20.0.37/22) reports the same network as the base address.
4. Malformed input shows an inline "Check your input" card with a specific message rather than a wrong answer.
5. Use the Copy action in the app bar to copy the breakdown as a labeled text block.

Inputs

Input	Unit	Range
IPv4 address	dotted-decimal, optionally with inline /prefix	four octets each 0-255; inline /prefix 0-32
Prefix or mask	CIDR prefix or dotted mask	prefix 0-32, or a contiguous-ones dotted mask (e.g. 255.255.252.0); ignored when the address carries an inline /prefix

Formula or method. Pure-Dart 32-bit integer math on the masked network base. mask = (0xFFFFFFFF << (32 − prefix)) & 0xFFFFFFFF (:242-246); network = addr & mask (:134); broadcast = network | (~mask & 0xFFFFFFFF) (:135); wildcard = ~mask & 0xFFFFFFFF (:136); total = 2^(32 − prefix), with /0

= 2^32 (:137-139). Usable hosts by prefix: /0–/30 → total − 2 with first = network+1, last = broadcast−1 (:157-162); /31 → RFC 3021 point-to-point, usable = 2, no broadcast, both addresses are hosts (:151-156); /32 → single host route, usable = 1, first = last = the address, no broadcast (:146-150). A dotted mask is converted to a prefix only if it's a contiguous run of 1 bits (prefixFromMask,:218-237). Address parsing is strict: exactly four octets, each 0–255 (:199-211).

Example. 10.20.0.0/22 → netmask 255.255.252.0, wildcard 0.0.3.255, network 10.20.0.0, broadcast 10.20.3.255, first host 10.20.0.1, last host 10.20.3.254, total 1024, usable 1022. (This is the screen's seeded default.)

Field notes

- /31 (RFC 3021) is a point-to-point link: there is no network/broadcast reservation, so both addresses are usable hosts (usable = 2). The screen annotates this.
- /32 is a single-host route: one address, no range, no broadcast (usable = 1).
- A host address inside the block reports the subnet's network, not the host; all values derive from the masked base.
- A dotted mask must be a valid contiguous-ones mask; 255.0.255.0 is rejected as malformed.
- No network I/O, just pure math, so it runs on every platform including web.

IP Subnetting (IPv6)

From an IPv6 address and prefix length, derives the expanded and compressed forms, network address, first/last address in the prefix, host count, and RFC address type.

Why it's here. IPv6 subnetting and address-type identification during network design and troubleshooting, where 128-bit math is error-prone by hand.

How to use

1. Enter an IPv6 address (default 2001:db8::1).
2. Enter the prefix length (default 32).
3. Read expanded form, compressed form, network/prefix, first and last address, host count, and address type.

Inputs

Input	Unit	Range
IPv6 address	any valid IPv6 literal	exactly one:: run allowed; invalid format → inline error
Prefix	integer	0 to 128 (out of range → error)

Formula or method. Expand:: to full 8-group / 4-hex-digit form, rejecting more than one:: (:99-133). Pack to a 128-bit BigInt (toBigInt). mask = ((1 << prefix) − 1) << (128 − prefix) (0 when prefix 0); network = addr & mask; last = network | (~mask & 128-bit-mask) (:275-280). Compress to canonical:: by collapsing the longest run of all-zero groups (compressIPv6). Host count (:229-234): host bits = 128

– prefix; > 63 → "More than 2^{63}"; 0 → "1 address"; else 2^hostBits with grouping. Address type by prefix match (detectIPv6Type)::: unspecified,::1 loopback, fe80 link-local, fc/fd unique-local, ff multicast, 2002 6to4,::ffff: IPv4-mapped, 2001:db8 documentation, else global unicast.

Example. 2001:db8::1 / 64 → expanded 2001:0db8:0000:0000:0000:0000:0000:0001; compressed 2001:db8::1; network 2001:db8::/64; host bits = 64 (> 63) → host count shows "More than 2^{63}"; type = Documentation (2001:db8::/32).

Field notes

- Host counts above 2^{63} are reported qualitatively ("More than 2^{63}") rather than as a full number.
- The "first address" is the network address itself (IPv6 has no broadcast and does not reserve the all-zeros host as IPv4 does).
- Address-type detection is prefix-pattern based and covers the common RFC ranges, not every reserved block.

Lookup (ARP/NDP)

Discover local-network neighbors: IP, and MAC where the platform exposes it.

Why it's here. A neighbor read of the local segment. Reach for it to map IP↔MAC on platforms that can, or just to list responders.

How to use

1. Open and run; it derives the local subnet and sweeps it, attaching cached MACs where available.

Formula or method. Active discovery: derive the local /24 (or real prefix), TCP-connect-probe each host (curated LAN ports 80/443/22/445/139/53/8080, refused RST counts as up), list the responders, and on Linux/Android read /proc/net/arp to attach the real cached MAC. No subprocess (arp -a is sandbox-blocked). Safety cap refuses anything larger than a /22 (1022 hosts).

Field notes

- Platform matrix (honest, in the source): Android / Linux, active sweep with MAC from /proc/net/arp. macOS / Windows, active sweep, no MAC (no readable ARP file; arp -a/GetIpNetTable out of scope), so MAC renders "Not exposed on this platform". iOS, unavailable; neighbor tables aren't accessible to third-party apps. Web, download fallback.
- A null MAC means the platform doesn't expose it, never an invented value. Incomplete/all-zero ARP entries are skipped.
- On macOS/Windows you get a responder list without MACs; that's the platform ceiling. For richer enrichment and macOS MAC (via sysctl), see Network Discovery.

Lookup (BGP/ASN)

ASN, holder, announced prefix, registry, and peer/upstream counts for an IP or ASN.

Why it's here. Routing-layer context: who announces a prefix, which RIR, how it's connected. Reach for it on upstream/ISP questions.

How to use

1. Enter an IPv4/IPv6 or an ASN (AS15169, 15169, or as15169), look up.

Formula or method. Queries the RIPEstat Data API (free, no key, HTTPS, authoritative, operated by RIPE NCC). IP path: network-info (prefix + ASN) then as-overview (holder, type, registry, announced). ASN path: as-overview then asn-neighbours mapped to upstream/peer/downstream counts. Input is classified IP vs ASN before any call.

Field notes

- Platform differences: built on JsonHttpClient (→ dart:io), so native-only; web is gated to the download fallback (RIPEstat CORS unverified, so no maybe-broken web tool).
- Three states: success, empty (API answered cleanly but resolved no ASN, e.g. a private/bogon IP not in the routing table), and failure. Peer/upstream counts are best-effort enrichment; a neighbours failure leaves them null ("Not available"), not zero.
- Every field is nullable; a datum the API omits renders "Not available", never fabricated. Neighbour "type" mapping: left→upstream, right→downstream, unknown→peer.

Lookup (DNS)

A portable dig/nslookup for the field. Resolve a name as a dig-style all-records view (SOA, NS, A, AAAA, MX, TXT, SRV, CAA at once), as a single record type, or run a one-tap reverse PTR for an IP.

Why it's here. Checks records authoritatively rather than from the local cache, and works through captive portals and corporate firewalls. The all-records view is the fast "show me everything" pass; single-type is the focused look; reverse PTR turns an IP back into a hostname without flipping any selector.

How to use

1. Enter a hostname (or an IP). Leave the mode on All records for the dig-style sweep, then look up.
2. For one record type, switch to Single type and pick A, AAAA, MX, TXT, NS, SOA, PTR, SRV, CAA, or SPF.
3. When the input is an IP, a Reverse lookup (PTR) button appears for a one-tap IP to hostname query.
4. Pick the resolver (Cloudflare default or Google) if one provider is blocked.

Formula or method. Resolves over DNS-over-HTTPS (DoH) via basic_utils DnsUtils.lookupRecord, an HTTPS GET to a JSON resolver, not raw UDP/53. DoH is chosen because raw UDP/53 sits behind iOS local-network gating and is often blocked, while DoH rides HTTPS:443 cleanly and needs no extra socket capability. All records mode fans out one DoH query per type (SOA, NS, A, AAAA, MX, TXT, SRV, CAA) concurrently with Future.wait, then groups the answers in dig order; a per-type failure becomes a per-section note, so the records that did resolve still show. Reverse PTR rewrites the input IP to its in-

addr.arpa or ip6.arpa name before querying (with a minimal IPv6 literal parser). SPF is not a separate query: it reads TXT and filters for the v=spf1 policy line (RFC 7208). Three result states per query: success, empty (resolved, no records of this type), and failure, kept distinct so the UI shows the right state.

Field notes

- Records resolve against a public resolver (Cloudflare or Google), not the device configured resolver, which is usually what a pro wants (authoritative, uncached). The device configured DNS servers are shown by Interface Information instead.
- SRV and CAA are parsed into readable fields; an unparseable form is shown raw. Empty is not an error: a name with no MX simply returns the empty state.
- Platform differences: identical on iOS, Android, macOS, and Windows (HTTPS only). Gated off on web per the native-only product decision.
- dig +trace (the root to TLD to authoritative delegation walk) is not built. DoH only reaches recursive resolvers that return the final answer, so a true iterative trace would need raw UDP/53 to named authoritative servers plus a hand-rolled DNS wire codec, which is heavier and less reliable in the field than this tool is meant to be.

MAC Vendor OUI Lookup

Turn a MAC address into its registered vendor, fully offline.

Why it's here. Identify a device's manufacturer from its MAC with no network, and, crucially, tell you honestly when a MAC has no real vendor (randomized phones).

How to use

1. Enter a MAC in any common notation (colon, hyphen, Cisco dot aabb.ccdd.eeff, or no separators; any case), look up.

Formula or method. Pure-Dart resolver over a bundled IEEE registry table loaded once and cached, no HTTP, no dart:io. It honors the three IEEE block sizes and matches most-specific-first: MA-S (/36, 9 hex), then MA-M (/28, 7 hex), then MA-L (/24, 6 hex), so a /36 sub-allocation names the real sub-assignee, not the /24 parent.

Field notes

- Platform differences: identical everywhere (offline, pure Dart), including web-safe in principle, though it ships in the gated Networking category. No platform exposes more or less here.
- Honesty bits: U/L bit set (0x02) = locally-administered / randomized address (the common case for modern iOS/Android Wi-Fi) → flagged, no vendor invented (a registry hit would be coincidence). I/G bit set (0x01) = multicast/group address, not a single NIC → flagged, no vendor. A globally-administered MAC not in the bundled snapshot → the raw 24-bit OUI is shown (e.g. B8:27:EB), never invented.

- A randomized phone MAC correctly returns "no vendor"; that's the right answer, not a miss. The bundled table has a documented retrieval date in its asset header; refresh it to pick up new allocations.

Network Discovery

Find live hosts on the local network and enrich each with name, services, inferred device type, and (where exposed) MAC/vendor.

Why it's here. A Fing-style LAN scan. Reach for it for a richer inventory than a bare ping sweep: what each host is, not just that it answered.

How to use

1. Open and run. The engine seeds the local /24, connect-scans it, reverse-DNS resolves, mDNS-browses, then (macOS) reads the ARP cache for MAC/vendor.

Formula or method. Four passes: 1. Subnet seed: derive the local /24 host list from network_info_plus (:181-199). 2. Connect-scan: bounded-concurrency (64) TCP connect across the /24 × a curated port set, run in a background isolate; streams progress (:201-262). 3. Reverse DNS: InternetAddress.reverse() per discovered host (null when no PTR) (:264-275, 499-510). 4. mDNS browse: in-house native NetServiceBrowser/NetService (Apple Bonjour daemon) over a curated DNS-SD service set. Then the ARP-cache read (macOS only, via Swift sysctl NET_RT_FLAGS/RTF_LLINFO, never a subprocess) runs after the scan warms the kernel cache, attaching MAC + vendor (:303-335). Finally a pure device-type heuristic runs on each host's open ports + mDNS services. Device-type rules (first match wins, most specific first): IPP/LPD/9100 or printing mDNS → Printer; RTSP → Camera/NVR; iOS lockdownd (62078) → iOS device; SMB (445) → Windows/SMB; _sonos/_spotify-connect → Speaker; _googlecast → Media streamer; _airplay/_raop/_companion-link → Apple device; then weak signals 80/443/8080 → Web server, 22 → SSH host; any mDNS → mDNS device; else Unknown. MAC→vendor resolves through the full bundled IEEE OUI registry; the resolver owns the honesty contract: null for randomized/local MACs, raw-OUI fallback for unlisted global prefixes, named vendor otherwise.

Field notes

- Platform differences (the heart of it): MAC + vendor: macOS only (the sysctl ARP read). On iOS/Android a sandboxed app cannot read the ARP table, so MAC/vendor stay null. mDNS: iOS + macOS via the native NetServiceBrowser channel. It deliberately does NOT use pure-Dart multicast (iOS 14+ silently drops it without Apple's multicast entitlement) and does NOT use bonsoir (GPL-3.0, incompatible with the closed-source App Store app). Android/other platforms get a clean empty mDNS pass (NsdManager deferred). Service types must be declared in Info.plist NSBonjourServices. The connect-scan core is pure-Dart and cross-platform; only mDNS/ARP enrichment are native.
- Device type is a heuristic from ports + mDNS, not a MAC-anchored database; Unknown is a first-class, non-apologetic outcome. The code deliberately does NOT fake an "access point" rule, because the only reliable AP signal is the OUI vendor (MAC), which mobile can't read.

- Any single pass can fail without aborting the run (a failed mDNS browse just means no mDNS enrichment; nothing is faked). On mobile, expect no MAC/vendor and APs to fall through to SSH/Web/Unknown, a documented ceiling.

Nearby AP Scan (runs on Android today)

List the Wi-Fi access points a scan can see around you, each with SSID, BSSID, channel, band, and signal, with a per-band channel-occupancy bar.

Why it's here. A fast read of who else is on the air. When you are picking a channel or chasing co-channel interference, seeing the nearby BSSIDs across 2.4, 5, and 6 GHz tells you which channels are busy and which are clear, without carrying a separate analyzer.

How to use

1. Tap Scan to run a Wi-Fi scan. Each visible AP lists its SSID, BSSID, channel, band, and RSSI. Sort by signal or channel, and read the occupancy bars per band. Re-run to refresh.

Formula or method. Wired for Android today. The screen reads WifiManager.getScanResults() through the native com.wlanpros.toolbox/ap_scan method channel; ApScanService parses each result into a clean record (SSID, BSSID, RSSI, channel and band derived from the center frequency). Off Android the screen renders an honest per-platform state and never touches the channel: on iOS and macOS it says the OS blocks nearby-AP scanning; on Windows it says the scan is not wired into this tool yet (Windows Native Wifi CAN enumerate nearby APs, the FFI already fetches every BSS, so this is a not-yet-built path, not an OS block).

Field notes

- Runs on Android today. iOS and macOS block nearby-AP scanning at the OS level, so the tool is gated out of the catalog there. Windows is capable via its Native Wifi API (WlanGetNetworkBssList already fetches every visible BSS), but the Windows scan path is not wired into this tool yet: the copy says so honestly rather than implying only Apple restricts it.
- Clean fields only. The Android scan API exposes SSID, BSSID, channel, band, and RSSI for a scanned (non-connected) BSS. It does not expose a per-BSS noise floor, SNR, or MCS, so those columns do not exist here and are never shown.
- Android throttles Wi-Fi scans. When throttled, a rapid re-scan returns the last cached results and the screen notes it rather than faking a fresh scan. Location permission and Wi-Fi must both be on for the scan to return results.

Packet Sender

Send a custom TCP or UDP payload to host:port and read the reply (raw bytes + hex + decoded text).

Why it's here. A field "netcat" for poking a service: banner grabs, custom probes, protocol pokes. Reach for it to send a crafted payload and see what comes back.

How to use

1. Enter host, port, transport (TCP/UDP), and a payload (plain text, or \xNN hex escapes plus \r \n \t \0 \), send. The reply is shown as hex and decoded text.

Formula or method. TCP via Socket, UDP via RawDatagramSocket, no raw sockets, no custom ICMP/IP framing (same sandbox wall as ICMP traceroute), so it ships clean everywhere. TCP: connect → send → read until the peer closes or the read goes idle for the timeout, with a hard total cap (timeout×3) so a chatty stream can't run forever. UDP: bind ephemeral → send one datagram → wait up to the timeout for a reply; no reply is a first-class non-error outcome (timedOut/isNoReply), not an exception, because UDP has no delivery guarantee. Errors are typed: DNS failure, refused, unreachable, timeout, invalid input, other, each mapped to a precise message.

Field notes

- Platform differences: identical on every native platform. Gated off on web.
- A UDP "no reply" is honest and expected for many services; it does not mean failure. The payload parser returns null only on a malformed \x (a clear authoring mistake worth surfacing).
- Binary replies decode with the Unicode replacement char rather than throwing, so they still render. This is a single-shot send/receive, not an interactive session.

Ping (ICMP)

Real ICMP echo round-trip on mobile: live RTT, min/avg/max, loss.

Why it's here. When you specifically want true ICMP echo (the classic ping), available where the platform genuinely supports it.

How to use

1. Enter a host, run (mobile only). Streams replies with running stats, same UI shape as TCP Ping.

Formula or method. Real ICMP echo request/reply via the native backend (dart_ping_ios SimplePing/GBPing on iOS; dart_ping spawning system ping on Android). The method label is "ICMP echo", never relabeled from a TCP probe.

Field notes

- Platform matrix (honest): iOS, real ICMP echo available. Android, real ICMP echo via system ping, available. macOS / Windows / Linux desktop: the only ICMP path is spawning /sbin/ping, which the macOS App Sandbox blocks, so it gives an honest "not available in the sandboxed desktop build", and the UI points the user at TCP Ping instead. Web: no sockets → download fallback.
- Where available, this is genuine ICMP RTT/loss, the real thing, not a TCP proxy.
- DEVICE-PENDING: the code itself flags that the iOS real-ICMP backend "cannot be verified without a real device". The logic and gating are unit-tested with a fake backend; the live round-trip is the device-pending piece. On desktop, use TCP Ping.

Ping (TCP)

A reachability + round-trip-latency probe that works on every platform, including the sandboxed desktop, by timing a TCP handshake, not ICMP echo.

Why it's here. The portable ping. ICMP is often filtered while a TCP port (443) answers; this is the tcping/paping approach pros already use. It is also the desktop path where real ICMP can't run.

How to use

1. Enter a host, optionally pick a probe port (443 default; presets 443/80/53/22/7) and count, run. Live min/avg/max/loss and a sparkline build as replies land.

Formula or method. Each "ping" is a timed Socket.connect to host:port (default 443). A completed handshake OR an actively-refused RST both count as a successful round trip for latency (exactly how tcping treats it); only a genuine timeout or lookup failure is a loss. Probes are spaced by the requested interval minus the time the probe took, so cadence stays steady under latency.

Field notes

- Platform differences: identical everywhere native. Gated off on web. The screen labels the metric "TCP RTT" and shows the target port so it's never mistaken for ICMP.
- RTT is the TCP handshake time to a port, slightly higher than ICMP and dependent on the chosen port answering. A "refused" target still gives you a valid latency number (the host answered).
- This is not ICMP echo (see Ping (ICMP) for the mobile real-ICMP path). If a host filters the probe port, it reads as loss even if the host is up on another port; try a different probe port.

Ping Plotter

Runs a sustained ping to a target and charts round-trip latency over time, instead of the single-shot result the Ping and ICMP Ping tools give. The live trend, jitter, and visible dropped probes show how stable a path is, not just whether it answers once.

Why it's here. The live performance graph. A single ping says "reachable now"; a trend says "steady, spiky, or dropping packets." Reach for this to watch a flaky link over seconds or minutes, the view the single-shot ping tools can't give.

How to use

1. Enter a host, optionally pick a probe port (443 default; presets 443/80/53/22/7) and a sample interval (0.5s / 1s / 2s / 5s), then Start plot.
2. The chart fills left-to-right as replies land: a lime line for RTT and a red dot on the axis for any lost probe. The readout above shows current / min / avg / max / jitter and loss%.
3. It runs until you tap Stop; the chart keeps the most recent samples (a bounded window) so a long run stays fixed-size. Copy exports the summary plus a per-sample table.

Formula or method. Drives the shipped TCP-handshake Ping engine (PingService) in continuous mode (count = 0), one probe per chosen interval. Each reply folds into a bounded rolling window (default last 60 samples); min/avg/max are over the landed RTTs in that window, jitter is the mean absolute difference between consecutive landed RTTs (a lost probe breaks the chain, so jitter never pairs across a gap), and loss% is lost/sent. A timed-out / unreachable probe is recorded as an honest gap (no RTT) and drawn as a red axis dot, never as a fabricated 0 ms.

Field notes

- This is a TCP round-trip probe, not ICMP echo. The metric is labeled "TCP RTT" and the probe port is shown, so it is never mistaken for ICMP (see Ping (ICMP) for the mobile real-ICMP path).
- Platform: runs anywhere native, including the sandboxed macOS desktop where real ICMP can't (the ICMP path needs a subprocess the App Sandbox blocks). Gated off on web with the download-the-app prompt.
- Dropped probes are shown, never hidden: a lost sample is a red dot on the axis and counts toward loss%, so a flaky path reads honestly instead of as a smooth line.
- The chart retains a bounded window of recent samples (so memory stays flat on a long run); the copy export notes how many of the total samples are shown.

Ping Sweep

Discover responsive hosts on a subnet via a TCP-probe sweep (no ICMP).

Why it's here. A quick "who's on this segment" without raw sockets or a subprocess. Reach for it to enumerate live hosts on a /24.

How to use

1. Enter a CIDR (192.168.1.0/24), a range (192.168.1.10-40 or full end address), or a single IP; run. A live progress bar and a running responsive count build as hosts settle.

Formula or method. For each candidate, a TCP Socket.connect to a common port (443 default; the sweep can probe 443/80/22/53 and a host is responsive the moment ANY answers, first-answer wins). A completed handshake or a refused RST both prove the host answered; a timeout means silent on that port. Bounded worker pool (default 32 in flight). Hard cap of 254 hosts (a /24); anything larger is rejected with "that's N hosts, the cap is M", never silently truncated.

Field notes

- Platform differences: identical on every native platform. Gated off on web.
- A host is reported "responded", NOT "up"; a host silent on the probed ports may still be alive (ICMP-only or firewalled). The tool never claims ICMP-style liveness.
- This finds hosts that answer TCP on the probed ports. For richer host detail (name, services, type, vendor), use Network Discovery. CIDR /31 and /32 include every address; larger blocks exclude network and broadcast.

Port Scan

TCP connect scan of a host, either a common-ports preset or a custom range, reporting each port open/closed/filtered.

Why it's here. A privilege-free nmap -sT for the field. Reach for it to see what services a host exposes.

How to use

1. Enter a host, pick the common-ports preset or type a custom spec (e.g. 22, 80, 443, 8000-8100), run. Results stream in as ports settle.

Formula or method. Per port, Socket.connect(host, port) with an 800ms default timeout. Open = handshake completes; Closed = actively refused/reset (host reachable, nothing listening); Filtered = no response before timeout (a firewall dropping the SYN). Same open/closed/filtered taxonomy nmap reports for a connect scan, with no raw socket. Connects run in a bounded worker pool (default 64 in flight) and stream incrementally. The common-ports preset is 44 curated ports a network pro actually checks (20–27017, with service labels like 443→HTTPS, 3389→RDP).

Field notes

- Platform differences: works identically on every native platform (no entitlement beyond network-client). Gated off on web.
- "Filtered" means the SYN went unanswered (likely a firewall); it does not mean the port is closed. A custom range parser de-dupes and bounds-checks (1 to 65535).
- This is a TCP connect scan, not a SYN/stealth scan; it completes the handshake then tears it down. A host that's all-filtered is usually unreachable or firewalled wholesale.

Time Server (NTP)

Query an NTP server over SNTP and report the server's time, your device clock's offset from it, and the round-trip network delay.

Why it's here. When a log timestamp, a certificate window, or an 802.1X/Kerberos handshake looks wrong, the first question is whether the device clock is off. This answers it against a real time source and tells you which way and by how much, in plain language.

How to use

1. Open the tool. The server field defaults to time.apple.com; tap Run to query it.
2. If the default is blocked or slow, switch to the public fallback pool.ntp.org.
3. Read the stratum, the server time (UTC and local), the device time, the signed clock offset with its plain-language verdict, and the round-trip delay.

Formula or method. A single unicast SNTP exchange (RFC 4330 / RFC 5905): a 48-byte client request whose first byte is 0x1B (leap 0, version 3, mode 3) is sent over an outbound UDP datagram to port 123, and one reply is read. The four-timestamp math runs on integer microseconds: clock offset = ((t2

– t1) + (t3 – t4)) / 2 and round-trip delay = (t4 – t1) – (t3 – t2), where t1/t4 are the client transmit/receive instants and t2/t3 are the server receive/transmit timestamps read from reply bytes 32–39 and 40–47. NTP timestamps are 64-bit fixed-point seconds since 1900-01-01; the converter subtracts the 2208988800-second NTP-to-Unix epoch offset. A positive offset means the device clock is behind the server; the screen translates the sign so you never have to remember the convention.

Field notes

- Not a subprocess and not an HTTPS time API by design: an HTTPS endpoint can report the server's time but cannot measure the symmetric round-trip the SNTP four-timestamp formula needs to separate offset from delay. The outbound UDP/123 datagram is the unicast cousin of Wake-on-LAN's send, covered by the same network-client entitlement, and works the same on iOS / macOS / Android / Windows.
- Honest failures, never a fabricated offset: a timeout (3 s), a DNS failure, a reply shorter than 48 bytes, or a "kiss-o'-death" stratum 0 each surface as a clear message that suggests the fallback server, rather than a zeroed delay or a made-up offset.
- A tiny negative delay from clock jitter on a fast LAN is clamped to 0 so the reading is never nonsensical. Stratum is reported as-is (1–15 valid, 16 unsynchronized, 0 the kiss-o'-death marker).
- Web has no dart:io UDP socket, so the screen routes to the download-the-app fallback via NetworkSupport.ntpSupported.

Traceroute (Mobile)

Hop-by-hop path via an ICMP TTL-walk, Android only (iOS unsupported).

Why it's here. Extends traceroute to mobile where the platform genuinely supports it, built on the same shared ICMP layer as Ping (ICMP).

How to use

1. Android: enter a host, run; hops fill in via a TTL-walk (one ICMP echo per increasing TTL, surfacing the router that answers each).

Formula or method. A TTL-walk on the ICMP layer: send echoes with TTL 1..maxHops; an intermediate router answering TimeExceeded names that hop; the target answering EchoReply ends the walk.

Field notes

- Platform matrix (the critical honesty point): Android, available; dart_ping's TTL maps to ping -t (outbound TTL) and the system ping prints the responding hop on a "Time to live exceeded" line. iOS, not feasible, honestly unavailable. iOS can echo (via GBPing), but GBPing's receive path only accepts ICMP EchoReply (type 0); it never parses TimeExceeded (type 11), the message a traceroute needs to name each hop. Setting a low TTL just makes the echo time out with no hop IP. So iOS gets an honest "not on this device", never faked hops. Desktop: the system traceroute is the path; this ICMP TTL-walk reports sandboxed-desktop.
- Where available (Android), hops are the same TTL/IP/RTT/* * * shape as the system traceroute.

- The iOS limitation is a real platform ceiling in GBPing, documented at length in the source; it is the reason this tool is Android-only. DEVICE-PENDING: the Android path is device-pending verification per the source.

Traceroute (System)

Hop-by-hop path discovery via the OS traceroute/tracert (desktop).

Why it's here. The genuine traceroute, where it can actually run. Reach for it on a Mac/PC to see the routed path and where latency or loss enters.

How to use

1. Desktop only. Enter a host, run; hops fill in live as the OS tool emits them. Cancellable mid-flight.

Formula or method. Spawns the system traceroute (Unix: -m maxHops -q 3 -w 2) or tracert (Windows: -d -h maxHops -w 2000) and parses each hop line live from stdout/stderr: TTL, host/IP, per-probe RTTs, and * * * timeouts. A real traceroute needs to read ICMP TIME_EXCEEDED replies, which require either a raw socket or the privileged system binary, so faking hops from TCP timing is explicitly refused.

Field notes

- Platform matrix (honest): macOS / Windows / Linux desktop spawns the OS binary. But under the macOS App Sandbox (the App Store build) spawning is blocked, so the screen runs a live isLaunchable() probe (a side-effect-free no-arg launch) and adapts: a non-sandboxed Developer-ID macOS build and Windows/Linux launch it fine; the sandboxed build shows an explicit "binary unavailable" verdict rather than hanging or pretending. iOS / Android: subprocess execution is sandboxed out entirely → "Traceroute runs on desktop, use Ping here.". Web: never reached (gated).
- Each hop shows TTL, the responding router (name + IP), and up to three probe RTTs; * * * is a hop that didn't answer (common and not necessarily a problem). Reaching the target is reported as a terminal "complete".
- On a sandboxed macOS App Store build this tool honestly reports unavailable. For a path read on mobile, see Traceroute (Mobile), but note its iOS limitation.

Wake-on-LAN

Send a Wake-on-LAN magic packet to wake a host by MAC address.

Why it's here. Wake a sleeping machine on the LAN from your phone or laptop.

How to use

1. Enter the target MAC (colons, hyphens, Cisco dots, or no separators all parse), optionally a subnet-directed broadcast (e.g. 192.168.1.255) and port (9 default, 7 alternative), send.

Formula or method. Builds the 102-byte magic packet (6× 0xFF then the 6-byte MAC repeated 16 times) and sends it as a UDP broadcast via RawDatagramSocket with broadcastEnabled = true, no subprocess, no privileged socket. The MAC is normalized to canonical form; an invalid MAC, bad broadcast IP, or out-of-range port is rejected with a clear message.

Field notes

- Platform differences: works on every native platform (outbound UDP broadcast is covered by the existing entitlements). Gated off on web.
- Success means the packet was sent; it makes NO claim the device woke. WoL is unacknowledged, a switch may not forward the all-ones broadcast across subnets, and the target may have WoL disabled. The tool shows the bytes sent and the packet hex.
- If the OS reports 0 bytes sent, it suggests a directed broadcast (e.g. 192.168.1.255) instead of 255.255.255.255. "Sent" ≠ "woke"; verify the host separately.

WHOIS

Domain/IP registration lookup over WHOIS (TCP port 43), with parsed highlights and the raw record.

Why it's here. Registrar, dates, status, and name servers for a domain or IP, from the field. Reach for it on ownership/expiry questions.

How to use

1. Enter a domain or IP, look up.

Formula or method. Raw WHOIS over TCP/43 via Socket.connect, not a whois subprocess (sandbox-blocked) and not RDAP/HTTPS (uneven coverage, no CORS). It does the hierarchical two-hop dance: query whois.iana.org for the target, parse the refer:/whois: referral to the authoritative registry, re-query that, and follow one optional further hop to a Registrar WHOIS Server: if it returns a fuller record. Highlights (registrar, created/updated/expires, status, name servers) are parsed from the free-form record where reliably present; anything missing is omitted, never faked.

Field notes

- Platform differences: same on every native platform. Gated off on web.
- Three states: success (raw record + highlights), empty (server answered but the object is unregistered / a "No match" banner), and failure (connection/timeout/bad input). The servers consulted are listed so the path is transparent.
- WHOIS output is registry-specific and free-form; highlights are best-effort. The raw record is the source of truth.

Calculators & Tools (32 tools)

RF math and field utilities. Link-budget building blocks, antenna and coverage geometry, capacity and power figures, coordinate work, and unit conversions, each computed locally on the device.

RF & Propagation (9)

Cable Loss

Estimates total coax attenuation for a run of a known cable type, length, and frequency, plus the per-100 ft loss coefficient at that frequency.

Why it's here. Cable loss is a direct subtraction in the transmit and receive chains. Pick a cable type and length and the tool tells you how much signal the feedline will eat.

How to use

1. Pick the cable type from the list.
2. Enter frequency and pick its unit (GHz or MHz).
3. Enter run length and pick its unit (ft or m).
4. Read total loss in dB plus the per-100 ft coefficient.

Inputs

Input	Unit	Range
Cable type	one of: LMR-100A, LMR-200, LMR-400 (default), LMR-600, LMR-900, LMR-1200, RG-58, RG-8/U, RG-213, RG-214	-
Frequency	GHz (default) or MHz	must be > 0
Length	ft (default) or m	must be > 0

How it works. Each cable type carries a table of manufacturer-published loss figures (dB per 100 ft) at several reference frequencies. To find the loss at your frequency, the tool interpolates smoothly between the two nearest published points along a square-root-of-frequency curve, which matches how coax loss actually rises with frequency. Below the lowest published point it holds at that lowest value; above the highest point it extends the curve from the top two points. Total loss is then the per-100 ft figure scaled by your run length: total loss (dB) = per-100 ft loss × length in feet ÷ 100. Frequency in GHz is converted to MHz (× 1000) and length in meters to feet (× 3.28084) as needed. Results are rounded to two decimals. As an example anchor, LMR-400 at 2400 MHz is 3.9 dB per 100 ft.

Example. LMR-400, 2.4 GHz, 50 ft → per-100 ft at 2400 MHz = 3.9 dB; total = 3.9 × 50 / 100 = 1.95 dB.

Field notes

- Values are manufacturer-typical at room temperature; real loss rises with temperature and aging, and connectors add loss not modeled here.
- The square-root-of-frequency interpolation is a smooth fit between published spec points, not a measurement.
- Extrapolation above the highest published frequency (e.g. above 5800 MHz for LMR cables) is a model estimate. Treat 6 GHz results with caution.

Earth Curvature

Computes the earth bulge (the height of the earth's curvature at the midpoint of a path) for a given path length and atmospheric K-factor.

Why it's here. On long microwave links the earth's curvature, modified by atmospheric refraction, eats into Fresnel clearance. This sizes how much extra antenna height the curvature demands.

How to use

1. Enter path length and pick its unit (km or mi).
2. Pick the K-factor.
3. Read the bulge in meters and feet.

Inputs

Input	Unit	Range
Path length	km (default) or mi	mi × 1.60934 → km
K-factor	select	4/3 standard (1.333, default), 1.0 geometric, 2/3 worst-case (0.667), 2.0 superrefraction

How it works. The bulge is the height of the earth's curvature at the midpoint of the path, using a mean earth radius of 6371 km scaled by the atmospheric K-factor: bulge in meters = (path length in km)² × 1000 ÷ (8 × 6371 × K). The result is also shown in feet (meters × 3.28084).

Example. 20 km path, K = 1.333 → 20² × 1000 / (8 × 6371 × 1.333) = 5.89 m.

Field notes

- K = 4/3 is the standard-atmosphere assumption; K = 2/3 is the conservative worst case used for availability-critical links (smaller K means more bulge).
- The bulge is the maximum at midpoint; add it to required Fresnel clearance when sizing tower heights.

- Atmospheric K-factor varies with weather and geography. This is a planning value, not a guarantee.

Free Space Path Loss

Computes the loss in dB a signal suffers traveling through free space between transmitter and receiver, given the operating frequency and distance.

Why it's here. The starting point of any link budget. Reach for it to estimate how much signal a point-to-point or outdoor link loses before you factor in antennas and cables.

How to use

1. Enter the frequency and pick its unit (GHz or MHz).
2. Enter the distance and pick its unit (km, mi, or m).
3. Read the path loss in dB. The result blanks to a dash until both fields hold valid positive numbers.

Inputs

Input	Unit	Range
Frequency	GHz (default) or MHz	must be > 0
Distance	km (default), mi, or m	must be > 0

How it works. Free space path loss in dB = 20 × log10(frequency in GHz) + 20 × log10(distance in km) + 92.45. The constant 92.45 is the fixed offset for the km/GHz form of the equation. Inputs are converted to those units first (MHz ÷ 1000 → GHz; mi × 1.60934 → km; m ÷ 1000 → km). The output is rounded to one decimal, and blanks if frequency or distance is zero or negative.

Example. 5 GHz, 1 km → 20 × log10(5) + 20 × log10(1) + 92.45 = 106.4 dB.

Field notes

- Free space only. No obstructions, no ground reflection, no atmospheric effects. Real-world links always lose more, so use this as a floor, not a prediction.
- The reference card lists anchor values (2.4 GHz @ 1 km = 100.1 dB, 6 GHz @ 1 km = 108.0 dB).

Fresnel Zone

Computes the first Fresnel zone radius along a point-to-point path, plus the 60% clearance value planners treat as the "keep it clear" threshold.

Why it's here. Line of sight is not enough for a reliable link; the first Fresnel zone must be mostly clear of obstructions. This sizes how much clearance you need above terrain, rooftops, and trees.

How to use

1. Enter frequency in GHz.

2. Enter total path distance in meters.
3. Optionally enter a point-from-TX distance in meters to get the radius at that specific point; leave blank for the midpoint (maximum) only.
4. Read first-zone radius and 60% clearance (in meters, feet shown alongside).

Inputs

Input	Unit	Range
Frequency	GHz (fixed unit)	must be > 0
Total path distance	meters	must be > 0
Point from TX	meters, optional	must be inside (0, total) to add an at-point result, otherwise ignored

How it works. First the wavelength is found from the frequency: λ in meters = 0.3 ÷ frequency in GHz (the speed-of-light form, with c = 3 × 10^8 m/s). The first Fresnel zone radius at any point along the path is r = √(λ × d1 × d2 ÷ (d1 + d2)), where d1 and d2 are the distances from that point to each end in meters. The midpoint radius (the maximum, always shown) uses d1 = d2 = total distance ÷ 2. The 60% clearance value is r × 0.6, and feet are r × 3.28084.

Example. 5.8 GHz, 10000 m total, midpoint → λ = 0.3/5.8 = 0.05172 m; r = √(0.05172 × 5000 × 5000 / 10000) = 11.37 m; 60% clearance = 6.82 m.

Field notes

- The midpoint radius is the worst case along the path; clearance there is the binding constraint.
- Frequency unit is fixed to GHz here (unlike FSPL, which offers MHz).
- 60% clearance is the common rule of thumb; some designs require more in heavy-foliage or reflective environments.

ITU Rain Fade

Estimates rain attenuation on a microwave link using ITU-R P.838-3 (specific attenuation) and a simplified ITU-R P.530 (effective path length).

Why it's here. Above ~10 GHz, rain is the dominant fade mechanism on outdoor links. This sizes the fade margin a backhaul link needs to survive heavy rain.

How to use

1. Enter frequency in GHz.
2. Enter rain rate in mm/hr.
3. Enter path length and pick its unit (km or mi).
4. Pick polarization (Horizontal or Vertical).
5. Read total rain attenuation, specific attenuation γ, and effective path length.

Inputs

Input	Unit	Range
Frequency	GHz (fixed)	must be > 0
Rain rate	mm/hr	must be > 0
Path length	km (default) or mi	must be > 0
Polarization	Horizontal or Vertical	-

How it works. The two coefficients k and α come from the ITU-R P.838-3 published table (18 frequency points spanning 1–100 GHz, with separate horizontal and vertical values). For your exact frequency the tool interpolates k on a log-log basis and α on a log-linear basis, clamping at the table ends. Specific attenuation is then γ (dB/km) = k × (rain rate)^α. The effective path length shortens the geometric length to account for non-uniform rain cells: L_eff = L ÷ (1 + L/d0), where d0 = 35 × e^(−0.015 × rain rate). Total rain attenuation (dB) = γ × L_eff. Outputs are shown to two decimals for attenuation, four for γ, and two for L_eff.

Example. 23 GHz, 42 mm/hr, 5 km, Horizontal → interpolating between the 20 and 25 GHz nodes gives k ≈ 0.1027, α ≈ 1.085; γ = 0.1027 × 42^1.085 = 5.7195 dB/km; d0 = 35 × e^(−0.63) = 18.65, L_eff = 5/(1 + 5/18.65) = 3.94 km; attenuation = 5.7195 × 3.94 = 22.55 dB.

Field notes

- The model assumes the rain rate is uniform across the path; real cells are smaller, which the L_eff reduction partly accounts for.
- Pick a rain rate matching your target availability (e.g. the 0.01%-of-time rate for your region). The tool does not supply regional rain statistics.
- Below ~10 GHz rain fade is usually negligible.

Link Budget

Full point-to-point link budget: combines transmit power, antenna gains, cable losses, path loss, and miscellaneous losses into a received signal level, then compares against receiver sensitivity for a fade margin.

Why it's here. The decision tool for whether a link will close, and by how much. Assemble all the gains and losses to see the margin before you deploy.

How to use

1. Enter TX power and pick its unit (dBm, W, or mW).
2. Enter TX antenna gain, TX cable loss, path loss, RX cable loss, RX antenna gain, and RX sensitivity.
3. Optionally enter other/miscellaneous losses (treated as 0 when blank).
4. Read received signal (dBm) and link margin (dB), color-coded by health.

Inputs

Input	Unit	Range
TX power	dBm (default), W, or mW	W/mW must be > 0
TX gain	dBi	-
TX loss	dB	-
Path loss	dB	-
RX loss	dB	-
RX gain	dBi	-
RX sensitivity	dBm	-
Other losses	dB	optional, default 0

How it works. Power in watts is first converted to dBm (10 × log10(W × 1000); milliwatts convert via W = mW/1000). The received signal is the sum of gains minus losses along the chain: received (dBm) = TX power + TX gain − TX loss − path loss − RX loss + RX gain − misc losses. The link margin is received signal minus receiver sensitivity. The margin is color-coded: 10 dB or more is healthy (green), 0 to 10 dB is marginal (amber), and below 0 dB is negative (red).

Example. TX 30 dBm, TX gain 20, TX loss 2, path loss 120, RX loss 2, RX gain 20, misc 0, RX sensitivity −85 → received = 30 + 20 − 2 − 120 − 2 + 20 − 0 = −54 dBm; margin = −54 − (−85) = 31 dB (healthy).

Field notes

- Path loss is supplied by the engineer (compute it with the FSPL tool, then add real-world margins).
- This budget does not include rain fade; use the PtP Link Check for that.
- A healthy margin (≥ 10 dB) is the common design target to ride out fading and multipath.

Noise Floor

Computes the thermal noise floor (kTB) for a channel, the receiver noise floor (kTB plus noise figure), and the quick −174 dBm/Hz rule-of-thumb value.

Why it's here. SNR is signal minus noise floor. Knowing the noise floor for a given channel width sets the minimum usable signal and frames link-margin discussions.

How to use

1. Pick the channel bandwidth.
2. Enter receiver noise figure in dB.
3. Optionally adjust temperature (defaults to 20°C when blank).
4. Read thermal noise floor, receiver noise floor, and the rule-of-thumb value.

Inputs

Input	Unit	Range
Bandwidth	MHz	select of 20, 40, 80, 160, or 320 MHz
Noise figure	dB	must be ≥ 0 (default 7; invalid/negative blanks the outputs)
Temperature	°C	defaults to 20 when blank

How it works. The thermal noise floor is the classic kTB relationship: thermal (dBm) = 10 × log10(k × T × bandwidth in Hz) + 30, where k is Boltzmann's constant (1.380649×10^{-23} J/K), T is temperature in kelvin (°C + 273.15), and bandwidth converts from MHz to Hz ($\times 10^6$). The receiver noise floor adds the noise figure: rx floor = thermal + noise figure. The rule-of-thumb value uses the standard −174 dBm/Hz reference (kTB at roughly 0°C): rule = −174 + 10 × log10(bandwidth in Hz). Outputs are rounded to one decimal.

Example. 20 MHz, NF 7 dB, 20°C → thermal = 10 × log10(1.380649e-23 × 293.15 × 20e6) + 30 = −100.9 dBm; rx floor = −93.9 dBm; rule = −174 + 10 × log10(20e6) = −101.0 dBm.

Field notes

- This is the theoretical noise floor; real receivers see a higher effective floor from co-channel interference, adjacent-channel leakage, and ambient RF.
- The −174 rule is a 0°C approximation and differs slightly from the temperature-aware thermal value.

PtP Link Check

Full point-to-point backhaul link budget end to end, from TX power through antenna gains, cable losses, free-space path loss, and rain fade to the received signal, compared against sensitivity for a fade margin and a PASS / MARGINAL / FAIL verdict.

Why it's here. The all-in-one decision tool for a microwave or Wi-Fi backhaul link, combining FSPL and rain fade with the link budget so you get a single go/no-go answer.

How to use

1. Enter frequency (GHz), distance (km/mi), TX power (dBm), TX gain (dBi), RX gain (dBi), and RX sensitivity (dBm).
2. Optionally enter TX loss, RX loss, rain rate, required margin, and pick polarization.
3. Read EIRP, free-space loss, rain fade, received signal, link margin, and the verdict.

Inputs

Input	Unit	Range
Frequency	GHz (fixed)	must be > 0
Distance	km (default) or mi	must be > 0

Input	Unit	Range
TX power	dBm	signed
RX sensitivity	dBm	signed; sensitivity usually negative
TX gain	dBi	-
RX gain	dBi	-
TX loss	dB	default 0
RX loss	dB	default 0
Rain rate	mm/hr	default 0 (no fade)
Required margin	dB	default 10
Polarization	Horizontal or Vertical	-

How it works. This tool chains the link budget, free space path loss, and rain fade into one calculation. EIRP = TX power + TX gain − TX loss. Free space path loss = 20 × log10(distance in km) + 20 × log10(frequency in GHz) + 92.45. Rain fade uses the same ITU-R P.838-3 table and interpolation as the Rain Fade tool (γ = k × R^α, d0 = 35 × e^(−0.015 × R), L_eff = distance ÷ (1 + distance/d0), rain fade = γ × L_eff), returning zero when the rain rate is not greater than zero. The received level = EIRP − free space loss − rain fade + RX gain − RX loss, and the margin = received level − RX sensitivity. The verdict is PASS when margin is at or above the required margin, MARGINAL when the link still closes (margin 0 up to the required value), and FAIL when the margin is below zero. Outputs are shown to one decimal, except rain fade to two.

Example. 5.8 GHz, 10 km, TX 20 dBm, TX gain 23, RX gain 23, sensitivity −78, TX/RX losses 0, rain 0, required margin 10 → EIRP = 20 + 23 − 0 = 43.0 dBm; FSPL = 20 × log10(10) + 20 × log10(5.8) + 92.45 = 127.7 dB; rxLevel = 43 − 127.7 − 0 + 23 − 0 = −61.7 dBm; margin = −61.7 − (−78) = 16.3 dB → PASS. (With TX loss 1 and RX loss 1: EIRP 42.0, rxLevel −63.7, margin 14.3, still PASS.)

Field notes

- This is the most complete link tool in the suite. Prefer it over the bare Link Budget when the link is outdoors and above ~10 GHz.
- FSPL alone underestimates real loss; the rain fade leg only covers rain, not fog, foliage, or multipath.
- The MARGINAL band (link closes but below your required margin) is the tool's own warning state; the underlying pass/fail is purely margin ≥ required.

RF Attenuation

Estimates total path loss through building materials by summing per-layer attenuation for a chosen Wi-Fi band.

Why it's here. Indoor coverage planning. Count the walls, floors, and obstructions between AP and client and get a quick total dB loss to add to the link budget.

How to use

1. Pick the band (2.4, 5, or 6 GHz).
2. Enter a quantity for each material the signal passes through.
3. Read the total attenuation in dB.

Inputs

Input	Unit	Range
Band	select	2.4 GHz, 5 GHz, or 6 GHz
Per-material quantities	integers	only quantities > 0 contribute

How it works. Each material carries a fixed per-layer loss for each band (for example, drywall is 3/4/5 dB at 2.4/5/6 GHz; poured concrete 13/16/19; metal door 20/26/30; foil insulation 25/30/35). The total is simply the sum of each material's per-layer loss for the chosen band multiplied by how many of that material the signal passes through, added up across every material with a quantity greater than zero.

Example. 5 GHz, 2 sheets of drywall (4 dB each) + 1 concrete block (13 dB) → 2 × 4 + 1 × 13 = 21 dB.

Field notes

- The per-material values are typical attenuation figures, not measurements of your building. Real walls vary widely with construction, moisture, and reinforcement.
- Metal, foil/vapor barriers, and low-E glass are near-total blockers and dominate the total.
- Use this for a first-pass coverage estimate, then validate with a survey.

Antenna & Coverage (4)

Antenna Downtilt

Computes the mechanical downtilt angle that aims an antenna's beam center at a target coverage distance on the ground.

Why it's here. When mounting a sector or directional antenna at height, you need the tilt angle to put the main lobe where the users are.

How to use

1. Enter antenna height above ground (AGL) and pick its unit (ft or m).
2. Enter target coverage distance and pick its unit (km, ft, or m).
3. Read the downtilt angle in degrees.

Inputs

Input	Unit	Range
Antenna height (AGL)	ft or m	-
Target coverage distance	km, ft, or m	-

How it works. The downtilt angle is the arctangent of the antenna height divided by the coverage distance, converted to degrees: downtilt = atan(height ÷ distance). Both values are converted to meters first (feet × 0.3048, kilometers × 1000).

Example. 30 m height, 200 m coverage → atan(30/200) in degrees = 8.53°.

Field notes

- This aims the beam center at the target distance; it does not account for the antenna's vertical beamwidth (use Downtilt Coverage for the near/far edges).
- Pure geometry, no consideration of antenna pattern shape or ground slope.

Downtilt Coverage

Computes the near and far ground coverage edges (and coverage depth) of a downtilted antenna, given height, tilt angle, and vertical beamwidth.

Why it's here. The complement to the Downtilt tool. It shows the footprint the beam actually covers, so you can check for coverage gaps or overshoot.

How to use

1. Enter antenna height (AGL) and pick its unit (ft or m).
2. Enter the downtilt angle in degrees.
3. Enter the antenna's vertical beamwidth in degrees.
4. Read near edge, far edge, and coverage depth.

Inputs

Input	Unit	Range
Antenna height (AGL)	ft or m	must be > 0
Downtilt angle	degrees	-
Vertical beamwidth	degrees	must be in the open range (0, 180)

How it works. The beam's upper and lower edges are the tilt angle minus and plus half the vertical beamwidth. The near edge on the ground is the height divided by the tangent of the near (lower) angle; the far edge is the height divided by the tangent of the far (upper) angle. Coverage depth is the far edge

minus the near edge. If the upper beam edge reaches or clears the horizon (its angle drops to zero or below), the far edge is unbounded and reported as beam-above-horizon, with no far edge or depth. The result blanks out if height is not positive or beamwidth falls outside the (0, 180) range.

Example. 30 m height, 10° tilt, 8° beamwidth → far angle = 6°, near angle = 14°; near edge = 30/tan(14°) = 120.3 m; far edge = 30/tan(6°) = 285.4 m; depth = 165.1 m.

Field notes

- When the tilt is shallow relative to beamwidth (e.g. tilt 6°, beamwidth 15° → far angle = –1.5°), the upper edge clears the horizon and the far edge is unbounded. The tool flags this rather than returning a negative distance.
- Uses beam edges at the half-power (or stated) beamwidth; actual coverage tails off gradually beyond those edges.

EIRP Calculator

Computes Effective Isotropic Radiated Power, the actual power radiated from an antenna system after accounting for transmitter power, cable/connector loss, and antenna gain.

Why it's here. EIRP is the regulatory ceiling number. Check it against FCC/ETSI limits and use it as the transmit-side input to a link budget.

How to use

1. Enter TX power and pick its unit (dBm, W, or mW).
2. Enter cable loss in dB and antenna gain in dBi.
3. Read EIRP in dBm, plus a secondary line in watts/milliwatts.

Inputs

Input	Unit	Range
TX power	dBm (default), W, or mW	W/mW must be > 0 (a non-positive value makes the log non-finite and blanks the result)
Cable loss	dB	signed allowed
Antenna gain	dBi	signed allowed

How it works. Power entered in watts is converted to dBm as 10 × log10(watts × 1000); milliwatts convert the same way after dividing by 1000 to watts. EIRP in dBm is then simply transmit power in dBm minus cable loss plus antenna gain. To show the watt equivalent, EIRP in dBm converts back to power: 10^(EIRP_dBm ÷ 10), divided by 1000 for watts. EIRP is shown to one decimal; the watt line shows watts at two decimals when the value is at least 1 W, otherwise milliwatts at one decimal.

Example. TX 20 dBm, cable loss 1.5 dB, gain 14 dBi → 20 − 1.5 + 14 = 32.5 dBm → 10^(3.25)/1000 = 1.78 W.

Field notes

- Cable loss should include all connector and jumper losses on the transmit chain.
- The reference card lists regulatory EIRP ceilings (e.g. 2.4 GHz FCC PtMP = +36 dBm / 4 W); those are planning anchors and vary by sub-band, channel width, and power-control rules. Verify against current local regulations.

Wavelength

Converts a frequency to its wavelength in meters, centimeters, feet, and inches.

Why it's here. Antenna element sizing, ground-plane and spacing rules, and quarter-wave / half-wave rules of thumb all key off wavelength. A quick reference when laying out antennas.

How to use

1. Enter the frequency and pick its unit (MHz or GHz).
2. Read the wavelength in all four units at once.

Inputs

Input	Unit	Range
Frequency	MHz (default) or GHz	must be > 0

How it works. Wavelength in meters is 300 divided by the frequency in MHz: the 300 comes from the speed of light (3×10^8 m/s) scaled for the MHz-to-meter form. From there: centimeters = meters × 100, feet = meters × 3.28084, inches = feet × 12. A frequency entered in GHz is multiplied by 1000 to get MHz first. Display precision: meters to 4 decimals, centimeters to 2, feet to 4, inches to 3.

Example. 2400 MHz → 300/2400 = 0.1250 m, 12.50 cm, 0.4101 ft, 4.921 in.

Field notes

- Uses c = 3×10^8 m/s (the rounded 300 constant), not the exact 299792458. The tiny difference is irrelevant for antenna work.
- This is free-space wavelength; velocity factor inside cable or dielectric is not applied.

Capacity & Power (3)

Capacity Planner

Recommends the number of access points needed for a space, based on user count, concurrency, per-user demand, AP capacity, target utilization, and an optional clients-per-AP density cap.

Why it's here. High-density design. Size AP count by both throughput demand and client-density limits, taking the larger of the two.

How to use

1. Enter total users, concurrent usage %, per-user throughput (Mbps), AP max throughput (Mbps), and target channel utilization %.
2. Optionally enter max clients per AP for a density check.
3. Read concurrent users, total bandwidth demand, APs by throughput, APs by density, and the recommended AP count.

Inputs

Input	Unit	Range
Total users	count	must be > 0 (default hint 200)
Concurrent usage	%	must be > 0 (default hint 70)
Per-user throughput	Mbps	must be > 0 (default hint 5)
AP max throughput	Mbps	must be > 0 (default hint 600)
Target channel utilization	%	must be > 0 (default hint 50)
Max clients per AP	count	optional (default hint 50); ≤ 0 or blank disables the density check

How it works. First it finds concurrent users: total users times the concurrency percentage, rounded up. Total bandwidth demand is concurrent users times per-user throughput. Each AP's usable capacity is its max throughput times the target utilization percentage. APs-by-throughput is the total demand divided by that usable per-AP capacity, rounded up. APs-by-density is concurrent users divided by the max-clients-per-AP cap, rounded up (skipped when no cap is set). The recommended count is the larger of the two, and never less than one.

Example. 200 users, 70% concurrent, 5 Mbps/user, 600 Mbps/AP, 50% util, 50 clients/AP → concurrent = 140; total demand = 700 Mbps; usable per AP = 300 Mbps; APs by throughput = ceil(700/300) = 3; APs by density = ceil(140/50) = 3; recommended = 3.

Field notes

- A planning model, not a survey. It ignores RF coverage, building layout, and co-channel interference, which often drive AP count higher than capacity math alone.

- "AP max throughput" should be a realistic per-AP usable rate, not a marketing PHY peak.
- Concurrency and per-user demand are the biggest assumptions; size them from the actual application mix.

PoE Budget

Checks a PoE switch's total power budget against the sum of connected device power draws, with an over/caution/OK verdict.

Why it's here. Before connecting a batch of APs (and cameras, phones) to a switch, verify the switch can power them all without exceeding its PoE budget.

How to use

1. Enter the switch's PoE budget in watts.
2. For up to six device rows, enter watts-per-device and quantity (quantity defaults to 1).
3. Read total draw, remaining budget, percent used, and the verdict.

Inputs

Input	Unit	Range
Switch PoE budget	watts	must be > 0 (else no result)
Device rows (up to 6)	watts (decimal, blank → 0) and quantity (integer, default 1, blank → 0)	

How it works. Total draw is the sum, across all rows, of watts-per-device times quantity. Remaining budget is the switch budget minus total draw. Percent used is total draw over budget, capped at 100 for display. The verdict follows a simple order: if remaining is negative it reads OVER; otherwise if usage is above 80% it reads CAUTION; otherwise OK.

Example. Budget 370 W, six APs at 25.5 W each (qty 6) → total = 153 W; remaining = 217 W; pct = 41% → OK.

Field notes

- Use each device's worst-case (max) draw, not idle, and account for the PoE class.
- The pct is capped at 100 for display, but the OVER verdict triggers on actual negative remaining, so an over-budget case is never masked.
- This does not model per-port limits or cable-length power loss, only the switch's aggregate budget.

Throughput Calculator

Computes the PHY rate and an estimated real throughput for a Wi-Fi connection from standard, channel width, MCS index, spatial streams, and guard interval.

Why it's here. Translates a client's negotiated rate parameters into expected data-rate numbers, and sets realistic expectations for "what speed should I see."

How to use

1. Pick the Wi-Fi standard (802.11n/ac/ax/be).
2. Pick channel width, MCS index, spatial streams, and guard interval (the options reclamp to what's valid for the chosen standard).
3. Read modulation, PHY rate, and estimated real throughput in Mbps.

Inputs

Input	Unit	Range
Standard	select	HT (802.11n), VHT (802.11ac), HE (802.11ax, default), EHT (802.11be)
Channel width	MHz	HT {20,40}; VHT/HE {20,40,80,160}; EHT {20,40,80,160,320}
MCS index	index	max per standard: HT 7, VHT 9, HE 11, EHT 13
Spatial streams	count	up to: HT 4, VHT/HE/EHT 8
Guard interval	µs	HT/VHT {0.4, 0.8}; HE/EHT {0.8, 1.6, 3.2}

How it works. The PHY rate in Mbps is the number of data subcarriers times the bits carried per symbol times the number of spatial streams, all divided by the OFDM symbol duration in microseconds. The data-subcarrier count depends on the standard and channel width (for example, 802.11ax at 80 MHz uses 980 subcarriers). Bits per symbol comes from the MCS index: its modulation and coding rate (for example, MCS 11 carries about 8.3333 bits per symbol). The symbol duration depends on the standard and the chosen guard interval (for example, 802.11ax at a 0.8 µs guard interval is 13.6 µs). Estimated real throughput is the PHY rate times a flat per-standard efficiency factor: 0.70 for 802.11n, 0.72 for 802.11ac, 0.76 for 802.11ax, and 0.80 for 802.11be. Results round to one decimal, and an invalid width/guard-interval combination or an out-of-range MCS returns a blank.

Example. HE, 80 MHz, MCS 11, 2 streams, GI 0.8 → PHY = (980 · 8.3333 · 2) / 13.6 = 1201.0 Mbps; real = 1201.0 · 0.76 = 912.7 Mbps.

Field notes

- The efficiency factors (0.70 to 0.80) are flat per-standard estimates of MAC/overhead, not measured for your environment; real throughput depends on contention, retries, frame aggregation, and airtime sharing.
- PHY rate is the theoretical peak for a single, clean link. Treat the real-throughput number as an optimistic ceiling.

Coordinates & GPS (4)

Distance and Bearing

Computes the great-circle distance and the initial (forward) and reverse bearings between two latitude/longitude points.

Why it's here. Aiming point-to-point antennas. You need the distance (for FSPL) and the compass bearing (to point the dish) between two sites.

How to use

1. Enter latitude and longitude for point 1 and point 2 (decimal degrees).
2. Read distance (km), forward bearing, and reverse bearing.

Inputs

Input	Unit	Range
Point 1 / Point 2 latitude and longitude	decimal degrees	-

How it works. It treats the Earth as a sphere with a mean radius of 6371 km and uses the haversine method for distance. In plain terms: it measures the angular gap between the two points from their differences in latitude and longitude, then multiplies that angle by the Earth's radius to get the distance along the surface. The forward bearing is the compass heading you'd start out on to follow the shortest (great-circle) path from point 1 to point 2, and the reverse bearing is simply that heading plus 180°.

Example. San Francisco (37.7749, −122.4194) to Los Angeles (34.0522, −118.2437) → distance = 559.12 km; forward bearing = 136.5°; reverse bearing = 316.5°.

Field notes

- Spherical earth model (6371 km mean radius), accurate to a fraction of a percent for terrestrial links, not for survey-grade geodesy.
- The forward bearing is the initial bearing of the great-circle path; it changes along the route on long paths.
- The reverse bearing is the simple +180°, which is exact only on a sphere.

Final Point

Computes the destination latitude/longitude given a starting point, an initial bearing, and a distance (the "direct" geodesic problem).

Why it's here. Projecting where a link points. Given a site, a heading, and a distance, find the coordinates of the far end (e.g. to check a candidate tower location).

How to use

1. Enter the start latitude and longitude.
2. Enter the initial bearing in degrees.
3. Enter the distance and pick its unit (km, mi, or m).
4. Read the destination latitude and longitude.

Inputs

Input	Unit	Range
Start latitude / longitude	decimal degrees	-
Bearing	degrees	-
Distance	km, mi, or m	mi × 1.60934 → km; m ÷ 1000 → km

How it works. It treats the Earth as a sphere with a mean radius of 6371 km. First it converts your distance into an angle by dividing it by the Earth's radius. Then, starting from your point and heading off on the bearing you gave, it walks that angular distance along a great circle and reports where you land. The result longitude is wrapped to the −180° to +180° range so it always reads as a normal coordinate.

Example. From (40, −105), bearing 90° (due east), distance 100 km → destination (39.99, −103.83), about 1.17° of longitude east at that latitude (latitude dips marginally because a constant-90° initial bearing follows a great circle, not a parallel).

Field notes

- Direct great-circle solution on a sphere; the bearing is the initial heading and the path curves on long distances.
- Same spherical-model accuracy caveats as the other geographic tools.

Lat / Long Conversion

Converts a coordinate between decimal degrees (DD), degrees-decimal-minutes (DDM), and degrees-minutes-seconds (DMS).

Why it's here. Different tools and maps use different coordinate notations; this normalizes between them when entering a site location.

How to use

1. Enter a latitude value (and/or longitude) in decimal degrees.
2. Read the DD, DDM, and DMS forms with the correct hemisphere letter.

Inputs

Input	Unit	Range
Latitude	decimal degrees	valid range ±90

Input	Unit	Range
Longitude	decimal degrees	valid range ±180

How it works. The sign of the value sets the hemisphere: a latitude of zero or more reads N (otherwise S), and a longitude of zero or more reads E (otherwise W). To break a decimal degree into degrees, minutes, and seconds, it takes the whole-number part as the degrees, multiplies the leftover fraction by 60 to get the minutes, then multiplies that leftover fraction by 60 again to get the seconds. Degrees-decimal-minutes simply keeps the minutes and folds the seconds back in as sixtieths. The three forms are displayed to a fixed precision (DD to six decimals). Any value that is out of range or not a valid number blanks the output.

Example. 40.7128° latitude → DD 40.712800; DMS 40° 42' 46.08" N; DDM 40° 42.7680' N.

Field notes

- Validates ranges (±90 lat, ±180 lon) and blanks on out-of-range input.
- The sign drives the hemisphere letter, so enter west longitudes and south latitudes as negative.
- No datum/projection handling; these are plain WGS84-style decimal degrees in, formatted out.

Midpoint

Computes the great-circle midpoint between two latitude/longitude points.

Why it's here. Finding a relay or repeater site halfway along a long link, or the center point of a coverage area between two known locations.

How to use

1. Enter latitude and longitude for point 1 and point 2.
2. Read the midpoint latitude and longitude.

Inputs

Input	Unit	Range
Point 1 / Point 2 latitude and longitude	decimal degrees	-

How it works. Rather than averaging the two coordinates, it finds the point that sits halfway along the shortest (great-circle) path between them. It does this by combining the two positions as directions on the sphere and locating the heading that lands exactly between them, then converts that back to a latitude and longitude. The result longitude is wrapped to the −180° to +180° range so it always reads as a normal coordinate.

Example. Between (40, 0) and (40, 90) the great-circle midpoint lies well north of the latitude average, at (49.88, 45.00), reflecting that the shortest path bows poleward.

Field notes

- This is the great-circle (spherical) midpoint, not the average of the coordinates. On east-west paths the midpoint is noticeably closer to the pole than the simple lat/lon average.
- Spherical model, same accuracy caveats as Distance and Bearing.

Conversions (5)

Channel / Frequency

Convert a Wi-Fi channel number to its center frequency and back, across 2.4 / 5 / 6 GHz, with the bonded-channel component lists.

Why it's here. Specs, radios, and regulators talk in both channels and megahertz. This converts either direction and shows which 20 MHz channels make up a wide channel, so a "channel 100, 80 MHz wide" entry on a controller stops being a guess.

How to use

1. Pick a band (band is required for channel→frequency because channel numbers collide across 5 and 6 GHz, channel 36 exists in both).
2. Enter a channel to get the center frequency, or a frequency to get the channel and the band it falls in (the frequency ranges are disjoint, so the band comes back for free).
3. Read the bonded-channel breakdown for 40/80/160/320 MHz widths.

How it works. One linear rule covers almost every channel: center frequency in MHz = the band's starting frequency + 5 × the channel number, where the starting frequency is 2407 (2.4 GHz), 5000 (5 GHz), or 5950 (6 GHz). Two special cases bypass it: 2.4 GHz channel 14 = 2484 MHz, and 6 GHz channel 2 = 5935 MHz. The 6 GHz start of 5950 is the verified value (channel 1 = 5955), not the 5940+5n some bad sources use. Channel↔frequency is universal physics, identical everywhere; channel availability (US 1–11 vs world 1–13, DFS, UNII-4, 6 GHz adoption) is regulatory and is surfaced as a caveat, never as a reason to change a computed frequency.

Field notes

- The selectable 20 MHz primaries are the exact allowed sets, not a naive arithmetic range: 2.4 GHz is 1–14; 5 GHz is the verified UNII-1/2A/2C/3/4 list (anything like 37, 49, 51, 145, 181 is rejected); 6 GHz is the sequence 1, 5, 9 … 233 plus the special channel 2.
- DFS channels, UNII-4, and the 6 GHz Preferred Scanning Channels are flagged as caveats, not enforced. Channel 2 (6 GHz) is flagged "Special (5935 MHz, reserved/guard)" and channel 14 (2.4 GHz) "Special (2484 MHz)".
- Channel data is drawn only from primary sources: IEEE 802.11-2020/2024 channelization plus the FCC 6 GHz / 5.9 GHz Reports & Orders. No channel data comes from a secondary source.

dBm / Watt Converter

Live two-way conversion between dBm, Watts, and milliwatts. Type in any one field and the other two update in real time.

Why it's here. RF power lands in your lap in three units depending on the spec sheet, the regulator, or the radio. This converts between them so you don't reach for a calculator at the AP.

How to use

1. Type a value in any of the three fields (dBm, Watts, or Milliwatts); the other two recompute instantly.
2. dBm and Watts accept scientific notation (e.g. 1e-10) plus a sign; Milliwatts is unsigned decimal.
3. Watts renders in scientific notation (5 sig figs) because real Wi-Fi receive levels are tiny (e.g. 1e-10 W); mW shows 4 fixed decimals.
4. Entering Watts or mW that are zero or negative shows a dash in the dBm field, since the logarithm of a non-positive number is undefined.

Inputs

Input	Unit	Range
dBm	dBm (decibel-milliwatts)	any real number; signed + scientific notation accepted
Watts	W	> 0 to convert to dBm; ≤ 0 yields a dash; signed + scientific notation accepted
Milliwatts	mW	> 0 to convert to dBm; ≤ 0 yields a dash; unsigned decimal

How it works. Three conversions do the work: dBm→W = 10^(dBm/10) ÷ 1000; dBm→mW = 10^(dBm/10); and W→dBm = 10 × log10(W × 1000). Converting mW→dBm just divides mW by 1000 to get Watts and reuses the W→dBm math. The factor of 1000 is the Watt-to-milliwatt relationship (1 W = 1000 mW). In plain form, dBm = 10 × log10(mW) and W = 10^(dBm/10) ÷ 1000. Watts display in scientific notation to 5 significant figures; mW shows 4 decimals and dBm shows 2. A zero or negative Watts or mW yields a dash in the dBm field because the logarithm is undefined there.

Example. 0 dBm → 10^(0/10) = 1 mW = 1e-3 W. +30 dBm → 10^(30/10) = 1000 mW = 1 W. 20 mW → 10 · log10(20) = 13.01 dBm.

Field notes

- 0 dBm is exactly 1 mW by definition, the reference point of the scale.
- The on-screen reference card anchors common values: +30 dBm = 1000 mW (1 W, FCC 2.4 GHz max conducted), +13 dBm = 20 mW (common default AP Tx power), 0 dBm = 1 mW, -67 dBm = 0.2 nW (minimum for enterprise data), -80 dBm = 10 pW (typical ambient noise floor).
- Watts uses scientific notation deliberately: Wi-Fi receive power sits around 1e-10 W, which fixed notation renders as an unreadable string of zeros.

Hex / ASCII

A live decimal/hexadecimal/binary integer converter plus a printable-ASCII reference table (codes 32 to 126).

Why it's here. Reading packet captures, MAC/OUI bytes, and config hex values. Flip a number between bases or look up an ASCII character without leaving the app.

How to use

1. Type a value into any of the three converter fields (decimal, hexadecimal, binary); the other two update live.
2. Scroll the ASCII table to look up a printable character's decimal/hex/binary code.

Inputs

Input	Unit	Range
Decimal	digits only	-
Hexadecimal	0-9, a-f, A-F (optional 0x prefix stripped)	-
Binary	0/1 (optional 0b prefix stripped)	Unsigned-integer domain, backed by arbitrary-precision integers so arbitrarily long strings never overflow; a blank/invalid field blanks the mirrors

How it works. Each value is parsed as an arbitrary-precision unsigned integer and re-expressed in base 16, base 2, and base 10. The ASCII table is generated from the character codes rather than hand-transcribed: hex is shown as 2-digit uppercase, binary as 8-bit zero-padded.

Example. Enter decimal 65 → hex 41, binary 1000001. In the table, decimal 65 = char A = hex 41 = binary 01000001.

Field notes

- Unsigned integers only: no negative numbers, no fractions, no two's-complement.
- The table covers printable ASCII (32 to 126) only; control characters and extended/Unicode code points are out of scope.
- The converter is base conversion, not text encoding (it converts a single integer, not an ASCII string to its byte sequence).

Metric Conversion

Converts a length between meters, kilometers, miles, feet, centimeters, inches, and nautical miles.

Why it's here. Field work mixes metric and imperial constantly (datasheets in meters, tape measures in feet, link distances in miles). A quick unit converter avoids arithmetic slips.

How to use

1. Enter a value and pick its "from" unit.
2. Read the value in every other unit (the tool pivots through meters).

Inputs

Input	Unit	Range
Value and source unit	one of m, km, mi, ft, cm, in, nmi	-

How it works. Every unit is defined by its length in meters: m = 1, km = 1000, mi = 1609.344, ft = 0.3048, cm = 0.01, in = 0.0254, nmi = 1852. To convert, multiply the value by the "from" unit's meters, then divide by the "to" unit's meters. Display decimals vary by unit: m 4, km 6, mi 6, ft 4, cm 2, in 4, nmi 6.

Example. 1 mi → 1609.344 m → 1.609344 km, 5280.0 ft, 0.868976 nmi.

Field notes

- Uses the international mile (1609.344 m) and international foot (0.3048 m), not the US survey foot.
- Conversions are exact within floating point; the per-unit decimal rounding is for display only.

Unit Converter

Converts a value between units in one of eight categories: data transfer rate, data storage, length, power, metric prefix, speed, temperature, and time.

Why it's here. Field work mixes units constantly. A datasheet quotes throughput in Mbps, your test rig reports MB/s, an RF reading lands in dBm when you wanted milliwatts. Pick a category, pick the two units, read the answer.

How to use

1. Pick a category (data transfer rate, data storage, length, power, metric prefix, speed, temperature, or time).
2. Type a value, then pick the "from" unit and the "to" unit.
3. Read the converted result. It blanks until the value parses to a valid number.

Inputs

Input	Unit	Range
Category	one of eight categories	-
Value, from-unit, to-unit	the units offered in the chosen category	signed and scientific input accepted (a temperature can be negative)

How it works. Linear categories convert through a single base unit: multiply the value by the "from" unit's factor, then divide by the "to" unit's factor. Power and temperature are not linear and use their own math. Power treats dBm with the same log math as the dBm/Watt converter (W = 10^(dBm/10) ÷ 1000; dBm = 10 × log10(W × 1000)); temperature converts through Kelvin using its affine (offset-and-scale) relationships.

Example. 100 MB → 800 Mbit (storage, decimal). 100 °C → 212 °F (temperature). 1 W → 30 dBm (power).

Field notes

- Decimal and binary are kept separate. A KB is 1000 bytes; a KiB is 1024 bytes. The tool never conflates the two, so a storage answer reads true to the standard you picked.
- Bits and bytes are different units, not a display toggle. 1 byte = 8 bits, and the converter treats them that way across both the storage and the rate categories.
- Temperature and dBm are not simple multiply-by-a-factor conversions. They are handled with the correct affine and logarithmic math, so a negative temperature or a sub-milliwatt power reads correctly.

Utilities & Generators (4)

DTMF Generator

Plays the Touch-Tone keypad tones (0-9, *, #, and A-D) from a standard 4×4 DTMF grid, one tap at a time or as a continuous loop.

Why it's here. Driving a system that still listens for Touch-Tones over the air or down a line: an IVR menu, a repeater controller, a legacy PBX, or a piece of test gear. Hold the device near the microphone and play the digit.

How to use

1. Tap a key to play its tone for a short burst. The selected key and its two frequencies show at the top.
2. Tap Play to loop the selected key's tone continuously; tap Stop to end it.
3. Tapping a different key during a loop retargets the loop to the new key.

Inputs

Input	Unit	Range
Keypad key	one of 1-9, 0, *, #, A-D	defaults to the center key, 5

How it works. Each DTMF key is the sum of two sine waves, one low-group frequency (the row) and one high-group frequency (the column), per ITU-T Q.23. Low group: 697, 770, 852, 941 Hz. High group: 1209, 1336, 1477, 1633 Hz. A single key tap plays the pair for about 200 ms. The tone is synthesized on the device; nothing is dialed or transmitted by the app itself.

Example. Key 5 is 770 Hz + 1336 Hz. Key 1 is 697 Hz + 1209 Hz. The * key is 941 Hz + 1209 Hz.

Field notes

- Every key is two tones added together, which is what "Dual-Tone Multi-Frequency" means. That is why the readout shows two frequencies, not one.
- The A, B, C, and D keys are real DTMF tones that never appeared on consumer phones. They show up in radio, military, and control systems, which is why the full 4×4 grid is here.
- Tones come out of the device speaker. The app generates audio; it does not place a call or send anything down a phone line. To control remote gear, play the tone into that system's microphone or audio input.
- Reliable detection depends on volume, the speaker, and the receiving system. Hold the device close and keep the level up if a stubborn IVR or controller misses a digit.

Blue Box (MF)

Plays the R1 Multi-Frequency (MF) trunk-routing tones a 20th-century long-distance switch once used to route a call, plus the 2600 Hz supervisory tone that signaled an idle trunk. A mode of the DTMF Generator.

Why it's here. Telephone signaling history and nostalgia. A blue box reproduced the in-band MF signaling AT&T's switches used internally. This mode lets you hear those tones and see the frequency pairs behind them. It is an educational exhibit that shows off the same tone synthesizer the DTMF tool uses.

How to use

1. Switch to the Blue Box mode with the selector at the top of the DTMF Generator.
2. Tap a digit (1-0), KP, ST, or 2600 to hear that signal.
3. The readout shows the signal's frequency pair and burst timing.

Inputs

Input	Unit	Range
Signal	MF digit 1-0, KP, ST, or 2600 Hz	tap to play; no input required

How it works. Each MF signal is the sum of two sine waves drawn from six frequencies: 700, 900, 1100, 1300, 1500, 1700 Hz. The digit pairs use the canonical ITU-T assignment (for example digit 1 = 700 + 900 Hz, digit 2 = 700 + 1100 Hz). KP (Key Pulse, start of digits) = 1100 + 1700 Hz; ST (Start, end of digits) = 1500 + 1700 Hz; the 2600 Hz supervisory tone is a single frequency. The tone is synthesized on the device at an 8000 Hz sample rate; nothing is dialed or transmitted.

Example. Digit 5 is 900 + 1300 Hz. KP is 1100 + 1700 Hz. The seize tone is a single 2600 Hz tone.

Field notes

- These tones do nothing on any modern phone network. The MF and 2600 Hz signaling a blue box exploited lived in the voice band, where the switch listened for it. Carriers moved call control out of the voice band onto a separate data channel (CCIS, then SS7, still in use) in the 1980s-1990s, so there is nothing in the audio path to seize or route. This is reproduced for history and education only.
- The digit-to-frequency-pair mapping here follows the canonical ITU-T assignment by frequency membership. Some hobbyist references order the pairs differently; both describe the same six-frequency set.
- Tones come out of the device speaker. The app generates audio; it does not place a call or send anything down a line.

Red Box (US coin tones)

Plays the US ACTS payphone coin-deposit acknowledgement bursts for a nickel, dime, and quarter. All three are a 1700 + 2200 Hz dual tone; only the burst count and timing differ. A mode of the DTMF Generator.

Why it's here. Telephone signaling history and nostalgia. A red box reproduced the tones an ACTS payphone sent up the line to report that coins had been deposited. This mode lets you hear those coin patterns and see the timing behind them, using the same tone synthesizer the DTMF tool uses.

How to use

1. Switch to the Red Box mode with the selector at the top of the DTMF Generator.
2. Tap Nickel, Dime, or Quarter to hear that coin's burst pattern.
3. The readout shows the 1700 + 2200 Hz pair and the burst timing for that coin.

Inputs

Input	Unit	Range
Coin	Nickel, Dime, or Quarter	tap to play; no input required

How it works. Every US red box tone is the 1700 + 2200 Hz dual tone, in coin-specific burst patterns: nickel = one 66 ms burst; dime = two 66 ms bursts, 66 ms apart; quarter = five 33 ms bursts, 33 ms apart. The tone is synthesized on the device at an 8000 Hz sample rate (the 2200 Hz top component is well under the 4000 Hz Nyquist limit).

Example. A quarter is five 33 ms bursts of 1700 + 2200 Hz, each 33 ms apart.

Field notes

- These tones do nothing on any modern phone network. Red boxing worked only on Automated Coin Toll System (ACTS) payphones that listened for these tones in the voice path. ACTS was phased out across the US, many payphones added acoustic filters, and payphones themselves are nearly extinct. This is reproduced for history and education only.

- This mode covers the well-documented US nickel / dime / quarter set only. The frequencies and burst timings are the corroborated standard 1700 + 2200 Hz values.
- Tones come out of the device speaker. The app generates audio; it does not place a call or send anything down a line.

QR Code Generator

Turns any text or URL you type into a scannable QR code, then lets you share or save it as an image.

Why it's here. Handing off a config URL, a guest Wi-Fi link, or a site address without making someone type it. Generate the code on the spot and let them scan it with a phone camera.

How to use

1. Type the text or URL you want to encode, or switch to Wi-Fi mode to build a "scan to join" network code.
2. The QR code renders live as you type.
3. Tap Share / Save to send the image through the system share sheet or save it.

Inputs

Input	Unit	Range
Text or URL	any string	empty input shows a prompt and renders no code
Wi-Fi (scan-to-join)	SSID, password, auth (WPA/WEP/open), hidden flag	builds the standard WIFI: payload iOS Camera and Android scanners honor as a join offer

How it works. The code is generated on the device with no network call. It renders as dark modules on a white background with a 4-module quiet zone, which is what scanners expect. Share / Save captures that white tile to a PNG image.

Example. Type https://wlanpros.com and the matching QR code appears below the field, ready to scan or share.

Field notes

- Dark code on a white background, always. Inverted light-on-dark codes look on-brand but many scanners fail to read them, so the tool does not offer that option.
- The white border around the code is the quiet zone, and it is part of the code. Do not crop it out when you share the image, or the QR may not scan.
- Everything happens on the device. The text you encode never leaves the app except through the share sheet you trigger.
- Encodes plain text and URLs. It is a generator, not a scanner, so it makes codes rather than reading them.
- Wi-Fi scan-to-join mode builds the de-facto-standard WIFI:T:;S:;P:;H:<true|false>;; payload that Apple and Google honor. Special characters in the SSID or password are backslash-escaped, and

hex-looking or space-padded values are double-quoted, so a code that scans on your phone scans on the guest's too. Open networks use nopass and omit the password field.

Ham Radio (2)

The pure-math amateur-radio tools. Channel/frequency physics that a Wi-Fi pro already trusts, applied to antenna dimensions and grid-square locators. The band-dependent amateur references live under Quick Reference → Ham Radio.

Antenna Length

Compute the half-wave dipole and quarter-wave vertical physical lengths from a frequency (or a wavelength), velocity-factor adjusted, shown beside the classic 468/f and 234/f rules of thumb.

Why it's here. Cutting an antenna is the one piece of RF where a wrong number means a saw cut you cannot undo. This gives the physics length and the field rule-of-thumb side by side, so you see the spread before you cut.

How to use

1. Enter a frequency in MHz (or switch to wavelength→frequency for the inverse).
2. Set the velocity factor; it defaults to 0.95 (thin bare wire).
3. Read the half-wave dipole and quarter-wave vertical lengths in meters, feet, and inches, plus the 468/234 rule-of-thumb feet.

How it works. It uses the exact speed of light, c = 299.792458 in MHz·m form, not the rounded 300. Wavelength in meters = 299.792458 ÷ frequency in MHz, and the inverse frequency in MHz = 299.792458 ÷ wavelength in meters. Physical length = electrical length × velocity factor: the half-wave dipole is (wavelength ÷ 2) × velocity factor, and the quarter-wave vertical is (wavelength ÷ 4) × velocity factor. The rules of thumb are the classic feet forms: length in feet ≈ 468 ÷ frequency in MHz for the dipole, and length in feet ≈ 234 ÷ frequency in MHz for the quarter-wave vertical. Those figures already fold in a typical end-effect/velocity factor, which is why they read a touch shorter than the bare physics half/quarter wavelength. Both are shown so you see the spread.

Field notes

- Velocity factor is the knob that matters: the 468/234 rules assume a typical wire, while the physics length × your chosen velocity factor is the honest computed value. Insulated wire, traps, and tubing all shift velocity factor, so confirm it for the element in your hand.
- Worked sanity checks: 14.2 MHz → wavelength ≈ 21.11 m, half-wave dipole ≈ 33 ft (velocity factor 0.95); 146 MHz → quarter-wave vertical ≈ 19–20 in; 2400 MHz → quarter-wave vertical ≈ 3 cm.
- Pure math, no network or platform call; the Copy action exports the full computed report as labeled text.

Maidenhead Grid Square

Encode a latitude/longitude to a Maidenhead (QTH) locator and back at 4, 6, or 8 characters, plus the great-circle distance and bearing between two grid squares.

Why it's here. Hams trade positions as grid squares, not lat/long. This crosses between the two and gives the distance-and-bearing leg, the same spherical math the Distance & Bearing tool uses, so cross-tool results agree.

How to use

1. Enter a lat/lon to encode (choose 4, 6, or 8 characters), or a locator to decode to its cell.
2. Decode returns the cell's south-west corner, its width/height in degrees, and the center point (what a locator→position lookup returns).
3. Enter two locators to get the great-circle distance (km and miles) and the initial bearing between their centers.

How it works. It follows the IARU locator standard: normalize longitude to 0–360 and latitude to 0–180, then subdivide each axis level by level, most-significant pair first, with per-level divisions of 18 / 10 / 24 / 10. Each pair writes the longitude character first, then the latitude character; the field is A–R, the square 0–9, the subsquare a–x, the extended square 0–9, and decoding is case-insensitive. Encoding clamps exactly +180 longitude and +90 latitude a hair inside the range so the top edge does not index past the last cell. The great-circle leg is the spherical haversine with an Earth radius of 6371 km, plus the forward-bearing formula; miles = km × 0.621371.

Field notes

- Output precision is in characters: 4 = field + square, 6 = + subsquare, 8 = + extended square. A 6-character locator is the common amateur exchange.
- Verified anchors: lon −122.0, lat 37.4 → CM87 (the SF Bay anchor); Berlin 13.4 E, 52.5 N → JO62.
- Pure math, no I/O or platform API, so it works identically on every platform offline.

Learn / RF intuition (1)

Interactive teaching tools that build RF intuition you can sense, not just compute.

Hear the Frequency

Turn a frequency into a sounding tone so the logarithmic instinct behind RF (pitch, octaves, the twelve piano keys, and harmonics) becomes something you hear, with the honest bridge to RF stated plainly.

Why it's here. "An octave is a doubling" and "a dB is a logarithm" are the same instinct, and hearing it lands harder than reading it. This makes frequency audible to build that intuition, then states the limit of the analogy out loud so you do not carry a wrong equivalence into RF work.

How to use

1. Tap Play to sound the current frequency. Use the octave ×2 / ÷2 buttons, the one-octave C4→C5 keyboard, or the octave-ladder presets; the sounding voice retunes live with no click.
2. Pick a waveform (Triangle is the default; sine, square, and saw are available).
3. Read the live nearest-note, cents offset, octaves-from-A4, and the first five harmonics.

How it works. The tool uses pure equal-temperament math: every note frequency is computed, never transcribed. Each note's frequency equals 440 Hz times 2 raised to the power of (n – 49) divided by 12, where n is the piano key number and key 49 = A4 = 440 Hz (ISO 16). The semitone ratio is 2 raised to the 1/12 power (about 1.0594630943592953), an octave is exactly a doubling (octave up multiplies by 2, octave down divides by 2), and 12 semitones make an octave. The nearest-note readout finds the note by taking 69 plus 12 times the base-2 logarithm of the frequency divided by 440, then reports the cents offset. Harmonics are simple integer multiples (the first is the frequency itself, the second is twice it, and so on). When you change octave, key, or preset, the single sounding voice retunes to the new frequency so it glides smoothly without a stop and restart.

Field notes

- The analogy honesty is baked into the copy: an octave is a base-2 FREQUENCY ratio while a dB is a base-10 POWER ratio: both are logarithmic ratio measures (the real, teachable bridge), but they are NOT the same unit, and the tool never converts an octave to a dB. Audio harmonics are WANTED (timbre); RF harmonics are usually UNWANTED (spurious emissions). The integer-multiple math is identical; the desirability flips.
- Playback is clamped to the audible range 20 Hz–20 kHz; a number above it may still be DISPLAYED for the RF analogy, clearly labeled, but is never driven to the speaker. Above ~5 kHz the waveform is forced to sine because square and triangle harmonics alias near the Nyquist limit.
- Honest no-audio state: if audio cannot initialize (no output device, a muted session), the screen shows a non-fatal "No audio output detected" message instead of pretending a tone played. The tone auto-stops on screen exit so it never runs unattended.

Quick Reference (89 tools)

Offline lookup tables and the laminated field cards. Channel plans, standards, thresholds, connector and cabling pinouts, protocol references, CLI and capture cheat sheets, checklists, and guides, all available without a connection.

Wi-Fi & RF (23)

802.11 Standards

A PHY-layer comparison of every major 802.11 amendment from the original 802.11 (1997) through Wi-Fi 7, with year, bands, max PHY rate, MIMO, channel widths, and modulation.

Why it's here. Settling "which generation does what": bands reached, max rate, MIMO ceiling, and modulation per amendment.

How to use

1. Scan by generation badge (Wi-Fi 4 through Wi-Fi 7).
2. The band filter answers "which generations reach 6 GHz" (Wi-Fi 6E and Wi-Fi 7).
3. The original 802.11 shows a dash for generation and MIMO (it predates both).

Field notes

- What it shows: one card per amendment with the IEEE designation, a Wi-Fi generation badge, year, and rows for Bands (GHz), Max PHY rate, MIMO, Channel width (MHz), and Modulation. An optional band filter (All / 2.4 / 5 / 6 GHz) narrows the list.
- Two footnotes: (1) "Wi-Fi 1/2/3 are informal/retroactive labels; official Wi-Fi Alliance naming begins at Wi-Fi 4"; (2) "Wi-Fi 7 certification began 2024; IEEE 802.11be was published 2025."
- Max PHY rate is the theoretical aggregate ceiling; real-world throughput is typically 50 to 60% of it.
- Data source: IEEE 802.11 amendments. Key rows: 802.11ac = Wi-Fi 5 (2013, 6.9 Gbps); 802.11ax = Wi-Fi 6 (2019) and Wi-Fi 6E (2021, adds 6 GHz); 802.11be = Wi-Fi 7 (2024, 46 Gbps MLO, 4K-QAM, up to 320 MHz).

AP Placement

Field-tested design rules for AP location, cell sizing and overlap, channel planning, and high-density venues.

Why it's here. A pre-survey checklist of placement do's and don'ts: mounting, spacing, overlap targets, and density ceilings.

How to use

1. Read top to bottom as guidance; each bullet is a complete recommendation.
2. Coverage radii given are starting points (20 to 30 m open office, 10 to 15 m walled), not guarantees.

Field notes

- What it shows: five rule groups, each a heading over a bulleted list: Start with requirements, AP location, Cell sizing and overlap (≥ 2 APs at −70 dBm everywhere, 15 to 20% overlap, typical coverage radii), Channel planning (2.4 GHz only 1/6/11, co-channel spacing, prefer 5/6 GHz, DFS), and High-density venues (reduce power add APs, directional antennas, 20 to 30 clients/radio ceiling, tri-radio caveats).
- The intro says coverage radii and spacing are starting points; validate every design with a post-installation survey.
- An access point is never called a router. Per-radio capacity is a model-dependent ceiling.
- Data source: accumulated WLAN design best practice, not a single named standard.

Channel Map

A visual channel-bonding map showing, per band, how 20/40/80/160/320 MHz channels bond together, which primary (center) channel labels each bonded block, and which blocks require DFS.

Why it's here. When planning channel widths and bonding, to see at a glance which 20 MHz primaries fold into a given 80 or 160 MHz channel, and which bonds drag in a DFS sub-channel.

How to use

1. Scroll the 5/6 GHz maps horizontally. A block's number is its primary (center) channel.
2. Color/chip legend: No DFS (neutral, an attribute not a verdict), DFS (amber, radar detection required), Mixed / DFS (danger, a 160 MHz bond spanning DFS and non-DFS sub-bands, where any DFS sub-channel subjects the whole bond to DFS), PSC (lime, the 6 GHz preferred scanning channels).
3. The 6 GHz 320 MHz row shows ch 31 as the primary block and ch 63 as a dashed alternative (they overlap, so only one is used at a time).

Field notes

- What it shows: a three-option band toggle. 2.4 GHz: the 11 US channels as 20 MHz blocks, with 1/6/11 emphasized (non-overlapping) and the rest faint. 5 GHz: rows for 20/40/80/160 MHz bonded widths, each block labelled with its primary/center channel and tinted by DFS class. 6 GHz: rows for 20/40/80/160/320 MHz across UNII-5 (ch 1 to 93), with PSC channels marked.
- US-default. The full US 6 GHz band extends to ch 233 (UNII-6/7/8 follow the same bonding pattern); the map shows UNII-5 only. 6 GHz UNII-5 needs no DFS and no AFC indoors (LPI).
- Data source: US (FCC). 5 GHz DFS: No DFS = UNII-1 (36 to 48) and UNII-3 (149 to 165); DFS = UNII-2A/2C. Colors are for readability; the meaning (DFS/PSC/mixed) is what matters.

dB Reference

A decibel reference card: dB change → power/voltage ratio with rules of thumb, and common dBm anchor values with their power and real-world context.

Why it's here. Quick mental math in the field: "+3 dB is double power," "what power is +30 dBm," "what's the FCC 2.4 GHz limit in dBm."

How to use

1. In the ratio table, positive gains render in lime, losses in red.
2. Key anchors: +3 dB = 2× power; +10 dB = 10×; +30 dBm = 1 W; 0 dBm = 1 mW; −67 dBm = enterprise VoIP minimum; −70 dBm = enterprise data minimum; −80 dBm = typical Wi-Fi receiver sensitivity.

Field notes

- What it shows: dB Power Ratios: dB change (+3 to +30, −3 to −20) → power ratio, voltage ratio, and a rule-of-thumb note. Common dBm Reference Points: dBm anchors (+36 down to −100 dBm) → power (watts/mW/nW/pW) and context (regulatory limits, typical Tx powers, sensitivity floors).
- Footnote: "0 dBd is about 2.15 dBi (dipole reference). dBW = dBm − 30. Regulatory limits shown are US FCC; verify before compliance decisions."
- Mixed US (FCC) and one ETSI anchor; read the context cell for the jurisdiction of each limit.
- Data source: the dBm context column cites specific regulatory limits: FCC 6 GHz standard-power EIRP (+36 dBm, AFC required), FCC 2.4 GHz max conducted (Part 15.247, +30 dBm), FCC UNII-2A/2C and UNII-1 conducted maxes, and ETSI 5 GHz EIRP (EN 301 893, +23 dBm).

MCS Index

Look up the modulation, coding rate, and PHY data rate for any 802.11 MCS index across channel widths, for 802.11n (HT), 802.11ac (VHT), and 802.11ax (HE), scaled by spatial-stream count.

Why it's here. When you see an MCS index in a capture or client stats and want to know the modulation/coding behind it and the rate it should deliver at a given width and stream count.

How to use

1. Choose the standard (n / ac / ax) and stream count (1 to 8); rates update (rate = per-stream value × streams).
2. MCS 0 (BPSK 1/2) is the most resilient/slowest; higher MCS indices use denser modulation (up to 1024-QAM for ax) for higher rates needing better signal.
3. Cells shown as "N/A" are genuinely invalid combinations, not zero or fabricated.

Field notes

- What it shows: two selectors, 802.11 standard (n / ac / ax) and spatial streams (1 to 8). Columns per standard: 802.11n = 20 LGI, 20 SGI, 40 LGI, 40 SGI; 802.11ac = 20/40/80/160 SGI; 802.11ax = 20/40/80/160 MHz (800 ns GI).
- 802.11ac MCS 9 is invalid at 20 and 40 MHz for a single stream (shown "N/A").
- Guard-interval definitions: 802.11n LGI = 800 ns / SGI = 400 ns; 802.11ac uses SGI; 802.11ax uses 800 ns GI.
- The notes card states actual throughput is typically 50 to 65% of the PHY rate. Rates are PHY-layer maximums, not delivered throughput.
- Data source: the IEEE 802.11n/ac/ax PHY rate tables.

Modulation

Teaches what a modulation constellation is on the I/Q plane and why each step up in order (BPSK -> QPSK -> 16/64/256/1024-QAM) carries more bits per symbol but demands a cleaner link (lower EVM, higher SNR).

Why it's here. The visual companion to the MCS Index table. Reach for it when you want to understand WHY a higher MCS needs better signal, or to explain constellations, bits per symbol, and EVM to someone, rather than to look up a spec rate.

How to use

1. Read the six constellation diagrams in order; each step doubles the points and adds one bit per symbol while the points pack closer together.
2. Read the Error Vector Magnitude (EVM) explainer to see what the receiver must overcome to keep each symbol on the correct side of its decision boundary.
3. Read the summary card for the order -> bits/symbol -> SNR/EVM relationship at a glance.
4. Tap any diagram to open a full-screen pinch-zoom view for the detail-dense planes.

Field notes

- The SNR and EVM figures on the cards are REPRESENTATIVE order-of-magnitude demands, not exact 802.11 MCS thresholds. The relationship, not the precise number, is the point. Use the MCS Index tool for the spec rate/modulation table.
- EVM (Error Vector Magnitude) is the distance from the ideal symbol point to where the symbol actually landed, expressed as a percentage (or dB) of the reference amplitude. As order rises, decision boundaries shrink, so higher-order QAM demands a lower EVM and a higher SNR.
- The eight diagrams sit on an always-dark card in both light and dark modes so they never read inverted; every fact is also in the screen's prose and copy text, so the screen reads fully without the images.
- The Copy action exports the order -> bits -> SNR/EVM summary as a tab-separated table plus the representative-numbers caveat.

Non-Wi-Fi Wireless Channels

Look up the channel/frequency plans of the common non-Wi-Fi radios that share or sit beside the bands a Wi-Fi pro works in: LoRaWAN, IEEE 802.15.4, Bluetooth Classic, Bluetooth LE, and Zigbee.

Why it's here. When you're chasing interference or co-existence in the 2.4 GHz ISM band, or sizing a sub-GHz IoT deployment, and need to know where these radios actually sit.

How to use

1. Each technology is its own section.
2. LoRaWAN plans flagged with a "verify" chip are version-dependent or sparsely sourced.
3. BLE rows are ordered by physical frequency (not index) so you can see how the 3 advertising channels interleave; an "Advertising" chip marks channels 37/38/39 (2402/2426/2480 MHz).
4. Zigbee's "common picks" (11, 15, 20, 25, 26) are convention, not a mandate.

Field notes

- What it shows: one card per technology. LoRaWAN: regional plan (EU868, US915, AU915, AS923, IN865, KR920, CN470, CN779, RU864), frequency range in MHz, and a channel-plan description. IEEE 802.15.4: band (868 MHz / 915 MHz / 2.4 GHz), channel-number range, spacing, center summary, region. Bluetooth Classic (BR/EDR): 79 channels, 1 MHz spacing, f = 2402 + k MHz, 2402 to 2480 MHz, ~1600 hops/sec, global. Bluetooth LE: all 40 channels in physical-frequency order with index, frequency in MHz, kind (Advertising / Data). Zigbee: 2.4 GHz 802.15.4 ch 11 to 26, sub-GHz bands, common 2.4 GHz channel picks.
- Bluetooth, BLE, and 802.15.4 use globally-fixed channel grids. LoRaWAN frequency plans are entirely region-defined; there is no global LoRaWAN channel map.
- Plans marked "verify" (CN470 version-dependent; CN779 deprecated/limited; RU864 sparsely sourced) must be confirmed against RP002 §2 and the local regulator.
- BLE channel frequencies follow an explicit per-channel lookup, not a naive linear formula; the simple "2402 + 2×index" shortcut is a common BLE chart error.
- 802.15.4's 868/915 MHz bands are region-restricted; 2.4 GHz ch 11 to 26 are global.
- Data source: cross-checked against LoRa Alliance RP002, IEEE 802.15.4, Bluetooth SIG Core Spec, and CSA/Zigbee spec. 802.15.4 centers verified against the standard formula (ch 0 = 868.3 MHz; ch 1 to 10 = 906 + 2·(k−1); ch 11 to 26 = 2405 + 5·(k−11)).

RF Bands

A frequency map of where the common wireless technologies live in the spectrum, low to high: RFID, GPS/GNSS, cellular, the 2.4 GHz ISM crowd, and Wi-Fi across all its bands. Five spectrum neighborhoods, each with its band rows, plus a region-variance list for the bands where "what operates where" changes by regulator. A log-scale spectrum-bar plate sits at the top.

Why it's here. You design inside three or four Wi-Fi bands, but those bands have neighbors, and the neighbors are who you fight for airtime and chase for interference. This is the one screen that shows the whole map so you can see what sits just upstairs and downstairs of your channels: C-band 5G right below 6 GHz, the sub-GHz IoT radios under 2.4, the microwave oven leaking at 2.45.

How to use

1. Read top to bottom, low frequency to high. Each neighborhood card carries its band rows and a one-line takeaway on why those radios cluster there.
2. The Wi-Fi rows carry the single lime accent so your home turf stands out inside each crowded band.
3. The warning-toned region-variance list at the bottom is the part that bites: 6 GHz, sub-GHz ISM, Z-Wave center, HaLow, UHF RFID, 2.4 GHz channel count, 60 GHz WiGig, and 5 GHz DFS all change by regulator.

Field notes

- This is a frequency map, not a channel plan and not a security chart. Every band edge is a nominal allocation, not a guaranteed-clear channel. Local power limits, DFS, and licensing constrain real use further.
- The single highest-stakes variance for a Wi-Fi pro is 6 GHz: the US runs the full 5.925 to 7.125 GHz (1.2 GHz), the EU opened only 5.945 to 6.425 GHz (the lower 480 MHz), and some regions have not opened 6 GHz at all. Never assume the upper 6 GHz exists outside the US.
- The 2.4 GHz neighborhood is the most contested on the chart: Wi-Fi, Bluetooth/BLE, Zigbee/Thread, and microwave ovens all sit on top of each other in 83.5 MHz. That is why only 1/6/11 are non-overlapping in North America.
- 5G NR FR1 runs right up to about 7.125 GHz, so its C-band is the immediate downstairs neighbor of 6 GHz Wi-Fi, and FR2 mmWave shares the 24 GHz neighborhood with the 24 GHz ISM band.
- The spectrum-bar plate sits on a dark card; every fact in it is also in the tables, so the screen reads end-to-end without the image.
- Data source: every load-bearing figure cross-verified against at least two independent sources. US/FCC default; region splits are called out explicitly.

Wi-Fi HaLow

A per-section reference for IEEE 802.11ah, Wi-Fi moved down into the sub-1-GHz ISM bands for IoT: what it is, bands by region, channel widths, the headline numbers, a single-stream MCS rate table, power features, PHY/MAC, use cases, a comparison against the other IoT radios, and 2026 maturity. A channel-width plate compares HaLow's 1 to 16 MHz channels against a 20 MHz Wi-Fi channel.

Why it's here. HaLow is the Wi-Fi most Wi-Fi pros have never touched, and it is full of confidently-wrong numbers online. This is the honest read: it trades raw speed for about 1 km range, multi-year battery life, and thousands of devices per AP, while keeping native IP and WPA3. When a client asks "should we use HaLow for the sensor network," you want the real ceiling and the real caveats, not a vendor slide.

How to use

1. Read the region-lock banner first, it is the load-bearing fact. Frequency and channel width are set by each regulator, so a device certified for one region cannot legally run in another.
2. Work down through the cards: headline numbers, bands by region (with a confidence tag on the secondary-source rows), channel widths, the MCS rate table, then power, PHY/MAC, use cases, and the comparison.
3. The MCS table's peak cell (MCS 9, 256-QAM, 16 MHz, short guard interval) carries the lime accent: that is the 86.7 Mbps headline.

How it works. The clean mental model: the 802.11ac PHY clocked at one tenth. Same OFDM machinery, ten times slower clock, so symbols are 10x longer (more resilient over distance and multipath) and rates land at about a tenth of 802.11ac. Capacity comes from a 13-bit Association ID (2^13 minus 1 = 8,191 devices per AP) plus a hierarchical TIM. Power efficiency comes from Target Wake Time, Restricted Access Window, Extended Max Idle, non-TIM mode, and short MAC headers.

Field notes

- The defensible single-stream maximum is 86.7 Mbps (MCS 9, 256-QAM, 16 MHz, SGI), from the Wi-Fi Alliance overview. Use that number, NOT the contested 433.3 Mbps. Wikipedia's 433 figure is a 4-spatial-stream claim, but a 4x scaling of the WFA figure is about 347 Mbps, and first-generation HaLow silicon is single-stream. The far edge of the cell drops to about 150 kbps.
- HaLow does NOT use 2.4, 5, or 6 GHz. Lower frequencies travel farther for the same power, which is the entire reason it exists. Even the widest HaLow channel (16 MHz) is narrower than the minimum 20 MHz 2.4 GHz Wi-Fi channel; narrow channels concentrate energy, which is how it reaches farther.
- Bands by region: US 902 to 928 MHz (widest, full 1 to 16 MHz channels), EU 863 to 868 MHz (narrow, duty-cycle limited, 1/2 MHz only), AU/NZ 915 to 928 MHz. Japan, Korea, China, and Singapore carry a Medium-confidence tag because they come from secondary technical sources, not the WFA doc.
- Where it sits: more range and device count than BLE, Zigbee, and Z-Wave, and more data rate plus native IP than LoRaWAN, Sigfox, and NB-IoT. It does not match LoRa/Sigfox/NB-IoT for multi-kilometer range, and unlike carrier NB-IoT it needs its own AP infrastructure. Its strongest case is replacing short-range mesh radios with longer reach and direct IP, and carrying video where LoRa/Zigbee cannot.
- Maturity (2026): certified and shipping, with Morse Micro the clear silicon leader (MM6108, MM8108) and sub-$130 developer gateways (HaLowLink 1 at $99, HaLowLink 2 at $129). It is early-mainstream, NOT yet mass-deployed like Zigbee or BLE. "HaLow replaced Zigbee" is a roadmap claim, not a 2026 fact.
- The channel-width plate sits on a dark card; every fact is also in the tables, so the screen reads without the image.

PoE Reference

Power-over-Ethernet reference: the 802.3 PoE standards (PSE/PD power, powered pairs, class range) and the PD power classes (0 to 8) with max power at the device.

Why it's here. When sizing PoE, confirm what a switch port delivers vs what reaches the device, and which 802.3 standard / class a given AP needs.

How to use

1. PSE power is supplied at the switch; PD power is what's left at the device after cable loss (e.g. 802.3at supplies 30 W PSE, delivers 25.5 W PD).
2. The class table maps a PD's negotiated class to its max draw (e.g. Class 4 = 25.5 W = PoE+ max; Class 8 = 71.3 W = Type 4 max).

Field notes

- What it shows: PoE standards: 802.3af (PoE), 802.3at (PoE+), 802.3bt Type 3 (PoE++/4PPoE), 802.3bt Type 4 (PoE++ Hi), each with PSE watts, PD watts, powered pairs (2 of 4 or 4 of 4), and supported class range. PD power classes: class 0 to 8 → max power at the PD, the 802.3 standard defining it, and a note.
- Footnote points to the PoE Budget tool for sizing a switch against connected devices.
- Written "802.3" (not "802.3x"). Standard reference, not region-specific.
- Data source: IEEE 802.3af/at/bt.

Roaming Parameters

The 802.11k/r/v fast-roaming protocols (what each does, what it requires) plus RSSI/SNR/latency design thresholds for enterprise roaming.

Why it's here. When designing or troubleshooting roaming for VoIP/UC, confirming the protocol roles and the signal-overlap targets that make handoffs work.

How to use

1. The protocol heading shows the designation (lime) and full name.
2. In the thresholds table, the scenario word is status-tinted with a dot: green = good (the two design targets), amber = marginal (the overlap zone), red = bad (sticky-client trigger and unusable).
3. Design rules carry the actionable target (e.g. "≥ 2 APs at –67 dBm everywhere," "15 to 20% cell overlap minimum").

Field notes

- What it shows: Protocols block: 802.11r (Fast BSS Transition), 802.11k (Neighbor Report), 802.11v (BSS Transition Management), each with what it does, deployment requirements, and a field note. Thresholds block: five scenarios (VoIP/UC design target, standard data design target, roaming

overlap zone, sticky-client trigger, unusable) each with min RSSI, min SNR, roam latency, design rule, and a status verdict.
- The intro states targets vary by client hardware and AP vendor: design guidelines, not guarantees.
- Field notes flag real client behavior: some legacy clients have 802.11r compatibility issues; Android/iOS generally honor 802.11v but some Windows drivers ignore BSS-TM entirely.
- Data source: IEEE 802.11k/r/v. The reference defines exactly these three protocols; OKC is deliberately not included.

Signal Thresholds

RSSI and SNR targets: a quality scale for RSSI, minimum RSSI/SNR by application, and the SNR needed to reach each typical MCS index.

Why it's here. When validating a survey or troubleshooting a complaint: "is −68 dBm good enough for VoIP?" or "what SNR do I need to hold MCS 7?"

How to use

1. RSSI bands carry a colored verdict dot plus the quality word (green = good, amber = marginal/Fair, red = bad/Weak/Poor); the word always carries the meaning, never color alone.
2. For the application table, read across to the min RSSI and min SNR your target use case needs.
3. The SNR→MCS table tells you the SNR floor to sustain a given rate.

Field notes

- What it shows: three blocks. RSSI quality scale: Excellent (> −50 dBm) / Good (−50 to −67) / Fair (−67 to −70) / Weak (−70 to −80) / Poor (< −80). Minimum signal by application: VoIP/real-time, HD video, general browsing, email/basic data, IoT/low-rate, and location/RTLS, each with min RSSI, min SNR, and a note. SNR to MCS (80 MHz, 1 SS): the SNR floor for each MCS index (MCS 0 at 5 dB up to MCS 11 at 35 dB) with an indicative rate.
- The intro explicitly states: values vary by client hardware, environment, and AP vendor; treat as field-planning guidelines, not guarantees.
- The SNR→MCS rates are indicative (80 MHz, single stream); MCS 10/11 rates are noted as 802.11ax.
- Data source: field-planning reference thresholds, not derived from a single named standard.

Spectrum Reference

A per-band fact sheet for the three Wi-Fi bands: total usable spectrum, supported standards, channel counts, non-overlapping counts, channel widths, DFS/radar requirements, common co-existing interferers, and key deployment notes.

Why it's here. The one-screen "tell me everything about this band" reference: what lives in it, what interferes with it, what the power/DFS rules are.

How to use

1. Pick a band (2.4 / 5 / 6 GHz); read the eight facts top to bottom.
2. The colored range badge is decorative chrome paired with the band label (2.4 GHz = amber/congested, 5 GHz = blue, 6 GHz = green).
3. The "Co-existence" row names the band's common interferers or managed incumbents.

Field notes

- What it shows: each band shows a range badge plus eight key/value facts: Total spectrum, Standards, Channels (US), Non-overlapping, Channel widths, DFS / Radar, Co-existence, and Key notes.
- US-default; the footnote on every band reads "US (FCC) regulatory domain. Verify local rules before deployment."
- Carries useful specifics: UNII-1 indoor-only in some regions; UNII-2A/2C DFS implies a 60-second channel-availability delay after radar detection; 6 GHz has three US power modes, namely Standard Power (up to 36 dBm EIRP, AFC outdoors), LPI (up to 30 dBm, no AFC), and VLP (up to 14 dBm EIRP, no AFC, mobile); WPA3 mandatory on 6 GHz.
- Data source: US (FCC) regulatory domain. Values: 2.4 GHz = 83.5 MHz (US); 5 GHz = ~580 MHz usable (UNII-1/2A/2C/3); 6 GHz = 1200 MHz (5925 to 7125 MHz).

Wi-Fi Glossary

Plain-language definitions of 92 Wi-Fi terms a working engineer meets, grouped by topic and searchable live across the term, abbreviation, and definition. The same grouped, searchable screen as the authentication glossary, with the general Wi-Fi dataset.

Why it's here. When a term in a config screen, a log, or a standards document is the thing standing between you and understanding what is happening. It answers what a term means in Wi-Fi terms, in Keith's voice, without a vendor's slant.

How to use

1. Browse the terms grouped by category, or type in the search box to filter live.
2. Each entry shows the full term, its abbreviation when it has one, and a definition written for working engineers.
3. Use the copy action to grab the current view (the filtered subset when searching, otherwise the full list).

Field notes

- Multilingual: the glossary carries the term definitions and, where they exist, translated entries, so a non-English-first engineer can read the same definition in their own language. The English definitions remain the source of truth.

- Vendor-neutral by design. Definitions describe standards-based behavior, not one vendor's implementation.
- Data source: the curated 92-term Wi-Fi Glossary. The Wi-Fi Authentication Glossary below is the security-focused sibling with its own dataset.

Wi-Fi Authentication Glossary

Plain-language definitions of the Wi-Fi authentication terms, 58 of them, that a network or security pro actually meets: the 802.1X / EAP framework, RADIUS and AAA, WPA2 / WPA3 and SAE, PSK and Enterprise modes, certificates, and the supporting acronyms. Each entry pairs the full term and its abbreviation with a definition written for working engineers, grouped by topic.

Why it's here. When a term in a config screen, a log, or a standards document is the thing standing between you and understanding the authentication flow. It answers what AAA, SAE, PMF, or EAP-TLS actually mean in Wi-Fi terms, without a vendor's slant.

How to use

1. Browse the terms grouped by category (Core Authentication, EAP methods, key management, and the rest).
2. Type in the search box to filter live across the term name, abbreviation, and definition.
3. Each entry shows the full term, its abbreviation when it has one, and a definition in Keith's voice.
4. Use the copy action to grab the current view (the filtered subset when searching, otherwise the full list) as plain text.

Field notes

- Vendor-neutral by design. Definitions describe the standards-based behavior, not one vendor's implementation.
- This is the authentication-focused sibling of the general Wi-Fi Glossary; same searchable, grouped layout with its own set of terms.
- Data source: the curated Wi-Fi Authentication Glossary edition, 58 terms.

Wi-Fi Tools Comparison

A vendor-neutral, offline reference that compares professional Wi-Fi survey, design, spectrum-analysis, and troubleshooting toolkits, grouped by the activity each one serves. Each config lists its vendor, product, license model, up-front cost, and 3-year total cost of ownership, with a neutral note on what the bundle does and does not include. A per-vendor summary and a typical-toolkit roll-up ride alongside.

Why it's here. There is no neutral, capability-level map of the professional Wi-Fi tooling landscape. This is that map, built from Keith's vendor-interviewed workbook, so a leveling-up engineer can see the four activities and which tools serve which job without a vendor sales pitch.

How to use

1. Browse by activity: Design, Validation, Spectrum Analysis, then Troubleshooting. Within each activity, configs are listed alphabetically by vendor.
2. Search by vendor (e.g. Ekahau), product, activity (e.g. spectrum), capability (e.g. survey), or license model (e.g. perpetual). The match is a case-insensitive substring and narrows the activities in place.
3. Read the up-front and 3-year TCO figures as planning estimates, not quotes. Use the vendor Website and Docs links to confirm current pricing and product details before you buy.
4. A query that matches nothing shows an honest "No match" card; it never fabricates a tool. The typical-toolkit roll-up and the per-vendor summaries are always available below the activities.

Field notes

- The figures are still being confirmed with vendors, and a few may change before the final release.
- Pricing is dated. The figures are as of July 2026; confirm current pricing with each vendor, because prices, bundles, and product names change often.
- Cost figures are MODELED ESTIMATES assembled by WLAN Pros from vendor-supplied numbers and list pricing, not vendor-published quotes. Treat every dollar amount as a planning estimate, never a binding price.
- This is NOT a ranking. There is no score and no "best". Tools are listed alphabetically and the same vendor can appear under more than one activity, because real toolkits are assembled across vendors. The set reflects vendors interviewed by WLAN Pros, not every tool that exists, so an omitted tool is not a snub.
- No vendor logos or product photos appear here. Those are trademarks and copyrighted images that need each vendor's written permission, which is still being gathered. The reference is text and data only for now.
- Fully offline: the comparison is bundled in the app and works with no connection.
- Source: the Wi-Fi Design, Validation, Spectrum Analysis & Troubleshooting Tools V6 workbook (Keith Parsons, vendor-interviewed, last revised 2026-02-16).

WPA Security

A matrix of Wi-Fi security modes (WEP through WPA3-Enterprise) with encryption, key method, PMF support, and a deployment verdict, plus a reference of the advanced features that distinguish them.

Why it's here. When choosing or auditing an SSID's security, confirm a mode's cipher, key method, PMF requirement, and whether it's recommended, acceptable, or to be avoided.

How to use

1. Each mode shows a verdict chip whose color reflects the deployment recommendation, always with the word: WEP/WPA1 = "Do not deploy"/"Deprecated" (red); WPA2-Personal = "Acceptable" (amber); WPA3-Personal/WPA3-Enterprise = "Recommended" (green); Enhanced Open = "Open networks", WPA2-Enterprise = "Enterprise std" (blue/info).

2. Below, the feature rows explain the mechanisms (e.g. SAE replaces the PSK 4-way handshake with forward secrecy; OWE encrypts open networks without a password; 6 GHz requires WPA3 or OWE).

Field notes

- What it shows: Security modes: WEP, WPA (WPA1), WPA2-Personal, WPA3-Personal, Enhanced Open, WPA2-Enterprise, WPA3-Enterprise, each with category, encryption suite, key method, PMF (Not supported / Optional / Required), and a status verdict chip. Advanced features: SAE, PMF (802.11w), OWE, Forward Secrecy, 192-bit Security Mode, WPA3-mandatory-on-6 GHz, and 802.1X/RADIUS roles.
- WPA3-Enterprise's 192-bit mode (GCMP-256 + HMAC-SHA-384 + ECDH/ECDSA-384) is noted as required for government/classified deployments. 6 GHz does not permit WPA2 or older.
- The verdicts are reference guidance reflecting current best practice.
- Data source: IEEE 802.11 / Wi-Fi Alliance security standards. Verdict colors meet WCAG contrast.

Diffie-Hellman

The Diffie-Hellman key exchange taught by colors: a staged paint-mixing analogy paired with the real modular-exponentiation math, then tied to WPA3 SAE. A paint-mixing diagram sits at the top.

Why it's here. SAE is the heart of WPA3, and SAE is Diffie-Hellman wearing a Wi-Fi hat. If you understand how two parties reach a shared secret over a public channel without ever sending their private values, you understand why WPA3 resists the offline dictionary attacks that broke WPA2-PSK. This is the read-first explainer for that.

How to use

1. Read the paint analogy and the matching math side by side: public base g and modulus p are the common paint; each party's private exponent is their secret color; the mixtures (g^a mod p and g^b mod p) are public; both sides reach the same blend.
2. The eavesdropper row is the point: a passive listener sees the common paint and both mixtures and still cannot un-mix them. Recovering a private exponent is the discrete-logarithm problem, which is hard.

How it works. Public parameters: base g, modulus p. Alice computes A = g^a mod p and sends it; Bob computes B = g^b mod p and sends it. Alice then computes s = B^a mod p, Bob computes s = A^b mod p, and both equal (g^a)^b mod p = (g^b)^a mod p, the shared secret. Mixing is easy (one-way); un-mixing (the discrete log) is hard, which is the security.

Field notes

- The Wi-Fi tie-in is the whole reason this is in the kit: Diffie-Hellman is the basis of SAE (Simultaneous Authentication of Equals), the Dragonfly handshake in WPA3. It replaced the WPA2 pre-shared-key 4-way exchange and resists offline dictionary attacks, because the password is never exposed to a passive listener.

- The diagram sits on a dark card; the analogy, the math, and the WPA3 note all live in the text too, so the screen reads without the image.
- This is a fundamentals explainer, not a calculator. It does not do the modular arithmetic for you; it teaches the shape so SAE stops being a black box.

Apple Wi-Fi Support Tips

Apple's own Wi-Fi support guidance distilled into four sections: recommended router/Wi-Fi settings for Apple devices, how to run Wireless Diagnostics on a Mac, the Option-click Wi-Fi menu, and iOS/iPadOS Wi-Fi troubleshooting steps. Each section links to the Apple article it came from.

Why it's here. Most of the clients you support carry Apple gear, and Apple publishes specific Wi-Fi guidance that engineers either ignore or never find. This is that guidance in one place: what Apple actually recommends for router settings, how to pull diagnostics off a Mac, and the iOS triage path, with the source articles one tap away.

How to use

1. Read the four sections; each carries a tappable link chip to the Apple support article it was distilled from.
2. The Option-click menu section links straight to the macOS Menu-Bar Wi-Fi companion, which owns the per-field "what each RF value means" detail.
3. If a link fails to open, the screen shows the full URL so you can copy it.

Field notes

- Honesty bits carried on-screen, not hidden: Apple is silent on transmit power (so the tool does not invent a figure), and the iOS troubleshooting steps come from a single Apple source, which is flagged. Keith's own domain note is attributed to Keith, not to Apple.
- This is reference guidance, not a live read of the device. For the live association detail on a Mac, use Wi-Fi Information or macOS Menu-Bar Wi-Fi.
- Data source: Apple's Wi-Fi support documentation, footnoted to Apple support URLs.

macOS Menu-Bar Wi-Fi

The RF data a Wi-Fi pro can pull from a stock Mac without a third-party app, across four built-in paths: the Option-click Wi-Fi menu, sudo wdutil info, the Wireless Diagnostics app, and the Shortcuts "Get Network Details" action. This screen owns the per-field "what each RF value means" reference.

Why it's here. Before you reach for any third-party Wi-Fi app on a Mac, the OS already exposes most of what you need: live RSSI, noise, channel, width, PHY, Tx rate. Knowing the four built-in paths and what each field means turns any borrowed Mac into a usable Wi-Fi read. This is the reference that names them and decodes the fields.

How to use

1. Start with the four-path overview: what each path gives you and whether it needs sudo.
2. Section A decodes the Option-click Wi-Fi menu fields (what each live value means and why a pro cares). Section B walks the wdutil info Wi-Fi block. Section C lists the Wireless Diagnostics Window-menu tools. Section D covers the Shortcuts "Get Network Details" action.
3. Hold Option and click the Wi-Fi menu-bar icon to see the live association detail inline, no app required.

Field notes

- The load-bearing honesty note: sudo wdutil info masks the RF values unless you run it with sudo. Without sudo you get a redacted block; with sudo you get the unmasked RSSI/noise/MCS. The callout states this plainly.
- The airport CLI is removed on current macOS and is NOT documented as usable. Do not reach for it.
- The Shortcuts "Get Network Details" action is the one path that exposes RF fields an app cannot otherwise read, and it works on iOS too. It is the same bridge the Wi-Fi Information tool uses.
- Data source: distilled from Apple docs plus corroborating sources. Reference text only.

How Strong Is Wi-Fi, Really?

A vendor-neutral comparison of how much energy Wi-Fi puts into your body versus the everyday sun, using verified, stated numbers. No inputs and no runtime math: it is a read-along reference.

Why it's here. Wi-Fi exposure questions come up constantly. This puts the energy in perspective against a thing everyone already lives with: sunlight, using verified numbers instead of hand-waving. Using stated assumptions (ten access points at 30 dBm / 1 watt EIRP, four meters away, free-space spread), the ring of APs delivers about 0.05 W/m². The midday sun delivers about 1,000 W/m², full spectrum, which makes it about 20,000 times stronger per square meter.

How to use

1. Read the short version first: one hour of midday sun puts about as much energy into your body as roughly 2.3 years sitting inside a ring of ten Wi-Fi access points 4 meters (13 feet) away.
2. The energy-parity table shows the same idea as time: 10 seconds of sun ≈ 2.3 days in the ring, 1 minute ≈ two weeks, 1 hour ≈ 2.3 years.
3. The assumptions card states every input behind the numbers, so nothing is hidden: 10 APs at 1 W EIRP, 4 m free-space spread, the sun at 1,000 W/m² full spectrum.
4. The safety-limit card states where this sits against the FCC (47 CFR 1.1310) and ICNIRP 2020 public limit of 10 W/m²: the ring is about 200 times below it.

How it works. Power density falls off with the square of distance: density = EIRP / (4πr²), free space at 4 m, stated as approximate.

Example. At 30 dBm (1 W) EIRP and 4 m, each AP is 0.00497 W/m²; ten of them ≈ 0.05 W/m². The sun at 1,000 W/m² is ≈ 20,000× stronger, so 1 hour of sun ≈ 2.3 years in the ring.

Field notes

- Honest note on mechanism: Wi-Fi radio energy is non-ionizing. It can only gently warm tissue. It cannot cause the photochemical or DNA damage that sunburn does, at any Wi-Fi power level. This is a comparison of total energy and warmth, never of sunburn: the UV in sunlight burns; Wi-Fi has no UV and no comparable mechanism.
- The basis is 30 dBm (1 W) EIRP per AP (the US 2.4 GHz maximum) at 4 meters (13 feet). 4 m is realistic for how far a person sits from the access points around them; free-space density falls with r^2, so the figures are conservative for the point.
- Where it sits against the limit: the FCC and ICNIRP cap public RF exposure at 10 W/m². The 10-AP ring at about 0.05 W/m² is roughly 200× below that limit, and measured real-world Wi-Fi runs far lower still.
- Every figure is stated as approximate and traces to a verified source: FCC 47 CFR 1.1310 and OET-65; ICNIRP 2020 RF guidelines; ASTM G173 / NREL reference solar spectrum; WHO and IARC on non-ionizing RF (Group 2B) versus solar UV (Group 1); peer-reviewed Wi-Fi exposure surveys (PMC5927334, PMC8172712).

Regulatory Domains

The radio regulator that governs Wi-Fi in each market: name, official website, governing docs, and 2.4/5/6 GHz band and power notes for 43 jurisdictions, as a dated snapshot.

Why it's here. Working a market you do not know? Find its regulator, the governing rule, and a starting band and power note, then confirm against the official site.

How to use

1. Search by jurisdiction or regulator abbreviation. Shared abbreviations (NCC, TRA, CRA) carry the jurisdiction so they stay unambiguous.
2. The page leads with a snapshot date: the band and power notes are a dated snapshot, not settled constants.
3. Tap a regulator's website to confirm any value against the current rule text before relying on it.

Field notes

- What it shows: region-level (not per-country) FCC and ETSI rules for the Wi-Fi bands plus the ITU three-region note. Structural rules (band edges, DFS/TPC, power-class availability) are high confidence; the exact dBm figures are a snapshot.
- Regulatory-volatility caveat: the 6 GHz dBm values are being amended (VLP extended band-wide, new geofenced classes). Never a settled constant: verify before deploying or certifying.
- Data source / standard: cross-checked across FCC/ETSI documents and vendor regulatory white papers, as of 2026. Offline, read-only.

Wi-Fi Standards Bodies

Who defines, certifies, and coordinates Wi-Fi and adjacent wireless: IEEE, the Wi-Fi Alliance, ITU-R, and more, grouped by the three layers a pro keeps straight.

Why it's here. People conflate the bodies. This separates who writes the standard from who certifies products from who sets the legal rules, so you cite the right one.

How to use

1. Three layers: a standards body (IEEE 802.11) defines how the radio works; a certification body (Wi-Fi Alliance) verifies interoperability and owns the brand; a regulator sets the legal channel and power rules per country.
2. Tiles are grouped by layer, plus an IoT/adjacent group (Wi-SUN, Connectivity Standards Alliance for Zigbee and Matter, LoRa Alliance).
3. 'Wi-Fi' is a Wi-Fi Alliance trademark, not an acronym and not 'Wireless Fidelity'. The per-country rules live on the Regulatory Domains reference, cross-linked here.

Field notes

- IEEE writes 802.11; the Wi-Fi Alliance certifies and brands it. They are different organizations.
- ETSI appears here as a standards body and also as the EU's referenced harmonizer on the Regulatory Domains page.
- Data source: each body's official site (IEEE, Wi-Fi Alliance, WBA, ITU, IETF, 3GPP, Bluetooth SIG, CWNP, ETSI, Ecma, Wi-SUN, CSA, LoRa Alliance).

Cabling & Connectors (10)

Antenna Connectors

An 18-connector practical reference for Wi-Fi antenna systems: each connector's full name, reverse-polarity variant, typical Wi-Fi use, indoor/outdoor fit, coupling mechanism, body size, RF signal path, impedance, frequency rating, mating compatibility, and field notes. Plus a polarity-explained diagram, an at-a-glance comparison table, enterprise vendor trends, a to-scale size comparison, and the top 6 a Wi-Fi engineer actually meets.

Why it's here. When you are identifying or mating an antenna lead in the field. It answers the questions a tech actually asks: is this RP-SMA or SMA, will these two mate, does it cover 6 GHz, and which connector does this vendor ship.

How to use

1. Browse the connectors grouped by where they live (enterprise panel/external, outdoor/point-to-point, board-level/internal, test/cellular). Each card shows the connector name, full name, a reverse-polarity marker when it applies, and a photo or line diagram.

2. Each card lists typical use, indoor/outdoor, coupling, size, RF path, impedance, frequency, and mating, then a field-notes line.
3. Type in the search box to filter live across every field (name, vendor, coupling, size, RF path, frequency, or any word in a note).
4. Below the table: a polarity-explained diagram, a Compare-at-a-glance table (connector, size, RF path, typical use) that scrolls sideways on a phone, enterprise vendor trends, the size order largest to smallest with a to-scale diagram, and the top 6 connectors most engineers meet.

Field notes

- The key field gotcha: every connector here is 50 ohm, and an RP connector will thread onto its standard counterpart but the center contacts will not connect.
- U.FL intermates with I-PEX MHF I (same footprint); the smaller MHF 4 does NOT mate with either.
- DART is Cisco's Smart Antenna Connector, a proprietary multi-port interface that breaks out to RP-TNC, N-type or RP-N via Cisco adapter cables.
- Size is the connector body width (across-flats for threaded parts, outer diameter for board-level parts): an approximate field-recognition aid, not a precision mechanical spec.
- Some connectors show a photo; where one isn't available (N-Type, TNC, RP-TNC), a line diagram is shown instead.

Source: Keith Parsons.

Coax Cable

A coaxial cable reference: impedance, velocity factor, outer diameter, maximum usable frequency, and typical use for common RG- and LMR-series cables.

Why it's here. When specifying an antenna run, pick the right LMR size for the length and frequency, and avoid a 75 Ω mismatch.

How to use

1. Each cable is a block: name, then a spec line (impedance / VF% / diameter / max GHz), then the use note.
2. The 75 Ω entry (RG-6) is shown dimmed: it's impedance-mismatched for 50 Ω Wi-Fi and shown for reference only ("CATV/satellite, NOT for Wi-Fi").
3. Higher VF means slightly lower propagation delay and loss.

Field notes

- What it shows: per cable (RG-58, RG-8/U, RG-213, RG-214, LMR-100A through LMR-1200, RG-6): impedance, velocity factor (%), diameter (mm), max frequency (GHz), and a typical-use note.
- A footnote points to the Cable Loss tool for exact attenuation. Wi-Fi is a 50 Ω system, so the dimmed 75 Ω RG-6 is a mismatch. Max frequencies are typical maximums. Standard reference, not region-specific.

- Data source: manufacturer and industry specification values for the named cable series.

Ethernet Cable & Connector

The consolidated twisted-pair reference in one tool: the Cat5e-through-Cat8 capability chart (bandwidth, max speed, distance at 1G/10G, PoE support, shielding, typical use) plus the T568A/T568B RJ-45 pinout. This brings the category capability and the pin colors into one place, so you don't have to flip between tools. Twisted-pair side only; coax has its own Coax Cable tool.

Why it's here. When you are choosing cable for a run and then terminating it, you want the category capability and the pin colors in one place, not two. Confirm a category's bandwidth and 10G reach, the PoE++ suitability (notably the Cat6A-for-PoE++ recommendation), and which color goes on which pin, without flipping tools.

How to use

1. Cat capability chart: scroll horizontally. "N/A" in a distance column means that rate isn't supported (e.g. Cat5e has no 10G distance). Key facts: Cat6 hits 10G only to 55 m; Cat6A hits 10G to the full 100 m and supports all 802.3bt; Cat8 carries 1G/10G to 100 m but its 25/40G design rate is limited to about 30 m.
2. T568 pinout: pick the standard (T568B is the default and most common). View is "plug face, clip down, pin 1 on the left." Striped wires show a split swatch (color over white), and the swatch colors are real copper-pair colors. T568A and T568B differ only in swapping the green and orange pairs.

Field notes

- Cat chart: per category (Cat5e, Cat6, Cat6A, Cat7, Cat7A, Cat8) it shows max bandwidth (MHz), max speed, distance at 1 Gbps, distance at 10 Gbps, PoE support, shielding, and a typical-use note. Cat7/Cat7A use non-standard plugs (the ISO/IEC Class-F caveat), so they are flagged "Specialty." Cells with no applicable value show "N/A".
- PoE++ tip: bundled Cat6 running PoE++ generates significant heat; Cat6A dissipates it better, so TIA-568 recommends Cat6A for PoE++ in bundles.
- Pinout: each standard's eight pin rows show the pin number, a wire-color swatch and name (e.g. "Orange / White"), the twisted-pair number (1 to 4), and the 100/1000 Base-T function (TX+, RX-, BI-D A+, etc.). A crossover cable uses T568A on one end and T568B on the other, rarely needed today since most switches/NICs auto-MDI-X.
- Standard reference (TIA-568 / ISO 11801), not region-specific.

Fiber Optic

Fiber types (OM1 to OM5, OS1/OS2) with core/cladding, modal bandwidth, jacket color code, and supported distance at 1G/10G/40G/100G.

Why it's here. When specifying fiber, confirm a fiber type's reach at a given rate, its jacket color, and whether it's current or legacy.

How to use

1. Scroll the distance matrix horizontally. A dash in a rate column means that rate isn't supported (OM1/OM2 don't do 40G/100G).
2. OM1/OM2 are shown dimmed (legacy). Jacket colors: OM1/OM2 = Orange, OM3 = Aqua, OM4 = Aqua (violet is a manufacturer convention, not the standard), OM5 = Lime Green, OS1/OS2 = Yellow.
3. Multimode (OM) lists modal bandwidth; singlemode (OS) shows "N/A" bandwidth but reaches 10+ to 80+ km.

Field notes

- What it shows: two sub-tables. Distance by data rate, per fiber type: core/cladding, modal bandwidth (MHz·km), and supported distance at 1G/10G/40G/100G. Jacket color code & notes, per fiber type: a jacket color swatch and name, and a deployment note.
- Footnote: distances are per TIA-568/ISO 11801; actual limits depend on transceiver, splice count, and connector loss. OM3/OM4 are the current deployment standards; OM1/OM2 are legacy (dimmed). OM5's wideband window note (~1,850 to 2,470 MHz·km near 953 nm) is preserved verbatim. Standard reference, not region-specific.
- Standard: TIA-568 / ISO 11801 (cited in the footnote).

Fiber Connectors & Polish

The fiber connector form factors (LC, SC, ST, FC, MPO/MTP) with ferrule size, coupling, and use, plus the three endface polishes (PC, UPC, APC) and the two separate color systems: cable jacket color and connector body color.

Why it's here. When you are identifying, specifying, or mating a fiber connector and need to answer the questions that actually cost time in the field: is this LC or SC, will an APC mate a UPC, why is one connector green and another blue, and is that aqua jacket OM3 or OM4. The two single biggest field errors live here: mating APC to UPC, and reading the OM4 jacket color wrong.

How to use

1. Read the connector table for form-factor recognition: ferrule diameter, coupling/latch, simplex or duplex, and typical use.
2. Read the polish rows for endface type, the 8 degree APC angle, the typical return loss, and the connector body color that goes with each polish.
3. Use the two-color-system note to keep cable jacket color and connector body color separate; the same color word can mean two different things depending on which one you are reading.

Connector form factors

Connector	IEC 61754 part	Ferrule	Coupling	Form factor	Typical use
LC	61754-20	1.25 mm	Push-pull latch (RJ-style clip)	Simplex + duplex	Data center / enterprise; SFP/SFP+ transceivers; dominant today
SC	61754-4	2.5 mm	Push-pull snap	Simplex + duplex	FTTH, telecom, enterprise patching; second most common
ST	61754-2	2.5 mm	Bayonet twist-lock	Simplex only	Legacy campus / multimode LANs
FC	61754-13	2.5 mm	Threaded screw nut	Simplex only	Test equipment, precision / high-vibration
MPO / MTP	61754-7	Rectangular multi-fiber (8 / 12 / 24)	Push-pull, keyed	Multi-fiber ribbon	40G/100G/400G parallel optics; data-center trunks

LC and SC dominate new deployments; LC leads the data center because its 1.25 mm ferrule packs more ports into the same SFP density. MTP is US Conec's branded, tighter-tolerance MPO; it is mechanically intermateable with MPO, not a separate standard.

Endface polish

Polish	Endface	Return loss (typical)	Connector body color
PC (Physical Contact)	Slight dome, flat-ish	about -40 dB	Legacy, not separately color-keyed
UPC (Ultra PC)	Finer dome, no angle	about -50 dB	Blue (single-mode)
APC (Angled PC)	8 degree angled ferrule	about -60 dB (best)	Green

The APC ferrule is polished to an 8 degree angle that reflects back-reflection into the cladding, which is why it carries the lowest return loss. APC mates only to APC. Return-loss figures are typical industry numbers, not per-part datasheet guarantees.

Field notes

- HARD RULE: APC and UPC must never be mated. The 8 degree angled ferrule against a flat ferrule causes very high insertion loss and can physically damage both ferrules. Green mates only green.
- Two separate color systems exist and the page keeps them distinct. Cable jacket color (TIA-598-D): orange = OM1/OM2, aqua = OM3/OM4, lime green = OM5, yellow = single-mode. Connector body color (TIA-568/598 convention): beige = OM1 62.5/125, black = OM2 50/125, aqua = OM3/OM4, lime = OM5, blue = single-mode UPC, green = single-mode APC.
- The colors collide. Green appears in both systems: lime-green jacket means OM5 multimode, green connector body means APC angled single-mode. Aqua appears in both: OM3/OM4 jacket and OM3/OM4 connector body. Always check which system you are reading before you trust the color.

- MYTH: "OM4 is violet." TIA-598-D assigns OM4 aqua, the same as OM3. Violet ("Erika Violet") is a manufacturer differentiation convention, not the standard color. Because OM3 and OM4 both default to aqua, the only reliable way to tell them apart is the printed legend on the jacket.
- OM1 nuance: the OM1 cable jacket is orange, but the connector body convention for 62.5/125 is beige. OM1 cable is not beige.
- Cladding is always 125 microns across every fiber type (9/125 single-mode, 50/125 OM2 to OM5, 62.5/125 OM1), a useful unifying fact when you are reading core/cladding sizes.

Sources: TIA-598-D, IEC 61754 (FOA, Cisco, Fluke corroboration).

Optical Transceivers

Searchable, offline reference of 35 optical Ethernet transceiver variants (1G to 400G) grouped by speed tier, plus the 9-row SFP-to-OSFP form-factor ladder. Each variant lists its designation, data rate, reach, fiber type, wavelength, and connector, with reach as the lead, trust-first column.

Why it's here. Picking an optic, the field's first question is "which module, and how far will it go on this fiber?" This answers it without leaving the app or going online, and keeps the IEEE-vs-vendor line visible so you never quote a vendor reach as a standard guarantee.

How to use

1. Browse by speed tier. The commonly-ordered tiers (10G / 25G / 100G) surface first and are flagged "Commonly ordered"; 1G / 40G / 200G / 400G follow.
2. Search by designation (e.g. LR4), reach, fiber (SMF / MMF / OM4), wavelength (e.g. 850 nm), connector (LC / MPO), or tier (e.g. 100G). The match is a case-insensitive substring across all of those, and it narrows the tiers in place.
3. Read the REACH line first: it is the IEEE maximum on the listed fiber. The fiber type (MMF / SMF) and connector (LC / MPO) give the physical-layer specifics at a glance.
4. A query that matches nothing shows an honest "No match" card; it never fabricates a module. The form-factor table is always available below the variants.

Field notes

- IEEE vs VENDOR: variants standardized in IEEE 802.3 carry a neutral IEEE label. Vendor / coherent variants (1000BASE-EX, 1000BASE-ZX, 10GBASE-ZR, and 400GBASE-ZR) carry a VENDOR label and a "loss-budget dependent" caveat. Their reach is real and widely sold but NOT an IEEE guarantee: it depends on the link's loss budget. Never quote a vendor reach as a standard figure.
- 400GBASE-ZR is coherent DWDM (OIF 400ZR), beyond base IEEE 802.3: a metro / data-center-interconnect optic, not a base Ethernet PHY. Its 80-120 km reach is loss-budget and DWDM-line-system dependent.
- Reach figures are IEEE-standard maximums on the listed fiber; real vendor modules may differ. Multimode reach is grade-dependent (OM3 vs OM4 vs OM5), so those rows give per-grade numbers rather than one figure.

- What it contains: 35 verified variants spanning 1G to 400G across SFP, SFP+, SFP28, QSFP+, QSFP28, QSFP56, and QSFP-DD / OSFP, plus a 9-row form-factor ladder (max rate, lane count, typical power envelope). Power figures are typical vendor ranges, not a single standard: treat as guidance.
- Fully offline: the reference is bundled in the app and works with no connection.
- Verified against IEEE 802.3 standards tables and Cisco / FS.com vendor datasheets. Out-of-scope bleeding edge (800G / 1.6T, CPO / LPO, SFP-DD) was deliberately excluded as not yet stable or field-relevant.

RJ Connectors

A reference to the registered-jack connector form factors (RJ11, RJ14, RJ25, RJ45/8P8C, RJ48, RJ48C, RJ48X): positions, conductors, the modular body each uses, and its typical use. This is about the connector body, not the wiring.

Why it's here. In the field you meet the same modular bodies for very different jobs: RJ11 carries one phone line, RJ45 (8P8C) carries Ethernet, and RJ48 reuses the 8P8C body for T1/E1 with a different pin assignment. Telling them apart keeps you from cabling the wrong jack.

How to use

1. Read each connector card: the name, its modular notation (e.g. 8P8C), positions, conductors, and typical use.
2. For T568A / T568B pin-to-pair-color wiring on the RJ45 (8P8C) connector, see the Ethernet Cable & Connector tool.

Field notes

- Accuracy: "RJ45" is the colloquial name for the 8-position 8-conductor (8P8C) modular connector used for Ethernet. Strictly, RJ45 was a telephone wiring standard; the data connector is properly the 8P8C modular jack.
- The PnCm notation means an n-position body with m conductors populated, e.g. 6P2C is a 6-position body with 2 conductors (RJ11), 8P8C is fully populated (RJ45/RJ48).
- This table does NOT duplicate the T568A/T568B pin colors: that wiring lives in the Ethernet Cable & Connector tool.
- Data source / standard: the registered-jack (USOC) interface standards and the modular-connector form-factor conventions.

Cable Bend Radius & Pull Tension

The install limits that keep a copper or fiber run inside spec: minimum bend radius, maximum pull tension, and the related termination and bundling limits, with each value marked as a TIA standard or as a rule of thumb.

Why it's here. When you are pulling cable and need the two numbers that actually matter: how tight you can bend it and how hard you can pull it. Installed 4-pair UTP bends to no tighter than 4 times the cable outer diameter, and 4-pair UTP pulls at no more than 25 lbf (110 N). Get either wrong and you degrade return loss and crosstalk on a cable that still passes a quick continuity check, so the fault hides until certification.

How to use

1. Read the bend-radius rows for copper and fiber. Compute the bend radius from the cable's actual outer diameter, never a fixed inch value, because Cat6A is meaningfully fatter than Cat5e and the limit moves with it.
2. Read the pull-tension rows before a long or high-friction pull. The 25 lbf UTP number is the one to internalize.
3. Treat the TIA-marked numbers as standards and the rule-of-thumb numbers as guidance. The cable's own datasheet overrides every figure here, in either direction.

Minimum bend radius

Condition	Limit	Standard vs practice
UTP installed (horizontal, 4-pair)	at least 4x outer diameter	TIA-568 (standard)
UTP during pull / under tension	at least 8x outer diameter	ISO 11801 / common practice, NOT a TIA copper clause
Multi-pair backbone copper (25+ pair)	at least 10x outer diameter	rule of thumb
Fiber installed / no load (standard cable)	at least 10x outer diameter	rule of thumb
Fiber during pull / under tension	at least 20x outer diameter	rule of thumb

Worked example: a common Cat6 cable at about 0.25 in outer diameter gives a minimum installed bend radius of about 1 in; a fatter Cat6A at 0.30 to 0.35 in gives about 1.2 to 1.4 in. Bend-insensitive single-mode fiber (ITU-T G.657) allows far tighter bends, with minimum design radii around 10 mm (G.657.A1) down to about 2 mm (G.657.B3), which is why it dominates FTTH and dense data-center patching. Those millimeter radii are for the bare fiber's design, not the jacketed cable assembly; defer to the assembly's datasheet.

Maximum pull tension

Cable	Max pull tension
4-pair UTP (24 AWG horizontal)	25 lbf = 110 N (TIA-568 §10.6.3.2)
Fiber and multi-fiber cable	per manufacturer; strength-member dependent, no single number

Cable	Max pull tension
Multi-cable bundle pull	lower per cable; total is not the sum, so derate

The 25 lbf figure is engineered, not arbitrary: copper tolerates about 10,000 psi without significant deformation, and the 4-pair 24 AWG copper cross-section works out to about 25 lbf (attributed to Paul Kish, former chair of the TIA TR-41.8.1 copper-cabling working group). Over-pulling stretches and thins the conductors, raises attenuation, and disturbs the twist geometry the cable depends on for NEXT and return-loss balance.

Related install limits

Limit	Value	Standard vs practice
Max pair untwist at termination	0.5 in (13 mm), Cat5e through Cat8	TIA-568-B.1 §10.2.3 (standard)
Cable-tie tension	hand-tight only; must slide on the bundle; no jacket deformation	TIA-568-B.1 + BICSI TDMM (standard + best practice)
Pathway fill	40% conduit/raceway; 50% cable tray	TIA-569 (commonly specified; verify current revision + local code)
Horizontal support spacing	5 ft (about 1.5 m) between J-hooks	TIA-569 + BICSI (standard + practice)

Field notes

- The mental model is "don't kink, don't over-pull, don't over-tighten, the datasheet wins." A kink permanently changes conductor spacing inside the jacket and degrades return loss and crosstalk even after you straighten the cable; the damage does not spring back. The 4x number exists to keep you clear of the kink threshold, not because 3.9x fails and 4.0x passes.
- The "8x during pull" copper figure is ISO 11801 and field practice, not a confirmed TIA-568 copper clause. The page labels it that way; do not quote it as TIA.
- Field test for cable-tie tension: after tying, you should be able to slide or rotate the tie around the bundle. If it cannot move, it is too tight and is crushing the pair geometry. Hook-and-loop straps over zip ties for data bundles is sound practice, not a TIA mandate.
- Illustrative failure data, not a spec: below about 50 lbf UTP shows little change, at about 70 lbf the copper visibly stretches, and at about 90 to 110 lbf the cable breaks. The 25 lbf limit is deliberately conservative; failures do not start at 26 lbf, but past 25 lbf you have left the engineered safety margin.
- TIA standards set minimum performance floors. A specific cable's datasheet can be more permissive (bend-insensitive fiber) or more restrictive (large-OD Cat6A, shielded constructions), and the datasheet is the binding number. Standards numbers here are verified through TIA-citing references and shipping manufacturer datasheets, not by reading the paywalled TIA documents clause by clause.

Sources: TIA-568, TIA-569, ISO 11801, ITU-T G.657 (CommScope and Belden datasheet corroboration).

Rack Units & Mounting Hardware

The 19-inch rack standard in field terms: the U-to-inches-to-millimeters conversion, the EIA-310 vertical hole pattern, rack widths, depth and clearance, and the mounting-hardware thread types with the tapped-versus-cage-nut distinction.

Why it's here. When you are mounting an Access Point controller, switch, or patch panel and need to confirm a height in U, lay out the irregular hole pattern, or carry the right screws. Two things trip installers: the vertical holes are not evenly spaced, and not every rack is tapped. The "19-inch" label describes only the front panel; nothing inside the rack is 19 inches.

How to use

1. Use the U conversion table to translate a device's height in U to inches or millimeters. The page computes any U live: inches = U x 1.75, mm = U x 44.45. These are exact; do not round a per-U millimeter constant.
2. Read the EIA-310 hole-pattern note before laying out a multi-U faceplate, because the holes repeat in groups of three at uneven spacing.
3. Check the thread-type table and confirm the rack's hole type (tapped, square-hole for cage nuts, or unthreaded) before install day, then pack the matching screws or cage nuts.

Rack-unit conversion

U	Inches	mm	Note
1U	1.75	44.45	base unit
2U	3.50	88.90	
4U	7.00	177.80	
12U	21.00	533.40	common wall / half-height
24U	42.00	1066.80	half-rack
42U	73.50	1866.90	standard full rack (about 6 ft of rail)
45U	78.75	2000.25	taller data-center cabinet
48U	84.00	2133.60	extra-tall cabinet

1U = 1.75 in = 44.45 mm by definition (EIA-310-D / IEC 60297), a fixed value, not a measured one. 42U is the de-facto full rack; 45U is a real but less universal tall variant.

EIA-310 vertical hole pattern. The mounting holes are NOT evenly spaced. Within each 1.75 in U, three holes repeat at 0.5 in, then 0.625 in, then 0.625 in (which sums to 1.75 in), then repeat. The U boundary falls in the middle of the 0.5 in gap. A correctly designed 1U faceplate uses the outer two

holes of its group of three. Count holes wrong by one and the panel binds; multi-U gear with evenly spaced holes will not line up. In millimeters, 0.5 in = 12.70 mm and 0.625 in = 15.88 mm.

Rack widths. The 19-inch (EIA-310) rack has a front panel/flange width of 19 in (482.6 mm), but the mounting-hole horizontal spacing is 18.312 in (465.1 mm) center to center and the rack opening between posts is at least 17.72 in (450 mm). None of the internal dimensions are 19 inches. The 23-inch telecom/WECO rack is a legacy world with several incompatible conventions; gear is not cross-compatible with 19-inch.

Mounting hardware

Thread	Major diameter	Pitch	Commonly seen on
10-32 (UNF, imperial)	0.190 in	32 TPI	Dell gear, audio/AV racks, lighter equipment
12-24 (imperial)	0.216 in	24 TPI	older / general-purpose racks; historical default
M6 (metric)	about 6 mm	1.0 mm	HP/Compaq gear, most modern square-hole + cage-nut setups

These three threads are close enough in size to start in the wrong hole but will cross-thread and strip if forced; a 12-24 screw forced into a 10-32 tapped hole destroys the thread. Match the screw to the rack's tap or to the installed cage nut. Vendor mapping (Dell to 10-32, HP to M6) is a common convention, not a fixed rule; it shifts across product generations.

Rails come in three types. Tapped (threaded round holes) take a screw straight in but are fixed to one thread type, and a stripped thread kills that position. Square-hole + cage nut clips a captive spring-steel nut into a square hole (about 3/8 in / 9.5 mm), converting it to a threaded hole of whatever spec you choose; it is thread-agnostic and strip-proof (replace the nut, not the rail) and is the modern default. Round unthreaded holes need the right clip nuts or nut-and-bolt hardware.

Field notes

- The #1 first-install mistake is assuming the rack is tapped. Many modern racks ship as bare square holes with no cage nuts included. Show up without cage nuts and you cannot mount anything; confirm the hole type before install day.
- "19-inch" describes only the front panel width. The opening is about 17.72 in and the hole spacing is 18.312 in.
- The 10-32 versus 12-24 mix-up is the most damaging hardware error: the two are visually near-identical and cross-thread and strip if forced. When in doubt, use a square-hole rack plus the right cage nut.
- U is height only. A 1U switch and a 1U server can have very different depths, so depth is a separate, independent check. Most network gear is shallow, so a 600 mm cabinet usually fits, but usable rail-to-rail depth runs roughly 100 to 150 mm less than the cabinet's external depth (lost to doors,

hinges, rear panel, and cable bend radius). Check usable depth against your deepest device with cables attached.
- Dimensional claims are triangulated across multiple independent sources that agree; the defined values (1.75 in, 44.45 mm, the hole pattern) are uncontested. The paywalled EIA-310-D and IEC 60297 standard texts were not read clause by clause.

Sources: EIA-310-D, IEC 60297 (NavePoint, AudioRax, RackSolutions corroboration).

Screw Drives & Driver Bits

The drive faces a network or Access Point installer actually meets on enclosures, brackets, racks, and outdoor gear: the common drives (slotted, Phillips, Pozidriv, hex, Torx, Robertson) and the security/tamper drives, with the bit you need for each and the Phillips-versus-Pozidriv distinction.

Why it's here. When you are opening or mounting gear and need to recognize a drive face and carry the matching bit. Two things cost field time: Pozidriv mistaken for Phillips (the wrong bit cams out and chews the head), and security/tamper drives on outdoor and public-space enclosures that need a specific bit you will not have unless you packed it. This covers drive types, not thread pitch or head shapes.

How to use

1. Match the drive face to the table to find the bit and size. PH1/PH2, PZ1/PZ2, T10/T15/T20/T25, and Robertson #1/#2 cover most network gear.
2. Before any public or outdoor job, read the security-drive section and pack a tamper-bit set. A standard bit set does not include these.
3. Use the Phillips-versus-Pozidriv tick-mark rule to pick the right cross bit before you strip a head.

Common drives

Drive	Typical bit / size	Where you see it on network gear
Slotted	blade matched to slot width	terminal blocks, grounding lugs, legacy brackets
Phillips (PH)	PH1, PH2	indoor AP covers, bracket screws, rack cage nuts
Pozidriv (PZ)	PZ1, PZ2	EU enclosures, DIN-rail gear, PDUs, EU mount kits
Combo (slotted/Pozi)	PZ2 or flat	electrical enclosures, "electrician's screws"
Hex (metric)	2.5 to 6 mm	antenna/pole mount set screws, bracket joints
Hex (imperial)	3/32 in to 1/4 in	US-sourced mounts, rack hardware
Torx	T10, T15, T20, T25	enclosures, rack ears, outdoor AP housings
Robertson (square)	#1 (green), #2 (red)	Canadian sites/hardware, ceiling work

Security / tamper drives. These show up on outdoor AP enclosures, public-space mounts, ceiling cages, and locked NEMA boxes. Each needs its matching security bit, which a standard set does not include.

Drive	What it looks like	Tool needed
Security Torx (Torx TR / pin-in Torx)	a normal Torx star with a small post (pin) in the center; a solid Torx bit will not seat	Torx security bit with a hole bored down the center, sized T10H to T40H
Pin-in hex (security hex)	a normal hex socket with a pin in the center	hex security bit with a center hole
One-way / clutch	slotted-looking head with curved ramps; turns to tighten, slips to loosen	flat blade to install; removal needs extraction (drill / specialty tool)
Tri-wing	three-bladed pinwheel/triangular recess	tri-wing bit
Spanner / snake-eye	two round holes ("snake eyes") on the face	spanner / pin-spanner bit with two matching pins

Field notes

- Phillips versus Pozidriv, the distinction that saves heads: a Pozidriv head has four shallow radial tick marks set at 45 degrees between the cross arms (a faint starburst); a Phillips head is a clean cross with no tick marks. Extra 45 degree tick lines means Pozidriv, use a PZ bit; clean cross means Phillips, use a PH bit. They are not interchangeable, and the wrong bit cams out and strips the head. Pozidriv is the European "electrician's screw" standard, common on enclosures and EU-sourced mounting kits.
- MYTH: "Phillips was designed to cam out to prevent over-torquing." False. The original 1933 patent explicitly sought a recess with no tendency to cam out. Cam-out is a byproduct of the angled, tapered walls, not a design goal.
- Say T-numbers, not "star." "Star bit" is a lay term that spans 6-point Torx, 5-point pentalobe, and security variants, which are different drives. Specify the T-number (for example T20) to avoid the mismatch.
- Torx Plus (the IP series) has squarer lobes for higher torque. A standard Torx driver fits a Torx Plus screw loosely and degraded, but a Torx Plus driver will not fit a standard Torx screw. It is rare on network gear; the note exists to prevent a forced mismatch.
- Hex keys come in metric (mm) and imperial (inch/fractional) series, and they are not cross-compatible; a 5 mm key is loose in a 3/16 in socket. Carry both.
- Robertson color-coding (yellow #0, green #1, red #2, black #3) is a genuine trade and manufacturer convention, not an ISO standard. The square socket's slight taper grips the bit so the screw hangs on the tip one-handed, a real advantage on overhead ceiling work.
- Pack tamper bits before the job, not at the site. Security drives exist specifically so a standard bit will not work; if you did not pack the matching bit, you do not open the enclosure. A consolidated set (security Torx T10H to T40H plus pin-hex, tri-wing, and spanner) covers the vast majority of what an installer meets.

- The governing standards are confirmed by number (ISO 8764 for cross-recess/Pozidriv, ISO 10664 for Torx/hexalobular, ISO 4762 for hex socket cap screws, ISO 2380 for slotted), but the clause-level tip dimensions behind the paid ISO documents were not quoted.

Sources: ISO 8764, ISO 10664, ISO 4762, ISO 2380 (patent text, ToolGuyd, Polycase corroboration).

Protocols (19)

Association Sequence

The frame-by-frame 802.11 association and roaming sequences, showing the order and direction of frames between the STA, AP, RADIUS server, and DHCP server.

Why it's here. When analyzing a capture or explaining an association/roam, confirm what frame should come next and which entities exchange it.

How to use

1. Pick a scenario (Open / WPA2-PSK, WPA3-SAE, WPA2-Enterprise (802.1X/EAP), 802.11r Roam); read the phases top to bottom.
2. Each frame carries a neutral type code: MGMT (management frame), EAP (EAP / EAPOL key), WIRED (RADIUS/DHCP over the wire), DHCP.
3. The legend at the bottom expands each code. Type is carried by the text code, not by color.

Field notes

- What it shows: a scenario selector with four scenarios. Each scenario is broken into named phases (e.g. Probe & Auth, Association, 4-Way Handshake, EAP Authentication, DHCP); each frame shows a step number, direction, frame name, a frame-type code, and an explanatory note.
- These are representative sequences, not exhaustive; optional/passive-scan paths and EAP-method round-trips are summarized (e.g. the WPA3 DHCP phase is shown as one combined Discover/Offer/Request/Ack line "identical to WPA2 flow").
- Data source: IEEE 802.11 association/handshake behavior. Reflects the real frame exchanges (e.g. SAE Dragonfly commit/confirm for WPA3, the 4-way handshake, 802.1X/EAP over RADIUS, 802.11r FT over-the-air). Standard reference, not region-specific.

802.11 Reason Codes

The 802.11 deauthentication/disassociation reason codes (RC) and association status codes (SC) that appear in captures, with a searchable filter.

Why it's here. When a capture shows a deauth with reason code 15 or an association response with status 17, and you need the plain-language meaning fast.

How to use

1. Type a code number or keyword (e.g. "15" or "handshake") to filter; a "no match" card appears if nothing matches.
2. Reason codes (RC) appear in Deauthentication and Disassociation frames; status codes (SC) appear in Authentication, Association, and Reassociation Response frames.
3. Code 0 in the status group is the success value, rendered green.

Field notes

- What it shows: reason codes grouped by theme: Common (1 to 9), Capability/Channel mismatch (10 to 11), Security frame/element errors (13 to 14, 17 to 22, 24), Security handshake failures (15, 16, 23), QoS/load management (34 to 39), Fast Roaming/802.11r (45 to 48). Plus a separate Association Status Codes group (the most-common subset, 0 to 104), where code 0 ("Successful") is highlighted in green.
- The status-code list is the "most common" subset, not the full table.
- Codes and meanings are reproduced verbatim from the standard; nothing invented.
- Data source / standard: IEEE 802.11-2020 §9.4.1.7 (reason codes) and §9.4.1.9 (status codes), cited in the footnote.

HTTP Status Codes

The HTTP response status codes, grouped by class, with a plain-English meaning for each. A fast offline lookup when a captive portal, web service, proxy, or API returns a code and you need to know what it means.

Why it's here. When a check returns 403 or 503, or a captive-portal probe comes back 511, and you want the meaning without leaving the toolbox or going online.

How to use

1. Type a code number or keyword (e.g. "404" or "redirect") to filter; a "no match" card appears if nothing matches.
2. The filter matches the code number, the reason phrase, and the plain-English meaning, so "timeout" finds 408 and 504.
3. Use the toolbar copy action to copy the full reference as tab-separated text, one section per class.

Field notes

- What it shows: codes grouped by class: 1xx Informational, 2xx Success, 3xx Redirection, 4xx Client Error, 5xx Server Error. Each row is the code number, its reason phrase, and a short meaning.
- 511 (Network Authentication Required) is the signature of a captive portal: the network blocks access until the client authenticates.
- 418 is registered in the IANA registry as "(Unused)". It is widely known as the "I am a teapot" joke code from RFC 2324; the tool labels it honestly and notes the history.
- Data source: the IANA HTTP Status Code Registry (the authoritative registry), fetched 2026-06-04. Most codes are defined by RFC 9110 (HTTP Semantics). Code numbers and reason phrases are

verbatim from the registry; the plain-English meanings are written for this tool. Unassigned and obsoleted codes are omitted; nothing is invented.

Speed Test Services

A curated, offline reference to the popular internet speed tests, framed on the two axes that actually change the number: single-stream vs multi-stream, and a nearby CDN edge vs a distant true server.

Why it's here. Two speed tests on the same connection can disagree by a wide margin, and the reason is almost never the connection. It is how many streams the test opens and how far away the server sits. This page lets you pick the right test for the question you are asking, and read a surprising result correctly instead of blaming the Wi-Fi.

How to use

1. Read each service against the two teaching axes: single vs multi-stream, and nearby-edge vs distant-server.
2. Search by name (e.g. Ookla, Fast.com, Cloudflare); a query that matches nothing shows an honest "No match" card.
3. Check the "Runs on" note before trusting a brand as independent: Waveform runs on Cloudflare, Fast.com on Netflix's CDN, ISP tests on Ookla or M-Lab. Tap a website chip to run that test.

Field notes

- Data-per-test figures are the weak column. Each carries a confidence marker ("est.", "rough est.", "measured"), and a persistent band states they are community-measured estimates, not vendor-published numbers.
- Not all of these are independent measurement backends. Where a brand rides on another service's network, the card shows a "Runs on" note.
- Orb is a continuous monitor, not a one-shot test; the Toolbox's own Network Quality tool is the analog and reports no single composite score.
- Fully offline: the service list is a bundled dataset, so the page renders with no network call. Vendor wordmarks render on a neutral chip so they read in both light and dark.

OSI Model

The 7-layer OSI reference model: layer number, name, one-word function, PDU, example modern protocols, and typical hardware.

Why it's here. Localizing a fault: which layer is failing tells you which tool to reach for.

How to use

1. Scroll the table horizontally. Read by layer number.
2. Example mappings: L3 Network = Routing, Packet, IPv4/IPv6/ICMP/IPsec, router/L3 switch; L2 Data Link = Framing, Frame, Ethernet (802.3)/Wi-Fi (802.11)/802.1Q/ARP, switch/AP/bridge/NIC; L1

Physical = Bits, Bit, RF/fiber/copper, cable/radio/hub.

Field notes

- What it shows: the 7 layers, top (7 Application) to bottom (1 Physical), each with: layer number (lime index), name, a one-word function keyword, PDU (Data/Segment/Packet/Frame/Bit), example protocols, and typical hardware.
- Footnote: PDU = protocol data unit; layers 5 to 7 are commonly grouped as "data" in TCP/IP practice; ARP is widely placed at Layer 2 (some texts call it L2/L3), and it resolves L3 addresses to L2 addresses.
- The function column is a neutral keyword (no custom mnemonic). Standard reference, not region-specific.
- Data source / standard: ISO/IEC 7498-1:1994 plus standard IETF/IEEE protocol-to-layer mappings.

PLMN ID Reference

An offline, searchable, grouped lookup of US Public Land Mobile Network identifiers. Each row pairs a carrier or operator with its MCC, MNC, full PLMN ID, and operational status, covering MCCs 310-316 (US mainland plus Puerto Rico, Guam, the US Virgin Islands, and American Samoa).

Why it's here. When you have a PLMN ID, MCC, or MNC off a cellular scan, a SIM, or a private-LTE / CBRS deployment and need to know which carrier it belongs to, or you need a carrier's code to configure or verify a private cellular network. It works fully offline, so it is reliable in the field where there is no data connection.

How to use

1. Browse the entries grouped by MCC (310 through 316), each group sorted ascending by PLMN ID.
2. Type in the search box to filter live by code (MCC, MNC, or full PLMN ID) or by carrier / operator name.
3. Each row shows the PLMN ID, the MCC/MNC pair, the carrier (and parent operator when different), and the operational status.
4. Use the copy action to grab the current view (the filtered subset when searching, otherwise the full table) as plain text.

Field notes

- MNC and PLMN ID are strings with significant leading zeros (e.g. MNC 004, PLMN ID 310004). A two-digit and a three-digit MNC are different codes; never read them as numbers.
- Status values include operational, not operational, reserved, and unknown. A reserved or not-operational allocation may still appear on the air or in old records, so the status column matters.
- US-only by design. The dataset covers ITU region 3xx (MCCs 310-316); it does not include non-US carriers.
- Data source: US MCC/MNC (PLMN ID) allocations verified against the live Wikipedia 'Mobile country code' tables (2026-06-05).

Top-Level Domains

A curated reference to the DNS top-level domains a network or IT pro actually meets, grouped by registry type: generic (gTLD), country-code (ccTLD), sponsored/restricted, infrastructure, and notable newer gTLDs. Each entry shows the TLD, its type, and a short managed-by / typical-use note.

Why it's here. Knowing a TLD's type tells you who runs it and what to expect: a sponsored domain like .gov or .edu is eligibility-verified, .arpa is reverse-DNS infrastructure you never register, and .io or .ai are country-code domains used generically rather than true gTLDs.

How to use

1. Scroll the grouped cards, or use the Type filter to narrow to one registry class.
2. Read the TLD (lime, left) then its note. Copy exports the full curated set as a table, regardless of the active filter.

Field notes

- Curated, not exhaustive: the live root zone has roughly 1,500 generic TLDs and about 250 country-code TLDs. This lists the meaningful, field-relevant set.
- Accuracy: .io (British Indian Ocean Territory), .ai (Anguilla), .co (Colombia), .tv and .me are technically country-code TLDs commonly used generically. They are NOT true generic TLDs, and the notes say so.
- Data source / standard: the IANA Root Zone Database for registry classification and sponsoring organizations; ICANN for new-gTLD program facts.

Well-Known Ports

Searchable, offline reference of 86 curated TCP/UDP ports a network or Wi-Fi pro meets in the field. Search by port number or by service-name / description substring.

Why it's here. At a packet capture or writing a firewall rule, you ask two questions: "what runs on port N?" and "what port does service X use?" This answers both without leaving the app or going online.

How to use

1. Type a port number (e.g. 443) for an exact-port lookup, or a service name / keyword (e.g. radius, dns, vpn) for a case-insensitive substring match against both the service name and the description.
2. An empty search box lists all 86 curated ports, sorted ascending by port number.
3. Each result shows the service name, the port number with its protocol label (TCP, UDP, or both), and a one-line description.
4. A query that matches nothing shows an honest "No match" card; it never fabricates a port.

Field notes

- What it contains: 86 curated entries spanning ports 1 to 27017 (e.g. 53 dns TCP/UDP, 67 dhcp UDP, 123 ntp UDP, 443 https TCP/UDP where UDP/443 carries HTTP/3 QUIC, 1812 radius UDP and 1813 radius-acct UDP for 802.1X / WPA2-Enterprise, 3389 rdp). Each entry: port number, protocol(s), short service name, one-line description.
- Protocols in the table are TCP and/or UDP only (no SCTP entries present, though the schema supports it). Combinations: 38 TCP-only, 26 UDP-only, 22 TCP+UDP.
- This is a curated subset of the IANA Service Name and Transport Protocol Port Number Registry, trimmed to field-relevant ports, not the full ~49,000-entry registry. Absence means "not in our curated set," which the screen states plainly rather than inventing a row.
- Fully offline: the table is a bundled asset loaded and indexed once at startup; numeric lookups are instant via a port index. Works on every platform.

IP Address Reference

IANA/IETF special-use address blocks for IPv4 and IPv6 (CIDR prefix, what each block is reserved for, and the defining RFC) plus the IPv6 text-notation rules.

Why it's here. When you see an unexpected address on a link, this tells you fast whether it is private, loopback, link-local, documentation, or multicast, and which RFC governs it.

How to use

1. Scroll the IPv4 and IPv6 special-use tables. Read by CIDR prefix.
2. Check the IPv6 notation card for canonical-form and zero-compression rules before reading or writing an IPv6 address.

Field notes

- What it shows: the IANA IPv4 and IPv6 Special-Purpose Address Registries (private-use, loopback, link-local, documentation, and more), each row carrying its defining RFC.
- The multicast blocks 224.0.0.0/4 (IPv4) and ff00::/8 (IPv6) are not in the special-purpose registries: they live in the separate IANA multicast registries and are sourced to RFC 5771 / RFC 1112 and RFC 4291 section 2.7, flagged with a footnote.
- Data source / standard: IANA IPv4/IPv6 Special-Purpose Address Registries (fetched 2026-06-08) cross-checked against each defining RFC; IPv6 notation per RFC 5952. Offline, read-only.

Subnetting / CIDR Table

A /0 through /32 lookup: prefix length, dotted subnet mask, total addresses, usable hosts, and wildcard (inverse) mask.

Why it's here. The fast way to read off a mask, host count, or wildcard for a prefix without doing the powers-of-two in your head.

How to use

1. Find the prefix length (/n) row and read the mask, total addresses, usable hosts, and wildcard across it.
2. For a specific network, broadcast, or host range, use the IPv4 Subnet Calculator tool.

Field notes

- What it shows: every prefix from /0 to /32. Total addresses for a /n is 2 raised to the power (32 minus n); usable hosts is that total minus 2, plus the dotted mask and the inverse (wildcard) mask.
- Exceptions honored: /31 has 2 usable host addresses (point-to-point, RFC 3021), not 0; /32 is 1 host (single-host route, RFC 4632), not -1. Both rows carry an inline note.
- Data source / standard: pure CIDR arithmetic cross-referenced to RFC 4632 section 3.1. Offline, read-only.

Naming & Addressing Conventions

Hostname / DNS-label rules, MAC EUI-48 and EUI-64 format, the U/L (universal/local) and I/G (individual/group) bits, and the OUI/CID concept.

Why it's here. When a hostname is rejected or a MAC looks locally administered or multicast, this names the rule and the bit that decides it.

How to use

1. Read the hostname rules for label length, allowed characters, and max FQDN length.
2. Use the MAC-format and U/L, I/G bit tables (and the first-octet bit-field diagram) to tell a universally administered address from a locally administered one, and an individual address from a group/multicast one.

Field notes

- What it shows: hostname/FQDN rules (RFC 952, RFC 1123 section 2.1, RFC 1035), MAC EUI-48/EUI-64 format, the U/L and I/G bit positions in the first octet, and OUI/CID assignment.
- The first-octet bit-field diagram is decorative for screen readers: every fact it depicts (U/L and I/G positions and meanings) is also in the bit table.
- Data source / standard: IEEE Std 802-2014 with RFC 5342 section 2.1 as the freely available IETF restatement, RFC 4291 Appendix A (Modified EUI-64). The IEEE Registration Authority is the sole assigner of OUI/CID. Offline, read-only.

DNS Record Types

The DNS resource-record TYPE codes a network or Wi-Fi pro meets: numeric TYPE code, what the record does, and the RFC that defines it.

Why it's here. When a lookup returns an unfamiliar record type, this says what it carries and where it is defined.

How to use

1. Scroll the table. Read by record type (A, AAAA, CNAME, MX, TXT, SRV, CAA, HTTPS/SVCB, and more) or by numeric TYPE code.

Field notes

- What it shows: each DNS resource-record TYPE with its numeric code, a plain-English purpose, and its defining RFC.
- Current registry state reflected: CAA is now governed by RFC 8659 (obsoleting RFC 6844); HTTPS/SVCB by RFC 9460.
- Data source / standard: the IANA DNS Resource Record (RR) TYPEs registry plus the defining RFC cited per row. Offline, read-only.

DHCP Options

The DHCPv4 option codes a network or Wi-Fi pro meets, led by Option 138 (CAPWAP-AC), how a lightweight AP learns its WLAN controller address from DHCP, plus the Option 53 message-type table.

Why it's here. Reading a DHCP capture or a scope config: this names the option code and what value it carries, with the controller-discovery option called out first.

How to use

1. Read the option-code table (Option 138 first, then standard options in code order) for code, name, and purpose.
2. Use the Option 53 message-type table to decode DISCOVER / OFFER / REQUEST / ACK and the rest.

Field notes

- What it shows: common DHCPv4 option codes with names and plain-English purpose, plus the Option 53 DHCP message types.
- Option 138 (CAPWAP-AC, RFC 5417) leads because controller discovery is the option most relevant to a Wi-Fi deployment.
- Data source / standard: the IANA BOOTP/DHCP Parameters registry; base options RFC 2132, relay agent info RFC 3046, domain search RFC 3397, message types RFC 2132 section 9.6. Offline, read-only.

HTTP Methods & Headers

The HTTP request methods with their safe / idempotent properties, plus the common request and response headers met checking a captive portal, web service, proxy, or API.

Why it's here. Diagnosing a captive portal or an API call: this says whether a method is safe to retry and what a given header means.

How to use

1. Read the methods table for each method and whether it is safe (read-only) and idempotent (repeat = same effect).
2. Read the request- and response-header tables to decode the headers you see in an Inspector (HTTP Header) result.

Field notes

- What it shows: HTTP request methods (GET, HEAD, POST, PUT, PATCH, DELETE, OPTIONS, and more) with safe/idempotent flags, plus common request and response headers.
- Definitions: safe = read-only, no intended state change; idempotent = repeating the request has the same effect as making it once.
- Data source / standard: the IANA HTTP Method and HTTP Field Name registries; RFC 9110 (HTTP Semantics) for method semantics; PATCH per RFC 5789. Offline, read-only.

DSCP / QoS Markings

The Wi-Fi-to-wired QoS mapping: the four WMM Access Categories, the 802.11 User Priority values they carry, and the DSCP markings RFC 8325 recommends so wired and wireless agree.

Why it's here. QoS only works end to end when the wired DSCP markings and the Wi-Fi access categories line up: this is that mapping, with the default-mapping trap flagged.

How to use

1. Read the mapping table for each WMM Access Category, its User Priority values, and the recommended DSCP marking.
2. Read the warning callout beneath the table: the common default mapping demotes voice (EF/DSCP 46) into the video queue.

Field notes

- What it shows: the four WMM Access Categories (Voice, Video, Best Effort, Background), their 802.11 User Priority values, and the DSCP code points (name, decimal, binary) RFC 8325 recommends.
- The voice-into-video trap is rendered as a warning callout, not buried in a footnote, because it is the misconfiguration this page exists to surface.
- Data source / standard: RFC 8325 (Mapping Diffserv to IEEE 802.11), IEEE 802.11e / 802.11-2020, IEEE 802.1Q, Wi-Fi Alliance WMM; DSCP values per RFC 2474 / 2597 / 3246 / 5865. Offline, read-only.

802.1X / EAP Types

The EAP methods a supplicant and authentication server negotiate inside 802.1X, compared across credential type, server-cert requirement, client-cert requirement, and mutual authentication, with a typical-use note.

Why it's here. Choosing or auditing an 802.1X method: this is the side-by-side of what each EAP type needs and protects.

How to use

1. Read the matrix for each EAP method (EAP-TLS, PEAP, EAP-TTLS, EAP-FAST, EAP-PWD, EAP-SIM/AKA, and more) across its four security axes.
2. Read the warning banner: clients that skip server-certificate validation on a tunneled method (PEAP/TTLS/FAST) are exposed to evil-twin credential theft.

Field notes

- What it shows: the EAP methods, their credential model, whether they require a server certificate, a client certificate, and mutual auth, plus a typical-use note.
- Note: EAP-FAST is RFC 4851; RFC 7170 defines TEAP, its standards-track successor (not the same method).
- Data source / standard: each method's defining RFC (5216 EAP-TLS, 5281 EAP-TTLS, 4851 EAP-FAST, 5931 EAP-PWD, 4186 EAP-SIM, 4187 EAP-AKA, 5448 EAP-AKA'). Offline, read-only.

802.11 Feature Matrix

A Wi-Fi 5 to Wi-Fi 7 capability comparison across the features that matter in design: bands, channel width, modulation, OFDMA, MU-MIMO, BSS Coloring, TWT, MLO, preamble puncturing, spatial streams, and the theoretical PHY-max rate.

Why it's here. Sizing what a generation can and cannot do, side by side, without hunting through four standards.

How to use

1. Read each feature row across the four generation columns (Wi-Fi 5 / 802.11ac, Wi-Fi 6 / 802.11ax, Wi-Fi 6E, Wi-Fi 7 / 802.11be).

Field notes

- What it shows: feature presence/absence and the stream/width/QAM ceilings per generation, plus a theoretical PHY-max rate row.
- The Max PHY rate is a theoretical ceiling, not a real-world client rate. It carries a 'ceiling' chip and a footnote with the math (e.g. Wi-Fi 7 ~46 Gbps = 16 streams x 320 MHz x 4096-QAM, with no shipping client above 2-4 streams). Never an achievable speed.

- Data source / standard: IEEE 802.11ac/ax/be; 802.11be (Wi-Fi 7) values are from the draft approaching final approval and may shift (flagged in the footnote). Offline, read-only.

Date / Time Standards

A reference for ISO 8601 / RFC 3339 date-time formats, UTC-offset notation, the Unix epoch and the 2038 problem, leap seconds (UTC vs TAI), and NTP stratum levels.

Why it's here. Reading a log timestamp, an API field, or an NTP status: this decodes the format and the time-scale behind it.

How to use

1. Read the format cards for ISO 8601, the strict RFC 3339 Internet profile, and the common format tokens.
2. Read the epoch, leap-second, and NTP-stratum cards for the time-scale context behind a raw timestamp.

Field notes

- What it shows: ISO 8601 and RFC 3339 formats and the difference between them, UTC-offset notation, the Unix epoch and 2038 wraparound, the UTC-vs-TAI leap-second relationship, and NTP stratum levels (1-15 valid; stratum 0 is the kiss-o'-death / unspecified marker).
- Honesty note: the UTC-TAI offset of -37 s is a static educational value current as of the 2017-01-01 leap second, shown with a 'static value' badge and a footnote pointing at the IERS Bulletin C: never presented as a live truth.
- Data source / standard: ISO 8601-1:2019, RFC 3339, RFC 5905 (NTP), POSIX.1-2017, BIPM/IERS. Offline, read-only.

Data Units

Bit vs byte units, the SI (decimal) vs IEC (binary) prefix ladders, kB vs KiB through EB vs EiB, and how link rates in bits relate to storage in bytes.

Why it's here. The reason a '1 TB' drive shows as ~931 GiB, and the reason a 100 Mbps link is not 100 MB/s, kept in one place.

How to use

1. Read the SI vs IEC ladder for each rank (kB/KiB through EB/EiB), its powers, byte counts, and the percent divergence.
2. Use the bit-vs-byte card and the divide-by-8 rule to convert a bit-rate to byte throughput.

Field notes

- What it shows: the two parallel prefix ladders (SI base-1000 vs IEC base-1024), the per-step divergence (compounding ~2.4 points per rank), and the bit-vs-byte relationship.

- Note: SI uses lowercase k for kilo (kB) but uppercase for M and above; IEC binary prefixes are Ki, Mi, Gi, Ti, Pi, Ei. Ki is a capital K, unlike SI's lowercase k. Network/link speeds are quoted in bits per second; storage in bytes.
- Data source / standard: BIPM SI prefixes and IEC 80000-13 binary prefixes. Offline, read-only.

Encoding (5)

ASCII / Hex / Binary

The full 128-character US-ASCII table with decimal, hex, octal, and binary for each code, plus supplementary quick-reference tables for reading hex dumps and protocol fields.

Why it's here. When decoding a hex dump or a protocol field, look up a byte's character, its four numeric representations, or the meaning of a control code.

How to use

1. The numeric columns (dec/hex/oct/bin) and the glyph render in a monospaced font so octets and look-alike characters (l/I/O/0) read unambiguously.
2. Control rows show a mnemonic (NUL, LF, CR, ESC…); printable rows show the glyph (space is shown as "SP").
3. Filter by typing a decimal, hex (with or without 0x), octal, binary, glyph/mnemonic, or keyword.

Field notes

- What it shows: a "how to read this" card, then Control codes (0 to 31, plus 127) and Printable characters (32 to 126) as tables with Dec / Hex / Oct / Bin / Char / Description. Plus supplementary cards: Range boundaries worth memorizing, Newlines on the wire, The case bit (0x20), Nibble → hex map, Powers of two, Hex place values, and High range (128 to 255): no single "extended ASCII".
- The high-range card is explicitly honest: ASCII stops at 127; bytes 128 to 255 mean different things depending on the encoding, so there is no single "extended ASCII." It documents UTF-8 (bytes 0 to 127 identical to ASCII; 128 to 255 are part of multi-byte sequences), ISO-8859-1/Latin-1, and Windows-1252 (a common mojibake source), with the rule "bytes 0 to 127 are portable; 128 to 255 are not, so know the encoding before decoding."
- Data source / standard: RFC 20 (US-ASCII) for the 128 values; high-range guidance per ISO-8859-1, Windows-1252, and the Unicode/UTF-8 spec. Standard reference, not region-specific.

Top 30 Emoji

The 30 most-used emoji, ranked 1 to 30, with the official Unicode CLDR name and a descriptive "common use" note.

Why it's here. A lightweight decode reference for what people actually mean by an emoji, useful when meanings drift (e.g. 💀 = "that's hilarious," not death).

How to use

1. Ranked most-used first (😂 face with tears of joy at #1).
2. The official CLDR name is the row's spoken/searchable key; the glyph is excluded from the screen-reader label (readers announce the glyph's own name).
3. The "common use" note describes typical reading, with generational/fandom context where it matters.

Field notes

- What it shows: per emoji: rank (1 to 30), the glyph, the Unicode CLDR official name, and a "common use" note on how people usually read it today.
- The intro is explicit: "common use" is how people usually read each emoji today, not an official Unicode definition; meanings drift by audience, region, and generation.
- The literal/codepoint fields are intentionally omitted. The glyph renders via the platform color-emoji font (Apple Color Emoji on iOS/macOS).
- Data source / standard: Unicode CLDR for the names; ranking is messaging-weighted (private-messaging keyboard frequency, not social-listening-weighted).

Markdown Cheatsheet

CommonMark and GitHub Flavored Markdown syntax shown as the literal text you type next to what it renders as, covering headings, emphasis, links, images, lists, task lists, blockquotes, code, and tables.

Why it's here. When writing docs, READMEs, wikis, or notes, confirm the exact Markdown for a structure without guessing, and know which pieces are GitHub extensions that a plain renderer may not support.

How to use

1. Each row pairs the literal Markdown ("You type") with what it renders as. The "You type" column is plain text, not rendered, so the exact syntax is copyable.
2. Rows marked GFM are GitHub Flavored Markdown extensions (tables, task lists, strikethrough, autolinks) and may not work in a plain CommonMark renderer.
3. Gotchas: put a blank line between block elements, escape special characters with a backslash, and end a line with two trailing spaces for a hard line break.

Field notes

- Covers the CommonMark core plus the widely-implemented GFM extensions, with GFM-only rows flagged.
- Data source / standard: the CommonMark specification and the GitHub Flavored Markdown specification.

Hash Lengths

Common hash algorithms by output size (bits / bytes / hex characters), family, and security status, with MD5 and SHA-1 flagged broken / deprecated.

Why it's here. Sizing a digest field or judging an algorithm choice: this is the length and the security verdict at a glance.

How to use

1. Read each algorithm's output length in bits, bytes, and hex characters, plus its family and security-status chip.

Field notes

- What it shows: MD5, the SHA-1, SHA-2, and SHA-3 families, and more, each with output length and a security-status verdict.
- MD5 and SHA-1 carry an explicit broken / deprecated status; the verdict word always accompanies the status color, so color is never the sole carrier of meaning.
- Data source / standard: NIST FIPS 180-4 (SHA-1, SHA-2), FIPS 202 (SHA-3 / Keccak), RFC 1321 (MD5); deprecation per NIST SP 800-131A and the 2017 SHAttered SHA-1 collision. Offline, read-only.

Regex Cheatsheet

Regular-expression syntax (anchors, character classes, quantifiers, groups and references, and alternation / escapes) scoped to the common PCRE2 (Perl-compatible) dialect.

Why it's here. The token you half-remember, looked up fast, with a flag where a dialect would behave differently.

How to use

1. Scroll the syntax tables by section: anchors, character classes, quantifiers, groups/references, and alternation/escapes.
2. Watch for the DIALECT badge: a token marked that way behaves differently outside PCRE2; check the per-row dialect note.

Field notes

- What it shows: the common PCRE2 regex token set with meaning and examples, grouped by function. A 'Dialect: PCRE2 (Perl-compatible)' banner sits at the top.
- Honesty note: there is no single normative regex authority: POSIX (BRE/ERE), PCRE2, ECMAScript, Python, Java, Go (RE2), and .NET differ. No token is presented as universal unless the source data marks it so; non-universal tokens carry a DIALECT badge.

- Data source / standard: the common PCRE2 subset per the PCRE2 syntax reference. Offline, read-only.

CLI & Capture (3)

Linux / WLAN Commands

A grouped Linux command reference for WLAN work: file/process basics, modern and legacy networking, the wireless-specific tools (iw, iwconfig, airmon-ng, rfkill), tested monitor-mode sequences, and the macOS non-root packet-capture setup.

Why it's here. You're driving a WLAN Pi, a Linux capture box, or a survey laptop and need to put an adapter into monitor mode on a specific channel/width, or recall the exact iw syntax to read the current link.

How to use

1. Commands are grouped (File, Directory, Process, Network, Wireless, Monitor-mode, macOS capture).
2. Filter by command or group name; a group-label match surfaces the whole group.

Example. Commands as shipped. Wireless: iw dev, iw dev wlan0 info (type/channel/mode), iw dev wlan0 link (SSID/signal/rate), iw dev wlan0 scan (sudo), iw dev wlan0 set channel 6, iw phy (PHY caps), iwconfig (legacy), iwlist wlan0 scan (legacy), rfkill list, rfkill unblock wifi. Monitor-mode: sudo airmon-ng start wlan0 (creates wlan0mon), sudo airmon-ng start wlan0 36 (monitor + channel 36), sudo airmon-ng stop wlan0mon, sudo airmon-ng check kill, sudo ifconfig wlan0 down / sudo iwconfig wlan0 mode monitor / sudo ifconfig wlan0 up (3-step), sudo iw dev wlan0 set channel 36 HT40+ (40 MHz secondary above), sudo iw dev wlan0 set channel 40 HT40- (40 MHz secondary below), sudo iwconfig wlan0 mode managed, sudo iw dev wlan0 info, lsusb, sudo dmesg, sudo ethtool -i wlan0, lsmod. macOS capture: sudo dseditgroup -o edit -a USERNAME -t user access_bpf (non-root capture), dscl . read /Groups/access_bpf (verify membership), sudo wdutil info (macOS 14+).

Field notes

- Caveat: iwconfig/iwlist/ifconfig are legacy "wireless extensions" tools; modern distros prefer iw and the iproute2 suite; monitor-mode sequences need sudo and a capable adapter/driver.
- Footnote: wireless extensions are deprecated in favor of iw + nl80211 (both shown because field gear still ships the legacy tools); HT40+/HT40- selects a 40 MHz channel with the secondary 20 MHz channel above (+) or below (-) the control channel; the access_bpf group lets a non-root macOS user capture via libpcap (BPF devices). Replace USERNAME and wlan0 with your actual user and interface.
- Source / basis: targets Linux primarily, with a macOS-capture group. Cross-checked against Linux man-pages, iw/nl80211 docs, and aircrack-ng docs.

Network CLI Commands

A three-column Windows, macOS, and Linux command reference for the everyday network-troubleshooting tasks (reachability, path tracing, DNS, interface config, sockets, ARP, routing, Wi-Fi link state), each with the field-common flags. A trailing Linux-only "shell essentials" group covers the capture-rig / WLAN Pi context.

Why it's here. You're at a client site on whatever laptop is in front of you and need the right command for this OS without looking it up: "what's the macOS equivalent of ipconfig /all," "how do I see the connected SSID/BSSID/RSSI from the CLI on Windows," "what's the Linux version on the WLAN Pi."

How to use

1. One card per task. Each card shows the Windows command, the macOS command, and the Linux command in three separate columns, plus a one-line description and a flag subset. Where macOS and Linux genuinely match, they read identically; where they diverged, each column carries its own command.
2. Filter by command name or task (e.g. "ping" or "DNS"). Where a platform has no native command, the card says so honestly rather than blanking. WLAN-relevant tasks lead.

Example. Commands as shipped (Task | Windows | macOS | Linux | key flags): Connected Wi-Fi interface state | netsh wlan show interfaces | wdutil info (sudo for full RF) | iw dev wlan0 link | show interfaces (Win), sudo wdutil info (macOS, unmasked RF), iw dev wlan0 link (Linux). List visible Wi-Fi networks | netsh wlan show networks mode=bssid | wdutil info | iw dev wlan0 scan (sudo) | (macOS removed the airport CLI). Reachability/RTT via ICMP echo | ping host | ping host | ping host | -t (Win continuous), -n count (Win), -c count (nix), -i interval. Trace L3 path | tracert host | traceroute host | traceroute host | -d (Win), -m max (nix), -I (nix ICMP), -T (Linux TCP SYN). DNS query full detail | (no native command) | dig name | dig name | +short, -x addr, @server. Interface IP config | ipconfig /all | ifconfig | ip addr | /all, /release, /renew, /flushdns. Active connections | netstat -ano | netstat -an | ss -tunap | -a, -n, -o (Win PID). ARP cache | arp -a | arp -a | ip neigh | -a, -d addr. IP routing table | route print | netstat -rn | ip route | print (Win), -rn (macOS).

Field notes

- The 3-column split is deliberate: macOS and Linux have diverged enough (ifconfig vs ip, netstat vs ss, DHCP renew, flush DNS) that folding them into one "macOS/Linux" column would ship a wrong command on one of the two platforms. Where they are identical, the two columns simply read the same.
- The caveat warns that some commands need administrator/sudo rights and that the flags shown are the field-common subset, not exhaustive.
- The footnote notes that ifconfig, route, arp, iwconfig, and netstat are legacy on Linux (modern distros prefer the iproute2 suite: ip addr, ip route, ip neigh, iw, ss); on macOS use wdutil info (sudo) or the Wireless Diagnostics app for Wi-Fi link details; and netsh wlan is Windows only.
- Source / basis: data consolidated from Keith's Network CLI sheet plus the WLAN Pros Linux cheat sheets, reconciled against current Windows/macOS/Linux docs. The macOS Wi-Fi entry shows only

wdutil info; the deprecated airport CLI was removed entirely.

Wireshark 802.11 Filters

Copy-ready Wireshark display filters (typed into the filter bar after capture) and capture filters (BPF syntax, applied during capture) for 802.11 analysis: frame type/subtype, addressing, BSSID/SSID, RadioTap metadata, and RSN cipher/AKM selectors. Now also carries the 802.11 status-code and reason-code lookup tables next to the filters, so the moment a filter surfaces a deauth or a failed assoc, the code's meaning is right there.

Why it's here. You have a capture open and need the exact display-filter field to isolate deauths, beacons, a specific BSSID, or a security cipher, without guessing field names from memory. Then, once the deauth is on screen, you need to know what reason code 15 actually means without leaving the tool.

How to use

1. Filters are grouped; filter the list by syntax or task, and a group-label match surfaces the whole group.
2. The syntax is selectable for copy.
3. Below the filters, the status-code and reason-code tables list the highest-frequency 802.11 codes: status codes appear in Auth/Assoc responses; reason codes appear in Deauth/Disassoc frames.

Example. Filters as shipped. Frame type/subtype (display): wlan.fc.type == 0 (all management), == 1 (all control), == 2 (all data); wlan.fc.type_subtype == 0 (Assoc req), 1 (Assoc resp), 2 (Reassoc req), 3 (Reassoc resp), 4 (Probe req), 5 (Probe resp), 8 (Beacon), 9 (ATIM), 10 (Disassoc), 11 (Auth), 12 (Deauth), 13 (Action), 24 (Block Ack Req), 25 (Block Ack), 26 (PS-Poll), 27 (RTS), 28 (CTS), 29 (Ack), 36 (Null data), 40 (QoS data), 44 (QoS Null). Address (display): wlan.addr == aa:bb:cc:dd:ee:ff (any field), wlan.ta, wlan.ra, wlan.sa, wlan.da. BSSID/SSID: wlan.bssid == ..., wlan.ssid == "MyNetwork", wlan.ssid contains "Guest". RadioTap: radiotap.channel.freq == 2412, radiotap.datarate >= 6, radiotap.dbm_antsignal > -70, radiotap.dbm_antnoise < -90, radiotap.channel.freq >= 2400 && < 2500 (2.4 GHz), >= 5000 && < 5900 (5 GHz), >= 5925 && <= 7125 (6 GHz). Capture filter (BPF): type mgt, type ctl, type data, type mgt subtype beacon, type mgt subtype probe-req, type mgt subtype deauth, type ctl subtype rts, type ctl subtype ack, wlan host aa:bb:cc:dd:ee:ff. RSN cipher (display): wlan.rsn.pcs.type == 4 (CCMP-128, 00-0F-AC:4), == 8 (GCMP-128, 00-0F-AC:8), == 9 (GCMP-256, 00-0F-AC:9), wlan.rsn.gcs.type == 2 (group cipher TKIP, 00-0F-AC:2). RSN AKM (display): wlan.rsn.akms.type == 1 (802.1X, 00-0F-AC:1), == 2 (PSK, 00-0F-AC:2), == 8 (SAE / WPA3-Personal, 00-0F-AC:8), == 18 (OWE, 00-0F-AC:18).

Field notes

- Caveat: display-filter field names match Wireshark's dfref; capture filters use libpcap/BPF "type/subtype" syntax and only work when capturing with a RadioTap/PPI header.
- Footnote: type_subtype is the combined value (type in the high bits, subtype in the low bits) matching IEEE 802.11 frame type/subtype assignments; capture filters require capturing with a

RadioTap header (monitor mode); for the full RSN cipher/AKM number-to-name map, see the RSN groups or the WPA Security reference tool. The bundled status-code and reason-code tables list the highest-frequency 802.11 codes only (the full tables live in the 802.11 Reason Codes reference tool).

- Two deliberate corrections are baked in. (1) The RSN cipher-suite vs AKM tables were rebuilt from IEEE 802.11-2020 Tables 9-149 (cipher = wlan.rsn.pcs.type / wlan.rsn.gcs.type) and 9-151 (AKM = wlan.rsn.akms.type) because the original source card mislabeled cipher values as AKM. (2) The 5 GHz/2.4 GHz/6 GHz band filters ship a deliberate safe fallback using documented radiotap.channel.freq ranges instead of the unverified radiotap.channel.flags.5ghz child-token. Band-edge detail: the 5 GHz range stops at < 5900 and the 6 GHz range starts at >= 5925, so center frequencies in the 5900 to 5924 MHz gap fall into neither band filter. This is intentional.
- Source / basis: targets Wireshark display-filter (dfref) and libpcap/BPF capture-filter conventions, sourced from the Wireshark dfref, the RadioTap dfref, pcap-filter(7), and IEEE 802.11-2020.

Checklists (2)

How to NOT Have a Wireless Problem

A run-through-it AP install checklist organized into Before / Install / After phases, most of which is wired-side verification, on the premise that many "wireless" problems are not wireless problems.

Why it's here. You're mounting and turning up an access point and want to confirm the cabling, PoE, VLAN, DHCP, and routing path are right before you blame the radio, then document and validate after.

How to use

1. Three phases; tap each item as you complete it; the top count tracks progress.
2. Remember the premise: many wireless problems are not wireless problems. Work these checks before, during, and after the install, using a LinkSprinter, LinkRunner AT, EtherScope, or CyberScope, or just a laptop with a command window and the right commands.

Example. Before Installing Access Point: Cable meets/exceeds Cat5e specs · Total cable distance with patch cords < 100 m · PoE meets the AP's specific requirements · Check 802.3 af, at, or bt · Confirm DHCP address & VLAN · Confirm correct VLAN assignment · Confirm access or trunk port as required · Confirm default gateway · Ping default gateway · Confirm target IP addresses reachable · Confirm DNS reachable · Confirm target DNS addresses reachable · Management VLAN assigned & available. Install Access Point: Install access point (kept as its own one-item phase). After Installing Access Point: Document AP's MAC & assigned name · Document AP's location · Document AP's switch / port used · Document AP's IP address · Confirm AP installed in proper orientation · Confirm external antennas installed correctly · Wait for access point to receive configuration · Wait for a 2nd reboot of the AP if needed · Listen in air for all SSIDs being broadcast · Connect client to each SSID · Check each SSID for proper VLAN & IP pool.

Field notes

- Checked state is not saved; it resets when you leave the screen.
- Source / basis: Keith Parsons / WLAN Pros original card (© 2024 WLAN Pros). The "After Installing" list is renumbered to a clean 1-12 (the original card was gap-numbered with no item 2).

Wi-Fi Client Testing Checklist

An ordered, twelve-step client-side connectivity test to run from a client device after an install or when triaging a connectivity complaint.

Why it's here. A user reports "Wi-Fi is broken." This walks you from "can the client even see the SSID" through association, auth, DHCP, gateway/DNS reachability, data rate, and a speed test, in the order failures actually cascade.

How to use

1. One flat ordered list (no phases). Tap each as you complete it; the count tracks progress.

Example. Items in order: Can see all SSIDs being broadcast · Associate to target SSID · Complete SSID authentication · Receive an IP address via DHCP · Receive default gateway & DNS · Ping default gateway · Ping DNS · Ping remote IP address · Ping remote DNS address · Check client MCS · Check client Tx data rate · Complete network speed test.

Field notes

- Per-session only; not saved.
- Source / basis: Keith Parsons / WLAN Pros original card (© 2024 WLAN Pros).

Guides (2)

Dual Orbs on WLAN Pi

A step-by-step how-to that turns a WLAN Pi R4 or M4+ into two Orb sensors (one testing the wired connection over Ethernet, one testing Wi-Fi) by installing a bundled Debian package. The screen walks the copy / install / reboot flow, covers the cloned-image identity reset, shows how to reconfigure the Wi-Fi credentials, and lists the service-management commands. A Download button exports the bundled wlanpi-dual-orb_1.1.3_all.deb via the share sheet.

Why it's here. When you want continuous, two-path connection monitoring (wired and wireless) from one small device you already carry, this gets a WLAN Pi running two free Orb sensors in a few commands. The results show up in your own Orb account; the Toolbox just gives you the guide and the package.

How to use

1. Tap Download wlanpi-dual-orb.deb and save it (Files / AirDrop / Mail). Keep the filename; the install command refers to it by name.

2. Copy the package to your WLAN Pi: scp wlanpi-dual-orb_1.1.3_all.deb wlanpi@:~
3. SSH to the Pi and install it: sudo apt install ./wlanpi-dual-orb_1.1.3_all.deb (it prompts for the Wi-Fi SSID, password, and encryption).
4. Reboot: sudo reboot. A boot-time service installs Orb and starts both sensors.
5. If you cloned the image onto more than one Pi, run sudo orb-reset-identity on each so the Orb dashboard sees a distinct new sensor.
6. To change the Wi-Fi credentials later, run sudo orb-wifi-configure.

Field notes

- What it installs: the free, open-source Orb sensor on the Pi, with its services running as root. You view results in your own Orb account, free for up to 5 devices. Nothing of Orb's is redistributed by the Toolbox; the bundled package only sets the sensor up.
- Tested on the WLAN Pi R4 and the M4+.
- On cloned images the OrbIDs and hostnames carry over from the original; sudo orb-reset-identity deletes the old ones and derives fresh IDs and names from the WLAN Pi hostname. Use sudo orb-reset-identity --dry-run to preview without changing anything.
- The Wi-Fi sensor runs in its own network namespace (orb-wifi); inspect it with sudo ip netns exec orb-wifi iw dev or sudo ip netns exec orb-wifi ip link show.
- Guide and package by Ferney Munoz. Orb: orb.net. WLAN Pi project: wlanpi.com.

FreeRADIUS on WLAN Pi

A step-by-step guide to standing up a lab RADIUS server on a WLAN Pi for learning and testing 802.1X (WPA2/WPA3-Enterprise), built around a bundled install script you copy to the Pi and run there.

Why it's here. Reach for it when you want a working RADIUS server to practice 802.1X against without building a full production AAA stack. The screen shows the install script inline, lets you download it, and points at the two files to customize (the shared secret and the user list).

How to use

1. Read the lab caveat: the script ships test accounts (student01-student10) with cleartext passwords and a shared secret named secretwlanpros. Change the secret and use real credentials before anything real.
2. Download install_freeradius.sh (or copy it from the inline view).
3. Copy it to the Pi: scp install_freeradius.sh wlanpi@:~
4. Make it executable: chmod +x install_freeradius.sh
5. Run it on the Pi: ./install_freeradius.sh
6. Verify with the listed radtest / journalctl / tcpdump commands.

Field notes

- The Toolbox runs no shell. This is reference text plus a file you run on your own WLAN Pi, not in the app. The inline script and the download are the exact same bytes.
- Lab use only as shipped: cleartext-password test accounts and a known shared secret. Change the secret in /etc/freeradius/3.0/clients.conf and replace the test users in /etc/freeradius/3.0/users before using it on a real network.
- The script configures PEAP/MSCHAPv2, opens UDP 1812, and runs a radtest against student01 to confirm the server answers.
- Guide and install script by Ferney Munoz; bundled with permission.

Power & Cooling (6)

Power Phasing

Reference for the three power topologies a field tech meets in IT spaces: US single-phase 120V (one hot + neutral), split-phase 120/240V (two hots 180 degrees apart, 240V hot-to-hot, 120V hot-to-neutral), and three-phase wye 208V (three hots 120 degrees apart, 208V line-to-line, 120V line-to-neutral). Renders the 208-vs-240 distinction installers confuse.

Why it's here. When you read a panel or nameplate, confirm which topology you're on and what hot-to-hot voltage to expect, so you don't feed 240V-rated gear from a 208V wye (about 13 percent low) or mislabel split-phase as two-phase.

How to use

1. Each topology card shows its hots, line-to-neutral (L-N), line-to-line (L-L), and phase angle, with a waveform.
2. Split-phase is 240V hot-to-hot because the two legs are 180 degrees apart (2 x 120 in anti-phase); three-phase wye is 208V hot-to-hot because the legs are 120 degrees apart (the square root of 3 times 120, about 208V).
3. Use the 208V-vs-240V table to settle the confusion: same 120V hot-to-neutral, different hot-to-hot, different source topology.

How it works. For a wye system, line-to-line voltage equals the square root of 3 times the line-to-neutral voltage: 120 x 1.732 ≈ 208V. For split-phase, line-to-line is 2 times line-to-neutral in anti-phase: 2 x 120 = 240V.

Example. A 240V single-phase heater fed from a 208V wye panel sees about 13 percent lower voltage and runs at reduced output, because two legs of a 208V wye are 120 degrees apart (208V), not 180 degrees apart (240V).

Field notes

- What it shows: three topology cards (single-phase 120V, split-phase 120/240V, three-phase wye 208V) with hots, L-N, L-L, phase angle, where it's seen, and a clarifying note; plus a 208V-vs-240V comparison table and a why-it-matters note.

- Split-phase is single-phase, NOT two-phase: the two phasors don't define a rotating field. The center conductor is the grounded, current-carrying neutral (center tap), not the equipment ground.
- 208V and 240V are not interchangeable: they differ by about 32V and arise from different topologies (a single-phase center tap vs two legs of a 3-phase wye).
- Written "Access Point" not "router". Nominal voltages per NEC Article 100/210.
- Data source: NEC Article 100/210, split-phase and three-phase electrical-engineering references, and the line-to-line = square root of 3 times line-to-neutral relationship.

Ohm's Law & Power Wheel

Reference for the V / I / R / P relationships every field tech reaches for: the four core identities (V = I x R, P = V x I, P = I^2 x R, P = V^2 / R), the 12-segment power wheel that expresses each of V, I, R, P in terms of any two of the others, and the single-phase vs three-phase power formulas with the power-factor term.

Why it's here. When you need to solve for a missing electrical quantity at a panel or on a nameplate, find the right form of the wheel and read off the relationship, without re-deriving it. The power-factor caveat keeps you from reading V x I as watts on an AC reactive load.

How to use

1. The core relationships table gives the four identities and what each solves for; everything else derives from V = I x R and P = V x I.
2. The power wheel lists each of V, I, R, and P in three forms, so you can pick the one written in terms of the two quantities you already know.
3. The single-phase vs three-phase table carries the apparent-power (VA) and real-power (W) formulas; real power includes the cos(phi) power-factor term.

How it works. Apparent power S = V x I (single-phase) or square root of 3 x V_LL x I_L (three-phase balanced). Real power P = S x cos(phi). Power factor cos(phi) = 1 only for purely resistive loads and DC; for reactive loads it is less than 1, so V x I gives VA, not W.

Example. A 230V single-phase load drawing 5A has an apparent power of 230 x 5 = 1150 VA. If its power factor is 0.8, real power is 1150 x 0.8 = 920 W, which is why UPS and PDU equipment is rated in both VA and W.

Field notes

- What it shows: the four core identities, the 12-form power wheel (four quantities x three forms), and the single-phase vs three-phase power formulas with a power-factor note.
- Power factor (cos phi) is 1 only for resistive loads and DC, where P = V x I is exact. On reactive loads (motors, compressors, switch-mode supplies) V x I gives apparent power in VA, not real power in watts; do not read P = V x I as universally giving watts in AC.
- Notation: I^2 / V^2 use the caret, the square root is written out, and the power-factor angle is cos(phi). Written "Access Point" not "router".

- Data source: the standard electrical-engineering identity set, the 12-form Ohm's-law wheel, and the AC real/apparent/reactive power relationships.

Cooling & Thermal

Reference for the heat-load conversions you need when sizing cooling for IT spaces: watts to BTU/hr to tons of refrigeration, the IT-load-becomes-heat-becomes-cooling relationship, and the standard sensible-heat airflow (CFM / delta-T) formula.

Why it's here. When you have to size a closet or rack cooling plant from an IT load, convert that load to BTU/hr or tons and read the airflow a given temperature rise needs, so the cooling equipment nameplate matches the heat the gear produces.

How to use

1. The conversion table gives common loads in watts, BTU/hr, and tons, all derived from two anchors: 1 W = 3.412 BTU/hr and 1 ton = 12,000 BTU/hr.
2. The IT-load-to-heat-to-cooling chain shows that essentially all IT power becomes heat, so the cooling load in watts equals the IT load in watts before margin.
3. The airflow note rearranges the standard-air sensible-heat formula so you can find the CFM a target supply-to-return rise needs.

How it works. BTU/hr = watts x 3.412. Tons = BTU/hr / 12,000 (so 1 ton is about 3,517 W). Sensible-heat airflow: BTU/hr = 1.08 x CFM x delta-T (deg F), so CFM = BTU/hr / (1.08 x delta-T). The 1.08 constant is for standard moist air at sea level and shifts with altitude and air density.

Example. A 1,000 W (1 kW) IT load is 3,412 BTU/hr, about 0.284 tons. Removed across a 20 deg F supply-to-return rise it needs about 158 CFM of airflow.

Field notes

- What it shows: a watts/BTU-hr/tons conversion table, the IT-load-to-heat-to-cooling relationship, and the sensible-heat airflow (CFM / delta-T) guidance. No diagram: the relationships are numeric and read cleanly as tables.
- Essentially all electrical power an IT device draws is dissipated as heat into the room, so the cooling load equals the IT load before any headroom for losses and growth.
- The 1.08 airflow constant is the standard-condition approximation, not a universal: verify against site altitude and air density before sizing equipment. "BTU/hr" not "BTUH". Written "Access Point" not "router".
- Data source: the anchor conversions 1 W = 3.412 BTU/hr and 1 ton = 12,000 BTU/hr, plus the standard sensible-heat airflow relationship.

IEC Power Connectors

Reference for the IEC 60320 appliance couplers (C1/C2 through C19/C20, including the C13/C14 PC cord and the C15/C16 kettle cord) and the IEC 60309 industrial pin-and-sleeve connectors, with current ratings, temperature classes, and keying.

Why it's here. When you grab a cord or read an inlet on a PDU, server, or UPS, confirm the coupler pair and its rating so you match cord to inlet, and tell the 70 degC C13/C14 PC cord apart from the 120 degC C15/C16 kettle cord.

How to use

1. The IEC 60320 table lists each coupler pair with its current rating, maximum temperature, nickname, and typical use. Odd number = cord connector (female); even = appliance inlet (male), one greater than its mate.
2. The IEC 60309 table maps connector color to its voltage band; the earth-pin clock position (in 30-degree steps) is mechanical keying so incompatible voltages cannot mate.
3. Use the notes to settle the kettle-cord confusion and to confirm both color and earth-pin hour must match for two IEC 60309 devices to connect.

Example. A C13 connector (female, on the cord) mates with a C14 inlet (male, on the back of a PC or PDU). A C15 cord fits a C14 inlet, but a C13 cord will not fit a C16 inlet because the C15/C16 keying notch blocks it.

Field notes

- What it shows: the IEC 60320 appliance-coupler table (pair, current, max temp, nickname, use) and the IEC 60309 industrial-connector table (color, voltage band, use), with keying notes. A connector-face diagram slot is reserved for a later graphics pass; the tables ship fully working without it.
- The "kettle cord" nickname properly belongs to C15/C16 (120 degC hot-condition, keyed by a notch), NOT C13/C14 (70 degC cold-condition "PC cord").
- IEC 60309 red spans 380-480V (not a single "415V"): it covers 400V European and 480V US three-phase. Both color AND earth-pin clock hour must match to mate.
- Data source: IEC 60320 and IEC 60309-2.

NEMA Connectors

Reference for North American NEMA straight-blade and locking plug/receptacle configurations: the designation decoder (L prefix, configuration code, current rating, P/R) and the device groups by voltage class with phase, wiring, and amp rating.

Why it's here. When you read a NEMA designation on a plug, receptacle, or PDU, decode it correctly instead of misreading the leading number as a voltage, and confirm the phase and wiring so you do not mistake split-phase for three-phase.

How to use

1. Walk the decoder left to right across a designation like L21-30P: L (locking), 21 (a configuration code, not a voltage), 30 (the amp rating), P (plug; R is receptacle).
2. The device-group tables list types by voltage class (125V, 208/240/250V) with phase, pole/wire count, and amp rating.
3. Use the phase flags to keep the single-phase split 14-series distinct from the three-phase wye L21-series.

Example. L21-30P decodes to L (twist-lock) + 21 (three-phase wye 120/208V, 4-pole 5-wire) + 30 (30A) + P (plug). L21-30R is its receptacle. The leading number is a configuration code, so do no arithmetic on it.

Field notes

- What it shows: the designation decoder with a worked example, plus device groups (125V straight-blade and locking, and 208/240/250V) listing type, voltage, phase, wiring, and amps. A face-diagram plate slot is reserved for a later graphics pass; the tables ship fully working without it.
- The leading number is a voltage/pole/phase CLASS code, not a literal voltage: 21 = three-phase wye 120/208V, 4-pole 5-wire. Never read it as a voltage.
- The 14-series is single-phase SPLIT (the 4th pin is neutral), NOT three-phase; only the L21-series is three-phase wye. P = plug (male), R = receptacle (female).
- Data source: the NEMA configuration and nomenclature references.

International Power Plugs

Reference for the IEC World Plugs letter system (Types A through M) by region, with the underlying national standard, nominal voltage class and frequency, and current rating, plus the CEE 7 European family breakout.

Why it's here. When you travel or spec gear for a site abroad, confirm the plug type, nominal mains voltage, and frequency you will meet, so power adapters and equipment ratings match what is at the wall.

How to use

1. Search by country at the top (e.g. Germany shows Type C, F at 230V/50Hz). Common names and abbreviations work too, such as USA, UK, or Holland; multi-type countries show all their letters.
2. The plug-type table lists each letter with its standard, voltage class, current rating, and the countries that use it.
3. The CEE 7 family breakout shows how the European C / E / F types and the CEE 7/7 hybrid relate.
4. Read the voltage class as nominal: most of the world is 230V except North America and Japan at 120V. Watch the warnings (for example, Argentine Type I has line and neutral reversed).

Example. A laptop charger rated 100-240V works on both a US Type A/B 120V outlet and a UK Type G 230V outlet with only a mechanical plug adapter; a 120V-only appliance needs a voltage converter, not just an adapter, on a 230V supply.

Field notes

- What it shows: the IEC World Plugs Type A through M table (type, standard, voltage class, current, countries) and the CEE 7 European family breakout. A face-diagram slot is reserved for a later graphics pass; the tables ship fully working without it.
- Voltage classes are nominal: 120V in North America and Japan, 230V across most of the rest of the world. Confirm frequency (50 vs 60 Hz) for motor-driven and timing-sensitive gear.
- Argentine Type I (IRAM 2073) has line and neutral reversed relative to the Australian Type I; the CEE 7/7 plug is a hybrid designed to fit both French (E) and Schuko (F) sockets.
- Data source: the IEC World Plugs letter system and national standards.

Ham Radio (6)

The band-dependent amateur-radio references, grouped so all the ham material is findable in one place. Every numeric value traces to FCC Part 97 (eCFR Title 47), the FCC Online Table of Frequency Allocations, ARRL, or NCVEC, current as of June 2026, with the corrected figures used throughout: never an older chart. The pure-math ham tools (Antenna Length, Maidenhead Grid) live under Calculators & Tools → Ham Radio.

US Amateur Band Plan

The US amateur bands from HF to SHF, by license class: each band's frequency range, the Technician / General / Extra privileges, the power limit, and the mode segments.

Why it's here. A Wi-Fi pro crossing into amateur radio needs to know what they may transmit where and at what power. This is the band plan with the current, corrected numbers, grouped by ITU region (HF / VHF / UHF / SHF).

How to use

1. Browse the bands grouped by region; read the per-class privileges and the mode/segment notes on each band.
2. Check the power summary and the HF data rule for the rules that cut across bands.

What it shows

- Per-band frequency range, license-class privileges, power, and mode segments.
- The corrected power picture: a 1500 W PEP general ceiling, with the real exceptions (30 m = 200 W PEP; 60 m = 100 W ERP (channels) and 9.15 W ERP (segment); 2200 m = 1 W EIRP; 630 m = 5 W EIRP; a Technician on any HF band = 200 W PEP), always under the "minimum necessary power" rule, 97.313(a). The old blanket "200 W" entries were wrong.
- 60 m as 4 channels plus the 5351.5–5366.5 kHz segment at 9.15 W ERP, effective 13 Feb 2026 (the old 5-channel chart is wrong).

- The HF data rule that replaced the obsolete baud column: no symbol-rate limit on HF, a 2.8 kHz maximum-bandwidth cap, effective 8 Jan 2024 (FCC 23-93). No baud value is shown anywhere, because the VHF/UHF baud status is unconfirmed.
- 9 cm (3300–3500 MHz) is omitted as a band in active sunset that does not overlap 5 GHz Wi-Fi.

Field notes

- Data source: FCC Part 97, the FCC frequency-allocation table, ARRL, and NCVEC. The Toolbox renders its own band plan from FCC data rather than reproducing ARRL's copyrighted chart.
- ASCII hyphen-minus throughout for ranges; a privilege absence reads as a sentence ("No Technician privileges"), never a bare dash. Offline, read-only.

Band Names & Wavelengths

The two-worlds translation: amateur band names (160 m, 2 m, 13 cm) to their actual frequency ranges, with the λ = 300 / f rule of thumb.

Why it's here. Hams name bands by wavelength in meters and centimeters; Wi-Fi people think in megahertz and gigahertz. This is the lookup that crosses between them.

How to use

1. Find a band by its wavelength name to read its frequency range and ITU region tag.
2. Use the λ = 300 / f rule to sanity-check the relationship in your head.

What it shows

- The band-name → frequency table from 2200 m (135.7–137.8 kHz) up through 3 cm (10.0–10.5 GHz), each with an LF/MF/HF/VHF/UHF/SHF region tag.
- 60 m shown honestly as "~5.332–5.405 MHz (4 ch + segment)" and 13 cm as the split "2300–2310 / 2390–2450 MHz"; 9 cm flagged sunset/out.

Field notes

- The wavelength is the nominal band name; the range is the actual allocation, so the rounded λ = 300 / f relationship is a memory aid, not the exact band edges. (Wavelength in meters is roughly 300 divided by frequency in megahertz.)
- Offline, read-only. Same data source as the band plan.

Spectrum Band Designations

The ITU decade-band designations (HF, VHF, UHF, SHF) with their frequency and wavelength spans and what each band's propagation implies, plus the aviation airband and military UHF neighbors a Wi-Fi pro lives next to.

Why it's here. Knowing which decade band you are in tells you how the energy travels: ionospheric skip, line-of-sight, or short-range. This sets that context around the Wi-Fi bands.

How to use

1. Read the four ITU decade bands (each is ×10 the previous) and their propagation character.
2. Check the neighbor list for the non-amateur services that sit beside the bands you work.

What it shows

- The four ITU decade bands a Wi-Fi pro lives near (HF 3–30 MHz sky-wave / ionospheric skip; VHF / UHF / SHF), each with frequency, wavelength, and an operational propagation note. MF/LF below and EHF above are noted, not tabled.
- The spectrum neighbors (the aviation airband and military UHF allocations) that border these bands.

Field notes

- Propagation notes are operational summaries (HF "talk around the world," higher bands line-of-sight), not predictions for a specific path.
- Offline, read-only.

Part 15 vs Part 97

Where the amateur 13 cm and 5 cm bands overlap 2.4 and 5 GHz Wi-Fi, and how the unlicensed (Part 15) and amateur (Part 97) rules differ: the rules behind running commodity Wi-Fi hardware as an AREDN mesh node.

Why it's here. The same 802.11 silicon can run under two completely different rule sets. This lays the two side by side so the trade (Part 97's higher power ceiling for Part 15's encryption and commercial freedom) is explicit.

How to use

1. Read the overlap table to see which Wi-Fi channels fall inside an amateur allocation.
2. Read the rule-delta table for how license, power, station ID, encryption, business use, and content differ between Part 15 and Part 97.

What it shows

- The overlapping allocations mapped to the Wi-Fi grid: 13 cm (2390–2450 MHz) covers the lower ~60 MHz of 2.4 GHz (channels 1–6 fully inside, 7 partially); 5 cm (5650–5925 MHz) covers upper U-NII-2C through U-NII-4; 33 cm (902–928 MHz) is full co-channel with the 900 MHz ISM band.
- The rule deltas: Part 97 requires a licensed control operator and a callsign every 10 minutes (including digital/mesh links), prohibits encryption (mesh traffic must be in the clear), and bars business use and broadcast content; Part 15 needs no license, allows WPA2/WPA3 and commercial use. Part 15 2.4 GHz power is 1 W (30 dBm) conducted / 4 W (36 dBm) EIRP at up to 6 dBi; Part 97 runs to the 1500 W PEP ceiling.

- The AREDN example: AREDN and Broadband-Hamnet run commodity 802.11 a/b/g/n hardware on the overlapping ham channels under Part 97 instead of Part 15.

Field notes

- The 3.3–3.5 GHz amateur band (9 cm) is being sunset and does NOT overlap Wi-Fi 5 GHz; do not confuse it with the 5 cm band (5650–5925 MHz), which does.
- Both Part 15 and Part 97 are secondary on these bands and carry the same "do not cause harmful interference" ethic. Offline, read-only.

General License Frequency Chart

A bundled, offline, pinch-zoomable copy of the US amateur General-class HF/VHF/UHF privilege chart: frequency segments, power limits, and the corrected 60 m channels.

Why it's here. You want the General-class privileges as a single printable chart you can zoom on the phone you already have, with no network needed.

How to use

1. Open the card and pinch / double-tap to zoom; the page is fit-to-screen on open.
2. This is a flat print PDF, not an in-app data table, so the content is not screen-reader readable.

Field notes

- Card inner content is not accessible to screen readers (flat rasterized PDF). The card title and "pinch to zoom" gesture are announced.
- Source / basis: Keith's corrected amateur-radio reference, bundled and rendered offline, exactly like the laminated Wi-Fi cards.

Ham Radio General Exam Study Notes

A bundled, offline, pinch-zoomable copy of condensed study notes for the amateur-radio General exam: rules, operating practice, and the RF concepts a Wi-Fi pro already knows.

Why it's here. You want a compact, zoomable General-exam study sheet on the phone, offline, with no network needed.

How to use

1. Open the card and pinch / double-tap to zoom; the page is fit-to-screen on open.
2. This is a flat print PDF, not an in-app data table, so the content is not screen-reader readable.

Field notes

- Card inner content is not accessible to screen readers (flat rasterized PDF). The card title and "pinch to zoom" gesture are announced.

- Source / basis: Keith's amateur-radio study notes, bundled and rendered offline.

Reference Cards (13)

2.4 GHz Channel Allocations

A bundled, offline, pinch-zoomable copy of Keith's published "2.4 GHz channel layout and allocations" laminated reference card.

Why it's here. You want the canonical WLAN Pros 2.4 GHz channel allocation map exactly as printed, viewable and zoomable on the phone you already have, with no network needed.

How to use

1. Open the card and pinch / double-tap to zoom; the page is fit-to-screen on open.
2. This is a flat print PDF, not an in-app data table, so the content is not screen-reader readable.

Field notes

- Card inner content is not accessible to screen readers (flat rasterized PDF). The card title and "pinch to zoom" gesture are announced.
- Source / basis: Keith's own published WLAN Pros laminated reference card, exported to PDF and bundled in the app. It renders offline, with no network needed.

5 GHz Channel Allocations

A bundled, offline, pinch-zoomable copy of Keith's published "5 GHz channel layout and allocations" laminated reference card.

Why it's here. You want the canonical WLAN Pros 5 GHz channel allocation map exactly as printed, viewable and zoomable on the phone you already have, with no network needed.

How to use

1. Open the card and pinch / double-tap to zoom; the page is fit-to-screen on open.
2. This is a flat print PDF, not an in-app data table, so the content is not screen-reader readable.

Field notes

- Card inner content is not accessible to screen readers (flat rasterized PDF). The card title and "pinch to zoom" gesture are announced.
- Source / basis: Keith's own published WLAN Pros laminated reference card, exported to PDF and bundled in the app. It renders offline, with no network needed.

6 GHz Channel Allocations

A bundled, offline, pinch-zoomable copy of Keith's published "6 GHz channel layout and allocations" laminated reference card.

Why it's here. You want the canonical WLAN Pros 6 GHz channel allocation map exactly as printed, viewable and zoomable on the phone you already have, with no network needed.

How to use

1. Open the card and pinch / double-tap to zoom; the page is fit-to-screen on open.
2. This is a flat print PDF, not an in-app data table, so the content is not screen-reader readable.

Field notes

- Card inner content is not accessible to screen readers (flat rasterized PDF). The card title and "pinch to zoom" gesture are announced.
- Source / basis: Keith's own published WLAN Pros laminated reference card, exported to PDF and bundled in the app. It renders offline, with no network needed.

Antenna Fundamentals

A read-along teaching reference for antenna literacy: what an antenna actually does (shapes where the radio's energy goes, it does not add power), azimuth vs elevation, why gain trades against beamwidth, polarization and the wall-clock mistake, downtilt, how to read a radiation-pattern polar plot (main lobe, the -3 dB beamwidth points, side lobes, nulls, front-to-back ratio), and which antenna type (omni, patch, sector, Yagi, dish) fits which space. Seven line diagrams are embedded at the points they teach.

Why it's here. It is the antenna-literacy companion to the directional tools in the toolbox (AP Placement, Downtilt, Point-to-Point, Fresnel). When you are choosing or mounting an antenna and want to reason about coverage shape rather than chase a gain number, this is the read-first reference.

How to use

1. Read top to bottom: it builds from one idea (an antenna is a shaper, not a booster) through azimuth/elevation, orientation, reading a pattern chart, and what to use where.
2. Use the embedded diagrams alongside the prose: the polar-plot anatomy diagram labels every feature (main lobe, -3 dB beamwidth, side lobes, nulls, back lobe, front-to-back) so a manufacturer's chart stops being a mystery.
3. The type table and the closing deployment quick-map are the fast lookups: match the space you are covering to the antenna whose pattern fits it.

Field notes

- This is conceptual teaching copy, not a calculator and not a manufacturer spec sheet. It deliberately gives no formula for gain vs beamwidth: "more gain means a narrower beam" is always directionally

true, but the exact beamwidth for a given gain depends on the specific antenna's design. Read the published beamwidth on the data sheet; do not try to back it out of the gain number.
- The per-type beamwidth figures (patch ~30-120 degrees, sector ~60-120, Yagi ~15-40, dish ~3-25) are RANGES across real products, not single specs. The antenna in your hand lands somewhere inside its band.
- The cross-polarization penalty is given conceptually: 90 degrees is the worst case (in theory total loss; in practice reflections leave some signal), and a wall-mounted omni costs roughly 6 dB. These are rules-of-thumb to reason with, not an exact loss figure.
- The floor-plan coverage shapes are illustrative of how a pattern fills a space, not survey predictions or measured coverage.
- The point-to-point note is one line by design: a link needs real Fresnel-zone clearance around the straight line between antennas, not just a visible path. Use the Fresnel and Point-to-Point tools to size that clearance.

Spectrum Analysis

A read-along teaching module on spectrum analyzers: what a spectrum analyzer sees that a Wi-Fi adapter cannot, how the instrument works, the knobs, the three views, a nine-signature interferer gallery, how to compare captures, the current tool landscape, and how to mitigate interference. Teaching content, not a live tool: the phone does not capture RF.

Why it's here. When users report poor performance but a Wi-Fi scanner shows a clean channel, the missing piece is almost always a non-802.11 energy source only a spectrum analyzer can see. This is the read-first reference for understanding that instrument and reading its output, the spectrum companion to Antenna Fundamentals. It is the single most important honest scope: a phone's Wi-Fi chipset decodes frames, it is not the wideband RF capture hardware spectrum analysis needs, so the module never pretends to be a working analyzer.

How to use

1. Start on the hub: an intro, the honest scope note, and eight numbered topic cards.
2. Read the eight topics in order (Why a spectrum analyzer?, How it works, The knobs, The three views, Fingerprinting interferers, Comparing captures, The tools, Mitigation) or jump to the one you need.
3. On "The knobs," tap through to the Wi-Fi Glossary for term lookups (a one-tap cross-link).

What it shows

- Two instruments contrasted: a Wi-Fi adapter is a protocol decoder that only perceives energy it can read as 802.11; a spectrum analyzer measures raw RF power versus frequency regardless of protocol.
- How it works: swept-tuned versus real-time (RTSA), why real-time matters for bursty and frequency-hopping Wi-Fi interference, the probability-of-intercept figure of merit, and what the frequency transform and its window function do.

- The knobs: span, RBW, VBW, reference level, the dBm amplitude scale, detectors, and max-hold/averaging.
- The three views: live power vs frequency, waterfall/spectrogram (frequency, time, color), and density/duty cycle (how often a frequency is occupied), plus which question each answers.
- A nine-signature interferer gallery, one waterfall shape each, with a fingerprint caption per card: microwave oven, Bluetooth Classic, analog video camera, Bluetooth Low Energy (BLE), analog baby monitor, drone/FPV downlink, ZigBee/802.15.4, analog cordless phone, and continuous wireless bridge.
- Comparing captures (max-hold vs averaging, overlays, before/after with a baseline) and the mitigation ladder, conclusion-first: removing the source is the only complete fix; relocate, shield, plan channels around 1/6/11, and move critical traffic from 2.4 to 5 to 6 GHz to route Wi-Fi around it.

Field notes

- The honest scope note leads the module and repeats on the first topic: this is teaching content, the phone cannot run a spectrum analyzer.
- The nine signature diagrams are illustrative teaching rasters drawn to the canonical signatures (an illustrative trace plus a waterfall and a dBm legend), not captures from your environment. Each card's fingerprint caption is real text below the diagram, so a screen-reader user hears the teaching content even though the raster itself is decorative.
- Tool landscape, current and honest: leaders are the Ekahau Sidekick 2, NetAlly NXT-2000 (the spectrum accessory for the AirCheck G3 Pro), and Oscium Lucid (Wi-Spy Lucid / WiPry Clarity, now under the same MetaGeek family); heritage gear (MetaGeek Wi-Spy DBx, Cisco Cognio) is named as heritage, not a current buy; RF Explorer is the budget entry below survey grade. Pricing is deliberately not quoted.
- 6 GHz being the cleanest band is flagged as a current condition that will erode as adoption grows.

Ham Radio Study Resources

A vetted, offline list of where to study for the amateur-radio exams (hamstudy.org, the ARRL License Manuals, FCC Part 97, and AREDN) each credited, with an "Open website" button and the current exam structure.

Why it's here. The companion to the in-app ham references: when you decide to actually sit an exam, this points you at the strongest current study material instead of a stale chart, with the two currency landmines flagged.

How to use

1. Read the resource cards; tap "Open website" to hand off to the browser (opens in your device's default browser over HTTPS).
2. Read the exam-structure block for the stable "N questions, M to pass" figures.

What it shows

- Four vetted resources: hamstudy.org (the study platform, not its links page), the ARRL License Manuals (the authority reference, cited not reproduced), FCC Part 97 + the frequency-allocation table (the law and the app's data source of truth), and AREDN (the explicit Wi-Fi tie-in: commodity 802.11 hardware run under Part 97).
- The stable exam structure (NCVEC / 97.503): Technician (Element 2) 35 questions / 26 to pass; General (Element 3) 35 / 26; Amateur Extra (Element 4) 50 / 37. No Morse requirement since 23 Feb 2007; the FCC application fee is $35.

Field notes

- The two currency landmines are surfaced as guardrails, never hard-coded around: a new Technician question pool takes effect 1 Jul 2026 (so the pool count is deliberately not memorized, only the stable 35/26 structure is shown), and the 60 m rules changed effective 13 Feb 2026.
- The only network action is a browser hand-off over HTTPS, not an in-app fetch, so no special network exception is needed. A failed hand-off shows an honest inline note with the raw URL, never a silent failure.

Extended Checklist (Non-Advertised Items)

A bundled, offline, pinch-zoomable copy of Keith's published "extended checklist, non-advertised items" laminated reference card.

Why it's here. You want the canonical WLAN Pros extended (non-advertised items) checklist card art exactly as printed, viewable and zoomable on the phone you already have, with no network needed.

How to use

1. Open the card and pinch / double-tap to zoom; the page is fit-to-screen on open.
2. This is a flat print PDF, not an in-app data table, so the content is not screen-reader readable.

Field notes

- Card inner content is not accessible to screen readers (flat rasterized PDF). The card title and "pinch to zoom" gesture are announced.
- Source / basis: Keith's own published WLAN Pros laminated reference card, exported to PDF and bundled in the app. It renders offline, with no network needed.

Extended Wi-Fi Checklist

A bundled, offline, pinch-zoomable copy of Keith's published "extended design checklist items" laminated reference card.

Why it's here. You want the canonical WLAN Pros extended checklist card art exactly as printed, viewable and zoomable on the phone you already have, with no network needed.

How to use

1. Open the card and pinch / double-tap to zoom; the page is fit-to-screen on open.
2. This is a flat print PDF, not an in-app data table, so the content is not screen-reader readable.

Field notes

- Card inner content is not accessible to screen readers (flat rasterized PDF). The card title and "pinch to zoom" gesture are announced.
- Source / basis: Keith's own published WLAN Pros laminated reference card, exported to PDF and bundled in the app. It renders offline, with no network needed.

Modulation and Coding Schemes (MCS Index)

A bundled, offline, pinch-zoomable copy of Keith's published "MCS index, rates, and modulation" laminated reference card.

Why it's here. You want the canonical WLAN Pros MCS index card art exactly as printed, viewable and zoomable on the phone you already have, with no network needed.

How to use

1. Open the card and pinch / double-tap to zoom; the page is fit-to-screen on open.
2. This is a flat print PDF, not an in-app data table, so the content is not screen-reader readable.

Field notes

- This PDF card is separate from the interactive MCS index data-table tool.
- Card inner content is not accessible to screen readers (flat rasterized PDF). The card title and "pinch to zoom" gesture are announced.
- Source / basis: Keith's own published WLAN Pros laminated reference card, exported to PDF and bundled in the app. It renders offline, with no network needed.

Top 20 Wi-Fi Checklist

A bundled, offline, pinch-zoomable copy of Keith's published "Top 20 Wi-Fi design checklist" laminated reference card (PDF card form).

Why it's here. You want the canonical WLAN Pros Top 20 checklist card art exactly as printed, viewable and zoomable on the phone you already have, with no network needed.

How to use

1. Open the card and pinch / double-tap to zoom; the page is fit-to-screen on open.
2. This is a flat print PDF, not an in-app data table, so the content is not screen-reader readable.

Field notes

- This is the PDF card form of the Top 20 checklist, separate from any tappable checklist.

- Card inner content is not accessible to screen readers (flat rasterized PDF). The card title and "pinch to zoom" gesture are announced.
- Source / basis: Keith's own published WLAN Pros laminated reference card, exported to PDF and bundled in the app. It renders offline, with no network needed.

Wi-Fi Connection Checklist

A bundled, offline, pinch-zoomable copy of Keith's published "client connection sequence checklist" laminated reference card.

Why it's here. You want the canonical WLAN Pros connection-sequence checklist card art exactly as printed, viewable and zoomable on the phone you already have, with no network needed.

How to use

1. Open the card and pinch / double-tap to zoom; the page is fit-to-screen on open.
2. This is a flat print PDF, not an in-app data table, so the content is not screen-reader readable.

Field notes

- Card inner content is not accessible to screen readers (flat rasterized PDF). The card title and "pinch to zoom" gesture are announced.
- Source / basis: Keith's own published WLAN Pros laminated reference card, exported to PDF and bundled in the app. It renders offline, with no network needed.

Wireless LAN Troubleshooting Causes

A bundled, offline, pinch-zoomable copy of Keith's published "common causes to check when troubleshooting" laminated reference card.

Why it's here. You want the canonical WLAN Pros troubleshooting-causes card art exactly as printed, viewable and zoomable on the phone you already have, with no network needed.

How to use

1. Open the card and pinch / double-tap to zoom; the page is fit-to-screen on open.
2. This is a flat print PDF, not an in-app data table, so the content is not screen-reader readable.

Field notes

- Ships as a PDF card on purpose: print-layout artwork, distinct from the equivalent in-app data tables.
- Card inner content is not accessible to screen readers (flat rasterized PDF). The card title and "pinch to zoom" gesture are announced.
- Source / basis: Keith's own published WLAN Pros laminated reference card, exported to PDF and bundled in the app. It renders offline, with no network needed.

WLAN Pros Bubble Diagram

A bundled, offline, pinch-zoomable copy of Keith's published "Wi-Fi design decision bubble diagram" laminated reference card.

Why it's here. You want the canonical WLAN Pros bubble-diagram design-decision flow exactly as printed, viewable and zoomable on the phone you already have, with no network needed.

How to use

1. Open the card and pinch / double-tap to zoom; the page is fit-to-screen on open.
2. This is a flat print PDF, not an in-app data table, so the content is not screen-reader readable (the screen names the card and the gesture as the honest accessible affordance).

Field notes

- Ships as a PDF card on purpose: print-layout artwork, distinct from the equivalent in-app data tables.
- Card inner content is not accessible to screen readers (flat rasterized PDF). The card title and "pinch to zoom" gesture are announced.
- This is the only rotated card (a portrait page with a 90-degree rotation flag). The viewer was specifically built so this card paints landscape and undistorted.
- Source / basis: Keith's own published WLAN Pros laminated reference card, exported to PDF and bundled in the app. It renders offline, with no network needed.

Field conveniences

A handful of tools in the kit are not Wi-Fi curriculum. They are the things that turn out to be handy when you are on a roof, in a data closet, or on a call across time zones, so they ship in the box without a training-grade writeup. Reach for them when you need them; there is nothing to study.

- **Phonetic Alphabet** (Encoding): NATO/ICAO spelling words for reading a BSSID or serial number over a noisy phone line, plus the Morse, semaphore, and maritime signal-flag equivalents on the same screen.
- **Morse Code** (Utilities & Generators): encode and decode International Morse (ITU-R M.1677-1), with audio playback.
- **Keyboard Shortcuts** (Encoding): macOS and Windows system and terminal keys, the Mac modifier symbols, and the Greek letters that show up in RF math.
- **Time Zones** (Time & Formats): world UTC offsets, anchor cities, and the US time-zone table, for coordinating work across sites and scheduling calls.
- **Emergency Phrases** (Travel & Field): travel and emergency phrases in English, Spanish, French, Italian, and German, searchable and offline, for the install trip that crosses a border.

Field & Trade Reference (20 tools)

The codes, trades, documents, compliance frameworks, and adjacent radios a WLAN pro meets on a real job and never learned in a Wi-Fi cert. Every entry does one job: it lets you recognize what you are looking at, quote it honestly, and hand the ruling to the right authority. They point you at the AHJ, the licensed electrician, the RCDD, the QSA, the biomed team, or the architect of record. They certify nothing and clear no one. Grouped and ordered the way the app groups them.

Every reference plate in this set is downloadable as a PDF from inside the app, so you can save, share, or AirDrop the full-resolution plate for print. The two entries under Vendor & Hardware are interactive drill-downs (they carry selection state) rather than static plates.

Codes & Safety (6)

Enclosure Ratings

The two enclosure-rating systems a WLAN pro reads on outdoor and industrial spec sheets: the IP code (IEC 60529, international) and NEMA types (NEMA 250, US). It decodes each IP digit, lists the NEMA types that matter for Wi-Fi, and shows the one-way NEMA-to-IP translation.

Why it's here. US spec sheets quote NEMA and international sheets quote IP, so you translate between them to compare gear on the same terms and spec an enclosure, AP, antenna, or PoE injector to the real hazard at the mount point.

How to use

1. Read an IP code left to right: first digit is solids and dust (0 to 6), second digit is water (0 to 9K). Each digit maps to a defined lab test.
2. Use the common-ratings table to place a code (IP66 is dust-tight plus powerful jets, the mainstream outdoor AP rating).
3. For a US spec sheet, read the NEMA type, then use the NEMA-to-IP table as a minimum floor. The reverse (IP to NEMA) is not valid.
4. Use the placement table to pick a rating for where the gear mounts.

Field notes

- X means not tested, not zero. IPX7 is water-rated 7 with the solids digit unrated. Read X as no data, never as fails.
- NEMA to IP is valid only as a minimum. NEMA tests corrosion, icing, gasket aging, and oil, which IP never checks, so a NEMA 4X box exceeds IP66 but an IP66 box is not automatically NEMA 4X.

- The water ladder is not a clean superset past 6. An IP67 (immersion) device is not guaranteed to pass IP66 (jets); gear that must survive both is dual-marked, such as IP66/IP67.
- IP tests use fresh water only. Salt, detergents, and solvents need separate chemical-resistance verification.
- This is a field reference, not code or design guidance. Confirm requirements with the AHJ, the architect of record, and a licensed electrician.
- Data source: IP per IEC 60529; NEMA per NEMA 250.

Hazardous Locations

A recognize-and-defer reference for classified (hazardous) areas: the NEC Class and Division system (US, Article 500) and the IEC Zone system (Article 505/506, ATEX, IECEx). It names what the hazard is (Class), how often it is present (Division or Zone), the recognized protection concepts, and the field read, then stops.

Why it's here. A standard commercial AP is a genuine ignition source in a flammable atmosphere. Mounting one in a classified area is illegal, uninsurable, and dangerous. You need to recognize a refinery, grain elevator, fuel depot, or spray booth before quoting, and route the install to rated gear and the AHJ.

How to use

1. Read Class as what the hazard is made of: Class I (gases and vapors), Class II (dust), Class III (fibers and flyings).
2. Read Division as how often it is present: Div 1 during normal operation, Div 2 only under fault. Div 2 is the far larger wireless market.
3. Map to the IEC Zone system where a facility uses it (Div 1 is roughly Zone 0 plus 1 for gas; Div 2 is roughly Zone 2).
4. Match a recognized protection concept (Ex d, Ex i, Ex p, Ex e/nR) to the Division or Zone, or keep the AP out and remote the antenna in.

Field notes

- Never just mount a commercial AP in a classified area. The default move is to keep the AP in the adjacent general-purpose area and remote the antenna into the zone through a rated penetration.
- "Class I Div 2 rated" buys permission to install that specific listed device, in that specific Division or Zone, wired per its control drawing. It is not a blanket safe-anywhere stamp; a Div 2 device is not approved for Div 1.
- You cannot mix Division and Zone classification in the same installation. Which one is in force is set by the facility area-classification drawing and the AHJ, not the installer.
- This is a field reference, not code or design guidance. Confirm requirements with the AHJ, the architect of record, and a licensed electrician.
- Data source: Class/Division per NEC Article 500; Zone per NEC 505/506, ATEX, and IECEx.

NEC Gotchas

A recognize-and-defer reference for the NEC articles that actually bite a WLAN installer: elevator hoistways (620), plenum air spaces (300.22), the communications-cable rating ladder (800), PoE bundle heat (725.144), antenna and mast grounding (810), firestopping fire-rated assemblies (300.21), and abandoned cable (800.25).

Why it's here. These are the code articles most likely to surface on an everyday install. Recognizing each one on site, then handing it to the AHJ, a licensed electrician, or the equipment listing, keeps you out of trouble. Recognize-and-defer, never how-to-comply.

How to use

1. Use it to recognize a code issue on site: the hoistway you cannot put an AP in, the plenum that dictates your cable jacket, the PoE bundle that can overheat, the antenna that needs grounding, the fire wall you must not breach, and the dead cable you must pull.
2. Read the cable-rating ladder as a jacket fire-rating only (CMP plenum, CMR riser, CM/CMG general, CMX limited); substitution runs downhill only.
3. On the two STOP items (PoE ampacity, firestop assembly), do not eyeball a number or pick an assembly yourself; hand it to the licensed electrician or the listed system.
4. Then defer to the right authority for the actual requirement.

Field notes

- STOP on PoE bundle heat (725.144): the exact ampacity and bundle-count numbers come from the code table and the specific install conditions. Do not eyeball them. Size the bundle with a licensed electrician or the cabling designer against the adopted NEC.
- STOP on firestopping (300.21): the approved firestop is a specific listed assembly matched to the wall type, penetrant, and opening. Never improvise it and never pick the assembly yourself.
- Grounding (810) caveat: nothing survives a direct strike. Bonding and surge protection mitigate nearby strikes and static, not a direct hit. Conductor sizing is an electrician's call.
- This is a field reference, not code or design guidance. Confirm requirements with the AHJ, the architect of record, and a licensed electrician.
- Data source: article numbers per the NEC (NFPA 70); confirm against the locally adopted edition.

Safety Basics

A field-awareness reference for the personal protective equipment a general contractor expects before badging you onto an active site, a short note on ESD (protecting the electronics you install, not the person), and the recognize-and-STOP hazards you must hand off rather than work.

Why it's here. Half of a WLAN pro's installs happen on a ladder or lift, on an active construction site, around other trades. Knowing the PPE ratings a GC will ask for keeps you from being turned away at the gate, cheap ESD gear prevents latent failures in the gear you install, and the STOP flags keep a routine cable pull from becoming a health or legal incident. Awareness and lookup, not compliance instruction.

How to use

1. Use the PPE ladder to recognize the four common baseline items and the standard each is rated to (hard hat Z89.1, safety-toe F2413, hi-vis ANSI 107, eye protection Z87.1).
2. Read the ESD note as gear protection: a grounded wrist strap, mat, and static bags under ANSI/ESD S20.20 guard against latent damage to APs, optics, and boards.
3. On the four recognize-and-STOP hazards (asbestos or lead, arc flash and energized work, confined spaces, seismic bracing), stop and hand it off. Never run any of them as a procedure yourself.
4. Confirm the site's specific PPE policy with the employer and GC, who own the assessment.

Field notes

- The four PPE items are the common "let me on the site" baseline. On a finished office install street clothes are usually fine; on active construction expect the GC to require all four to badge on.
- ESD risk is highest in data centers and any time you handle bare optics or boards. For a sealed AP coming out of a static bag, mounted and cabled, the risk is lower but not zero.
- The recognize-and-STOP items are named-and-stopped on purpose: the app names the hazard and tells you to hand it off, and deliberately gives no procedure for working any of them.
- This is a field reference, not code or design guidance. Confirm requirements with the AHJ, the architect of record, and a licensed electrician.
- Data source: standards designators ANSI/ISEA Z89.1, ASTM F2413, ANSI/ISEA 107, ANSI Z87.1, ANSI/ESD S20.20, NFPA 70E; confirm the site's PPE policy with the employer and GC.

Site Access

A "Know Before You Go" pre-mobilization checklist: eight site types (aerial and man-lifts, rail, hospitals, maritime, warehouse and distribution, schools, data centers, correctional) and the credential, screening, orientation, or escort that can gate you from reaching the work before it even starts.

Why it's here. On many sites you cannot reach the work area without a specific credential, background check, orientation, or escort. That is a quoting and scheduling factor, not just a safety one. Rail screening, a hospital ICRA permit, or lift-operator proof can each add days or weeks between winning a job and touching a cable.

How to use

1. Before you quote, scan the checklist for the site type you are bidding and read what may gate you.
2. Use the "ask about" column as the questions to put to the GC, site owner, or authority: screening programs, orientations, escorts, flagman, PFD and TWIC, ICRA and ILSM, tool control.
3. Budget the lead time (rail screening and hospital ICRA in particular run long) into the schedule you promise.
4. Confirm each requirement with the site, general contractor, and authority before you mobilize.

Field notes

- The pattern across every item: the requirement is set by someone other than you, must be satisfied before work starts, and carries real lead time and cost.
- Maritime and over-water work adds PFD and drowning-hazard rules plus heavy salt corrosion, so spec NEMA 4X or high-IP with explicit corrosion resistance.
- There is no OSHA-issued "lift license"; the employer trains and certifies lift operators, and GCs demand documented proof.
- This is a field reference, not code or design guidance. The site, general contractor, and authority set every requirement; the Toolbox certifies nothing and clears no one.

Credentials & Licenses

The portable IDs and licenses a WLAN pro carries from job to job, and the mobilization landmine in them: why you almost never need an FCC operator license (even for licensed microwave backhaul), and the federal and background-check credentials (TWIC, CAC, DBIDS, SIDA, HAZWOPER-40, background checks) that gate restricted sites with weeks-to-months lead time.

Why it's here. The credential you do not already hold is the schedule you cannot keep. A TWIC, a base credential, or a SIDA badge can each add weeks between the award and touching a cable, so scope the credential before you quote, not after you win. Companion to the Site Access entry.

How to use

1. Settle the license question first: unlicensed Wi-Fi (Part 15) and licensed point-to-point microwave (Part 101) both require no FCC operator license (GROL).
2. For a restricted site, read the credential table for the ID, issuing authority, lead time, and validity you will need.
3. Bucket the lead time: fast and you control it (GROL exam, school check), weeks (SIDA, TWIC, DBIDS), or weeks-to-months (CAC).
4. Confirm what your specific job requires with the issuing authority before you quote and before you mobilize.

Field notes

- You would only need a GROL if the work crossed into servicing aviation or marine radios, which is outside a normal WLAN scope of work.
- TWIC gets you the maritime jobs; TSA tells applicants to enroll at least 60 days out and warns processing can exceed 45 days.
- CAC versus DBIDS is the military-base question: the CAC needs a sponsor and an investigation that can run many months, while DBIDS is the lighter get-on-base path.
- Reference only. The issuing authority sets and grants every credential; the Toolbox certifies nothing and clears no one.
- Data source: FCC license scope per Parts 15, 97, and 101; credential detail per TSA, USCG, DoD, and OSHA 29 CFR 1910.120(e).

AEC & Documentation (4)

Plan-Set Literacy

A reference for reading an architectural drawing set: how a sheet number is built (discipline letter, sheet-type digit, sequence), what each discipline designator and sheet-type digit means, why the Reflected Ceiling Plan (RCP) is the sheet for AP placement, the rest of a plan set worth knowing, and US drawing scales.

Why it's here. A WLAN designer who can pull the right sheet, read the RCP, and speak in sheet numbers looks like a peer to the other trades instead of a junior. The RCP is where a coverage design meets the physical ceiling, and catching an AP-versus-diffuser conflict on paper is far cheaper than catching it at rough-in.

How to use

1. Read a sheet number left to right: discipline letter, sheet-type digit, sequence. A-201 is Architectural, an elevation, sheet 01.
2. Use the discipline table to find the sheet owner (A architectural, E electrical, T telecommunications; the WLAN and structured cabling usually live on the T sheets).
3. Use the sheet-type digit to find the drawing kind (1 plans, 2 elevations, 6 schedules), so A-1xx is an architectural plan and E-6xx an electrical schedule.
4. For AP placement, open the RCP: overlay your AP locations and resolve conflicts with diffusers, troffers, sprinklers, and soffits before rough-in. Never place APs on the floor plan alone.

Field notes

- The RCP is drawn as if a mirror on the floor reflects the ceiling upward, so it reads in the same left-right orientation as the floor plan; it carries the ceiling grid, mounting heights, and every ceiling-mounted element.
- Always build to the latest revision: the revision cloud marks what changed and the delta triangle carries the revision number.
- US architectural sheets use fractional-inch scales (1/8" = 1'-0" is 1:96); to get the ratio, invert the fraction and multiply by 12.
- Reference only. Confirm drawing conventions, revisions, and responsibility with the architect of record and your contract for the specific project.
- Data source: conventions per the US National CAD Standard; sheet-numbering practice varies on smaller jobs.

CAD & BIM Formats

What the building files an architect hands you actually are: the format decode table (DWG, DXF, DGN, IFC, RVT, NWD/NWC, COBie), the Level of Development (LOD 100 to 500) ladder for how much of a model to trust, and how a building file flows into a Wi-Fi design tool (Ekahau, Hamina, iBwave).

Why it's here. You receive these files on every design job. The difference between a clean import and a wasted afternoon is knowing what the format is, what LOD you were handed, which CAD layers to ask for, and that scale calibration is the single step that makes or breaks the whole design. The Toolbox explains these formats; it does not open, render, or convert them.

How to use

1. Use the decode table to recognize a format on sight: DWG and DXF are Autodesk CAD, IFC is the open BIM standard, RVT is a Revit model, NWD/NWC are Navisworks coordination files, COBie is asset data.
2. Read the LOD you were handed before trusting geometry: LOD 100 to 200 is a massing study (do not derive wall attenuation from it), LOD 300 and up is dimensionally trustworthy.
3. Follow the import flow: import the CAD or PDF, calibrate the scale on a known distance (get this wrong and every downstream distance is wrong), then assign wall materials and run the prediction.
4. Ask the architect for a clean layer set (walls, doors, structure; strip furniture, dimensions, title blocks) instead of an unusable 80-layer dump.

Field notes

- Scale calibration is the step that matters most: draw a line over a known length and enter the real measurement. An error here corrupts every distance, coverage prediction, and attenuation value downstream.
- LOI (Level of Information) is the data-completeness sibling of LOD: LOD is geometry, LOI is data.
- The boundary: the Toolbox explains DWG, IFC, RVT, and the rest and how they reach the design tools. It is not a CAD or BIM viewer, converter, or editor.
- Reference only. Confirm file handling, model reliability, and responsibility with the architect of record and your design-tool documentation.
- Data source: format and LOD conventions per Autodesk, buildingSMART, and the AIA/AGC LOD framework.

Structured Cabling

The TIA and BICSI structured-cabling standards that decide where an AP can go and what feeds it: the TIA-568 family, the 90 m permanent link plus 10 m of cords for a 100 m channel, the cable-category ladder (Cat 5e through Cat 8), and the MDF-IDF topology that ties AP locations to telecom rooms.

Why it's here. The 90 m rule and the Cat 6A bar shape real WLAN decisions: they set the outer edge of where you can place an AP and what uplink it can carry, which quietly drives IDF placement, switch selection, and the whole coverage plan. The physical grounding-as-safety side lives in the NEC codes reference; this is the TIA and BICSI infrastructure side.

How to use

1. Use the TIA family to know who owns what: 568 is cabling, 569 pathways and spaces, 606 labeling and administration, 607 telecom bonding and grounding.

2. Apply the 90 plus 10 m channel rule as an AP-cable-run reality check: an AP more than about 90 m of cable-path from the IDF needs an intermediate closet, a different topology, or fiber.
3. Read the cable-category table for the multi-gig bar: Cat 6A carries 10 Gbps to 100 m and is the practical minimum for Wi-Fi 6, 6E, and 7 APs with multi-gig uplinks.
4. Coordinate with the RCDD (BICSI's cabling-design credential) whose 90 m constraint ties AP locations to IDF locations.

Field notes

- The 90 m permanent link is solid horizontal cable from the telecom room to the outlet; add up to 10 m of stranded patch and equipment cords for a 100 m channel maximum. Exceed it and you are outside TIA.
- T568A and T568B are the two pin-out standards; a jack is wired to one, so be consistent end to end.
- AP count and placement drive IDF count and PoE-switch port budgeting, so the WLAN design and the cabling design have to talk to each other.
- Reference only. Confirm cabling design and standards currency with the RCDD, the architect of record, and your contract for the specific project.
- Data source: standards per ANSI/TIA-568/569/606/607 and BICSI; confirm the current edition.

AEC Process & Glossary

The architecture-engineering-construction workflow you work inside and the shorthand the other trades use: the AIA design phases (Programming, SD, DD, CD, Bidding, CA) and when Wi-Fi should engage, plus the AEC glossary (RFI, submittal, ASI, AHJ, GC, MEP, OAC, AOR/EOR, and more) that trips WLAN pros up. US convention (AIA).

Why it's here. Speak the process fluently and you look like a peer in the room instead of the network person who wandered in. The RFI, the submittal, the punch list, and the OAC are the levers that get your coverage design built the way you drew it. Engaging at Schematic Design is the difference between designing Wi-Fi in and retrofitting it later.

How to use

1. Use the phase table to engage at the right moment: establish RF requirements and reserve IDF and ceiling access at SD, coordinate AP locations against the RCP at DD, finalize plans and telecom sheets at CD.
2. Raise an RFI the moment the Reflected Ceiling Plan and your AP plan disagree, so the conflict is resolved on paper before rough-in.
3. Get your APs, mounts, and cable approved as a submittal (the contractor's proof that what is installed matches the spec).
4. Treat anything about contractual responsibility as "confirm with the architect of record and your contract," never as a ruling.

Field notes

- The AHJ (Authority Having Jurisdiction) is the building official, fire marshal, or inspector who interprets and enforces code locally; the AHJ's word governs.
- An ASI is a minor clarification with no cost or time impact; a Change Order is a formal, signed change to scope, cost, or schedule (what an ASI is not).
- The OAC is the recurring Owner-Architect-Contractor coordination meeting; if Wi-Fi matters, someone has to carry it into the OAC.
- Reference only. Confirm contractual responsibility, code compliance, and phase deliverables with the architect of record, the AHJ, and your contract for the specific project.
- Data source: phase and contract vocabulary per US AIA convention.

Compliance & Governance (2)

Cloud Tool Trust

How to read the security badges on a cloud Wi-Fi tool before you upload a client's floor plan, AP inventory, or survey data: ISO/IEC 27001 (a certified management system), SOC 2 (an attestation report, not a certificate), GDPR (a law you conform to), the five Trust Services Criteria, and the adjacent badges (ISO 27017/27018, FedRAMP, CSA STAR).

Why it's here. A compliance badge is a claim about a defined scope and a time window, not a guarantee about your data. You are the one uploading a client's network into someone else's cloud, so the badge tells you which questions to ask, not that the data is safe.

How to use

1. Ask whether it is a certificate or an attestation: ISO 27001 is a certificate; SOC 2 is a report plus an opinion, so "SOC 2 certificate" is a wording tell.
2. Read the scope: does it cover the actual product you will use, or the vendor's corporate IT?
3. For SOC 2, confirm Type 2 (operating effectiveness over a period) over Type 1, and that Confidentiality (and Privacy, if personal data) is in scope.
4. Ask where the data lives and under whose law: residency is not sovereignty, and EU hosting does not by itself solve GDPR.

Field notes

- Third-party-audited claims (ISO 27001 certificate, SOC 2 Type 2, CSA STAR Level 2, FedRAMP) outrank self-asserted ones, and even an audited claim only covers the scope, criteria, and time window printed on it.
- A vendor unwilling to share a SOC 2 report under NDA is itself a flag; read the opinion, scope, period, and the complementary controls you are responsible for.
- Some data, a client's full network design, is sensitive enough that the right home is your own device, not a cloud you never vetted.

- Reference only. Compliance status, scope, and sufficiency are determined by the client's security or compliance officer and a qualified auditor (a CPA firm or counsel), not by this tool.
- Data source: frameworks per ISO/IEC 27001:2022, the AICPA SOC 2 framework, and the EU GDPR.

Network in Scope

The regulatory frameworks that reach the WLAN itself: PCI DSS (cardholder data), HIPAA (electronic health information), SOX (public-company financial systems), and GDPR (EU personal data). Each entry names the trigger, what the framework generally asks of the network, and the owner to route the specifics to.

Why it's here. When the Wi-Fi carries cardholder data, ePHI, or sits inside a public company's financial systems, the design inherits requirements long before anyone audits it. Recognizing the scope lets you quote honestly and design toward the requirement instead of retrofitting after the auditor shows up.

How to use

1. Recognize the trigger: card-present retail and hospitality (PCI), covered entities and business associates (HIPAA), US public companies (SOX), EU personal data on the network (GDPR).
2. Read the general asks: segmentation and strong crypto for PCI, the section 164.312 technical safeguards for HIPAA, access and change controls for SOX.
3. Route the specifics to the owner: the QSA for PCI, the privacy or security officer for HIPAA and GDPR, internal audit and the external auditor for SOX.
4. Never tell a client the design "meets PCI" or "is HIPAA compliant"; you recognize the framework, and the assessor rules on it.

Field notes

- PCI DSS v4.0.1 makes WPA2-PSK inadequate for the Cardholder Data Environment; WEP and WPA/TKIP are prohibited, and quarterly rogue-AP scanning is required even at a site with no Wi-Fi at all.
- HIPAA encryption is currently "addressable," meaning you implement it or document a defensible reason not to; a January 2025 proposed rule would make it mandatory.
- SOX is the narrowest fit: the network shows up as the access-and-change-control substrate under the financial systems, which is why an AP config change can suddenly need a ticket and an approver.
- Reference only. Whether a network is in scope, and whether a design is sufficient, is determined by the client's compliance officer and a qualified auditor, not by this tool.
- Data source: framework requirements per PCI DSS v4.0.1, HIPAA 45 CFR 164.312, SOX Section 404, and the EU GDPR.

Wireless Landscape (1)

Adjacent Radio Systems

The non-Wi-Fi radios a WLAN pro coexists with, designs around, and gets asked about: which five (BLE, Bluetooth Classic, Zigbee, Thread, ANT+) contend for your 2.4 GHz airtime, and which (LoRaWAN, Z-Wave, UWB, NB-IoT, LTE-M, CBRS and private 5G, Wi-Fi HaLow) run coexistence-clean in sub-GHz or licensed spectrum.

Why it's here. The client increasingly hands you the whole smart-building radio stack, not just the Wi-Fi. Knowing what shares the 2.4 GHz air, what runs clean in sub-GHz, and when to talk a client out of a private-cellular pitch that Wi-Fi already covers is both authority and self-defense in a spectrum sweep.

How to use

1. Scan the 2.4 GHz contenders: BLE, Bluetooth Classic, Zigbee, Thread, and ANT+ subtract real airtime at scale, so coordinate a Zigbee lighting mesh with your 1/6/11 plan.
2. Use the coexistence table to route congested-band IoT onto a coexistence-clean radio (LoRaWAN, Z-Wave, cellular IoT) rather than fighting for airtime.
3. Carry the three corrections: Matter is an application layer that can ride Wi-Fi, 802.15.4 is not 2.4 GHz only, and CBRS and private 5G never touch your air.
4. Use the "which radio when" picker to answer the "should this be Wi-Fi or something else" question honestly.

Field notes

- Read every range, rate, and battery figure as a real-world envelope, not a hard spec; they move with power, antenna, spreading factor, channel width, and environment.
- CBRS lives at 3.55 to 3.70 GHz and does not overlap any Wi-Fi band; private cellular competes for the job and the budget, never for your air.
- A Zigbee mesh and a wall of BLE beacons show up in a 2.4 GHz sweep as non-Wi-Fi energy; naming what you see is the difference between a diagnosis and a guess.
- Reference only. Confirm current standards editions and regional band allocations for the specific deployment.
- Data source: bands and topologies per the relevant IEEE 802.15.4, Bluetooth SIG, LoRa Alliance, and 3GPP specifications.

Verticals (4)

Verticals Index

A plain-language index of the industries you quote and what each one tends to trigger: manufacturing, oil and gas, warehouse, healthcare, hospitality and stadiums, education, retail, data centers, maritime, and correctional or government. It maps the vertical you name to the other reference entries you should read before you quote it.

Why it's here. Nobody standing in a building thinks "this is NAICS 622110"; you think "it's a hospital." Classification codes are back-office filing labels, not design inputs. The vertical you name in the first phone call tells you which reference entry to open before you quote.

How to use

1. Find the vertical you are bidding in the map and read what it tends to trigger.
2. Open the "read first" entry (Hazardous Locations, Enclosure Ratings, Healthcare Wi-Fi, Site Access, Data Centers, or Telecom Spaces) before you quote.
3. For retail, recognize PCI DSS scope when Wi-Fi touches cardholder data, then defer the ruling to the client's QSA.
4. For high-density venues, remember the design axis flips from coverage to capacity, driven by seat count and peak concurrent users.

Field notes

- Miss the hazloc on an oil site, the ICRA gate on a hospital, or the PCI scope in retail, and the timeline and price you promised were wrong before you started.
- Treat a classification code (NAICS, SIC, GICS, ISIC, NACE) as a label a client hands you, never a design input.
- Use the index the other direction too: pick the industry, see the cluster, read the entry that owns the detail.
- This is a field reference, not code, design, or compliance guidance; confirm every requirement with the client, the AHJ, and the relevant assessor.
- Data source: this entry is the index that points at the other Field & Trade Reference entries.

Healthcare Wi-Fi

Why a hospital is the one building where you cannot design Wi-Fi like an office: the protected WMTS telemetry band, medical-device EMC (IEC 60601-1-2), the voice-and-RTLS roaming grades, shielded rooms, and the four authorities (HIPAA, FDA, FCC, The Joint Commission) plus the in-house biomedical engineering handoff.

Why it's here. The air is shared with life-critical, EMC-regulated devices and a protected telemetry band, a dropped roam can mean a missed alarm, and the "just cover it like an office" instinct is exactly the mistake that gets telemetry stepped on and alarms missed.

How to use

1. Coordinate with clinical and biomedical engineering before you touch a hospital RF environment; that is the single most important handoff.
2. Do not assume all patient monitoring is on Wi-Fi: WMTS is a separate, licensed system on 608 to 614, 1395 to 1400, and 1427 to 1432 MHz, coordinated by a WMTS coordinator.
3. Design to voice-and-RTLS grade, not data grade: continuous facility-wide roaming, overlapping cells, and no dead zones, and design around shielded rooms and the MRI Faraday cage.
4. Recognize each of the four authorities, then defer the rulings to the people who own them.

Field notes

- Medical devices are tested to tolerate a defined RF environment under IEC 60601-1-2, but tested to a limit is not the same as immune to anything; a dense, high-power deployment can push local field strength toward what a nearby monitor was qualified for.
- Coverage grade drives AP count more than floor area does; a hospital quoted at data-grade density fails when it has to carry voice handsets and location services.
- Segmentation is a performance control here, not only a security one: guest, clinical, device, and RTLS traffic get separated for airtime protection as much as for HIPAA.
- This is a field reference, not clinical, code, or compliance guidance; confirm every requirement with biomed, the compliance and security officers, the WMTS coordinator, and the AHJ.
- Data source: standards per the FCC WMTS allocation, IEC 60601-1-2, HIPAA, FDA guidance, and The Joint Commission.

Data Centers & Wi-Fi

The read a WLAN pro needs before quoting Wi-Fi in or around a data center: why production Wi-Fi on the floor is usually minimal, why the room is RF-hostile by construction, and how the two resilience frameworks (ANSI/TIA-942 Rated-1 to Rated-4 and the Uptime Institute Tier I to Tier IV) differ.

Why it's here. Walk into a data center expecting office-style coverage and you get two things wrong at once: you fight an RF environment engineered to reflect and contain, and you quote a schedule that ignores the badging and change-control gate.

How to use

1. Clarify the actual use case first: it is rarely production; it is usually out-of-band management, staff mobility, or guest.
2. Survey with the containment in place, because empty-room predictions lie once the racks and aisle barriers are up.

3. Keep the frameworks straight: TIA-942 says Rated, Uptime says Tier, and mixing them ("a Tier 3 TIA rating") is a telling error.
4. Budget the badging and change-control lead time up front, and defer the facility rating to the operator and its design engineer.

Field notes

- Dense metal racks, hot-aisle and cold-aisle containment, and overhead trays chop the space into sealed RF pockets, so the design problem is coverage, not capacity.
- The telecom rooms themselves are small and metal-dense, so APs often go outside or at the doorway rather than inside the rack cage.
- Access is a credentialing exercise: expect badging, mantraps, escort, no-photography rules, NDAs, and change-control windows, the same "know before you go" pattern as any high-security site.
- This is a field reference, not design or facility-rating guidance; confirm resilience ratings, access, and change-control rules with the operator and its design engineer.
- Data source: framework detail per ANSI/TIA-942 and the Uptime Institute Tier Standard.

Telecom Spaces

A decoder for the telecom room names that get used interchangeably and are not interchangeable: Entrance Facility, Equipment Room, Telecommunications Room (the current TIA-569 term), MDF and IDF (legacy distribution-frame terms the field still uses), and "data closet" (slang), plus the hierarchical-star topology that ties them together.

Why it's here. When the electrician says "MDF," the architect's drawing says "TR," and the client says "the data closet," knowing they usually mean the same room keeps you from designing a phantom extra space or missing a real one. The IDF and TR locations are where your APs get their uplink and power.

How to use

1. Use the decode table to map a room name to what it actually is and whether it is a standard, legacy, or slang term.
2. Recognize that MDF and IDF are legacy frame terms and TR is the current TIA-569 word; IDF and TR describe the same space.
3. Read the topology as a hierarchical star: EF in, MDF or Equipment Room at the center, backbone out to the IDF or TR per floor, horizontal out to the AP.
4. On an international job, recognize the ISO/IEC 11801 "distributor" terms (Campus, Building, Floor Distributor) for the same hierarchy.

Field notes

- The cable mechanics (the 90 meter horizontal link, categories, PoE budgeting) live in the Structured Cabling reference and are not repeated here.
- One TR per floor at minimum, more for large floors, because horizontal runs are distance-limited.
- "Data closet" is not a standard term at all; it is an informal catch-all for any TR or IDF.

- Reference only. Confirm the space names, standards currency, and cabling design with the architect of record, the RCDD, and your contract.
- Data source: space vocabulary per ANSI/TIA-569-E, alongside ANSI/TIA-568 and ISO/IEC 11801.

Vendor & Hardware (2)

LED Decoder

An interactive cross-vendor decoder for an access point's status LED. Pick the vendor, pick the model line when the vendor forks (Cisco Catalyst vs Meraki, Aruba Campus vs Instant On, Extreme IQ Engine vs WiNG), then read that line's own color-and-blink state table: booting, needs adoption, healthy, upgrading, fault, locate, and factory reset. Each state carries a literal colored indicator, a green, amber, red, blue, white, purple, or magenta dot, solid or gently flashing, beside the verbatim signal text. A master cross-vendor comparison chart rides at the top of the vendor picker with the whole color matrix on one plate.

Why it's here. The color on the front of an AP is the fastest read you get before you open a laptop, but the same color means opposite things across vendors, so a flat "green equals healthy" legend is actively wrong. Solid green is healthy-but-no-clients on Meraki and healthy-with-clients on Ruckus; solid white is needs-adoption on UniFi and healthy-on-cloud on Extreme. Resolving the model line first is what keeps the read honest.

How to use

1. Scan the master cross-vendor comparison chart at the top of the picker to see the whole color matrix on one plate; tap it to pinch-zoom, or download it as a PDF.
2. Pick the vendor (enterprise lines first, consumer mesh kept separate).
3. If the vendor forks by management line, pick the line; a single-line vendor jumps straight to its table.
4. Read the state row: the colored indicator (color plus solid or flashing), the verbatim signal text, and what it means in the field.
5. Treat every color as a heuristic and confirm against the exact model's install or getting-started guide.

Field notes

- The colored dot is never the only signal: the color is always named in words beside it, and the verbatim signal text stays the authority for any nuance a dot cannot carry (sequences, alternating patterns, blink-count error codes).
- Undocumented states are marked honestly. A state with no reachable vendor doc renders "Not documented by the vendor, confirm on a lab AP" with a neutral "?" indicator and no invented color. There are exactly six such states.
- Some vendors ship no distinct signal by design (a Meraki factory reset reads as an ordinary reboot); the "reads as X" note is the answer, not a gap.
- MikroTik ships as an honest note, not a table: RouterOS LEDs are user-configurable, so there is no standardized status-LED scheme to decode.

- LED behavior can change with a firmware or dashboard release on cloud-managed lines. Reference only; confirm on the vendor's own documentation.
- Data source: per-line vendor docs cited on each table, including the Cisco Catalyst Getting Started Guides, the Meraki MR46 Installation Guide, the Juniper Mist LED documentation, the Aruba AP-635 and AP22 installation guides, the Extreme Networks documentation portal, help.ui.com, and Ruckus KB 000001629.

Vendor Model Decode

A per-vendor reference for reading an enterprise AP model number. Pick the vendor, then read that vendor's own model-number scheme: what each segment of the SKU encodes (product series, Wi-Fi generation, radio and stream tier, antenna type, regulatory domain), plus a worked example that decodes one real SKU end to end. Covers Cisco (Catalyst/Meraki/CW), HPE Aruba, Ubiquiti UniFi, Ruckus, and Extreme.

Why it's here. Model numbers are position- and suffix-encoded and stable within a vendor generation, so a clean decode is possible offline. But every vendor encodes differently, so this is a per-vendor decoder, never a shared letter dictionary: the letter E alone means a regulatory domain on Cisco, an external antenna on Aruba, and a product tier on UniFi. One universal letter map would turn all three into one wrong answer.

How to use

1. Pick the vendor.
2. Read the token table left to right: each segment and what it encodes.
3. Read the worked example to see one real SKU decoded segment by segment.
4. Check the confidence-and-caveats note for where a segment needs a per-model datasheet lookup instead of a digit rule.

Field notes

- This is deliberately not a "paste a model number, auto-decode" input. Several vendors (Extreme especially) do not digit-encode Wi-Fi generation, streams, or antenna, so an auto-decoder would fabricate precision the SKU does not carry.
- Aruba's even/odd last-digit antenna rule (even = external, odd = internal) is confirmed back to the Wi-Fi 5 300 series; the 200 series is reported to follow it but is unverified, so decode pre-300-series from a per-model lookup.
- Extreme is Medium confidence: only the first digit (tier) decodes; Wi-Fi generation, stream count, and antenna come from the datasheet.
- Juniper Mist, Fortinet, Cambium, and Omada are flagged for a later pass and are not decoded here; when built, each gets its own module rather than another vendor's rules stretched onto it.
- Reference only. A decode is a heuristic; confirm against the exact model's datasheet or ordering guide before you spec, order, or troubleshoot on it.

- Data source: per-vendor field reference compiled by Keith Parsons / WLAN Pros; sources include the Cisco Catalyst 9130AX datasheet and Getting Started Guide, the Aruba 310 Series datasheet, UniFi Tech Specs, the Ruckus product guide, and Extreme Networks product pages.

Calculators (1)

Architectural Scale

Converts a named drawing scale to its dimensionless ratio and back, and converts a distance measured on a drawing to its real-world length or the reverse. Covers US architectural fractional-inch scales, engineer's decimal scales, and common metric scales. This is a calculator, not a reference page.

Why it's here. WLAN pros work inside plan sets and scaled PDFs they were never taught to read. Every coverage prediction, wall distance, and mounting height is wrong if the scale is wrong. This is the same calibration step Ekahau, Hamina, and iBwave ask for on import.

How to use

1. Pick the scale family (Architectural, Engineering, or Metric).
2. Pick the specific scale, for example 1/4" = 1'-0". Read its ratio (1:48).
3. In Measure, choose a direction: Drawn to Real, or Real to Drawn.
4. Set the drawing units (in / mm / cm) and real-world units (ft / m).
5. Enter your measurement and read the converted length, with a friendly feet-inches or fractional-inch form beneath it.

How it works. ratio = real / drawn. Imperial: ratio = 12 / inches-per-foot (1/8" = 1'-0" gives 12 / 0.125 = 96, so 1:96). Engineer's: feet-per-inch times 12 (1" = 20' gives 240, so 1:240). Metric scales are already ratios. Drawn to real multiplies the measurement by the ratio; real to drawn divides. The ratio is dimensionless, so the two ends can carry different units.

Example. 3.5" measured on a 1/8" = 1'-0" (1:96) sheet is 3.5 x 96 = 336 inches = 28 ft. In reverse, a 45 ft hallway at 1/4" = 1'-0" (1:48) draws at 45 / 48 = 0.9375 ft = 11-1/4 inches.

Field notes

- The result is only as good as the scale. If a PDF has no embedded scale, confirm which scale it is by measuring a known dimension, a 3'-0" door or a 2x4 ft ceiling tile.
- Printed and PDF plans are often not to true scale (fit-to-page printing, cropped sheets). Trust a dimensioned callout on the sheet over any measurement.
- US-primary: architectural scales use fractional inches, engineer's scales use decimal feet per inch. Metric ratios (1:50, 1:100) are provided for ISO drawing sets.
- Data source: pure on-device math, no network.