Changelog

All notable changes to this project will be documented in this file.

The format is based on Keep a Changelog, and this project adheres to Semantic Versioning.

Unreleased

1.9.4 - 2024-09-09

Added

- Support UDP dialing with gVisor. (#1181)

Changed

- Make some Nebula state programmatically available via control object. (#1188)
- Switch internal representation of IPs to netip, to prepare for IPv6 support in the overlay. (#1173)
- Minor build and cleanup changes. (#1171, #1164, #1162)
- Various dependency updates. (#1195, #1190, #1174, #1168, #1167, #1161, #1147, #1146)

Fixed

- Fix a bug on big endian hosts, like mips. (#1194)
- Fix a rare panic if a local index collision happens. (#1191)
- Fix integer wraparound in the calculation of handshake timeouts on 32-bit targets. (#1185)

1.9.3 - 2024-06-06

Fixed

- Initialize messageCounter to 2 instead of verifying later. (#1156)

1.9.2 - 2024-06-03

Fixed

- Ensure messageCounter is set before handshake is complete. (#1154)

1.9.1 - 2024-05-29

Fixed

- Fixed a potential deadlock in GetOrHandshake. (#1151)

1.9.0 - 2024-05-07

Deprecated

- This release adds a new setting `default_local_cidr_any` that defaults to true to match previous behavior, but will default to false in the next release (1.10). When set to false, `local_cidr` is matched correctly for firewall rules on hosts acting as unsafe routers, and should be set for any firewall rules you want to allow unsafe route hosts to access. See the issue and example config for more details. (#1071, #1099)

Added

- Nebula now has an official Docker image `nebulaoss/nebula` that is distroless and contains just the `nebula` and `nebula-cert` binaries. You can find it here: https://hub.docker.com/r/nebulaoss/nebula (#1037)
- Experimental binaries for `loong64` are now provided. (#1003)
- Added example service script for OpenRC. (#711)
- The SSH daemon now supports inlined host keys. (#1054)
- The SSH daemon now supports certificates with `sshd.trusted_cas`. (#1098)

Changed

- Config setting `tun.unsafe_routes` is now reloadable. (#1083)
- Small documentation and internal improvements. (#1065, #1067, #1069, #1108, #1109, #1111, #1135)
- Various dependency updates. (#1139, #1138, #1134, #1133, #1126, #1123, #1110, #1094, #1092, #1087, #1086, #1085, #1072, #1063, #1059, #1055, #1053, #1047, #1046, #1034, #1022)

Removed

- Support for the deprecated `local_range` option has been removed. Please change to `preferred_ranges` (which is also now reloadable). (#1043)
- We are now building with go1.22, which means that for Windows you need at least Windows 10 or Windows Server 2016. This is because support for earlier versions was removed in Go 1.21. See https://go.dev/doc/go1.21#windows (#981)
- Removed vagrant example, as it was unmaintained. (#1129)
- Removed Fedora and Arch nebula.service files, as they are maintained in the upstream repos. (#1128, #1132)
- Remove the TCP round trip tracking metrics, as they never had correct data and were an experiment to begin with. (#1114)

Fixed

- Fixed a potential deadlock introduced in 1.8.1. (#1112)
- Fixed support for Linux when IPv6 has been disabled at the OS level. (#787)
- DNS will return NXDOMAIN now when there are no results. (#845)
- Allow `::` in `lighthouse.dns.host`. (#1115)
- Capitalization of `NotAfter` fixed in DNS TXT response. (#1127)
- Don't log invalid certificates. It is untrusted data and can cause a large volume of logs. (#1116)

1.8.2 - 2024-01-08

Fixed

- Fix multiple routines when listen.port is zero. This was a regression introduced in v1.6.0. (#1057)

Changed

- Small dependency update for Noise. (#1038)

1.8.1 - 2023-12-19

Security

- Update `golang.org/x/crypto`, which includes a fix for CVE-2023-48795. (#1048)

Fixed

- Fix a deadlock introduced in v1.8.0 that could occur during handshakes. (#1044)
- Fix mobile builds. (#1035)

1.8.0 - 2023-12-06

Deprecated

- The next minor release of Nebula, 1.9.0, will require at least Windows 10 or Windows Server 2016. This is because support for earlier versions was removed in Go 1.21. See https://go.dev/doc/go1.21#windows

Added

- Linux: Notify systemd of service readiness. This should resolve timing issues with services that depend on Nebula being active. For an example of how to enable this, see: `examples/service_scripts/nebula.service`. (#929)
- Windows: Use Registered IO (RIO) when possible. Testing on a Windows 11 machine shows ~50x improvement in throughput. (#905)
- NetBSD, OpenBSD: Added rudimentary support. (#916, #812)
- FreeBSD: Add support for naming tun devices. (#903)

Changed

- `pki.disconnect_invalid` will now default to true. This means that once a certificate expires, the tunnel will be disconnected. If you use SIGHUP to reload certificates without restarting Nebula, you should ensure all of your clients are on 1.7.0 or newer before you enable this feature. (#859)

- Limit how often a busy tunnel can requery the lighthouse. The new config option `timers.requery_wait_duration` defaults to `60s`. (#940)
- The internal structures for hostmaps were refactored to reduce memory usage and the potential for subtle bugs. (#843, #938, #953, #954, #955)
- Lots of dependency updates.

Fixed

- Windows: Retry wintun device creation if it fails the first time. (#985)
- Fix issues with firewall reject packets that could cause panics. (#957)
- Fix relay migration during re-handshakes. (#964)
- Various other refactors and fixes. (#935, #952, #972, #961, #996, #1002, #987, #1004, #1030, #1032, …)

1.7.2 - 2023-06-01

Fixed

- Fix a freeze during config reload if the `static_host_map` config was changed. (#886)

1.7.1 - 2023-05-18

Fixed

- Fix IPv4 addresses returned by `static_host_map` DNS lookup queries being treated as IPv6 addresses. (#877)

1.7.0 - 2023-05-17

Added

- `nebula-cert ca` now supports encrypting the CA's private key with a passphrase. Pass `-encrypt` in order to be prompted for a passphrase. Encryption is performed using AES-256-GCM and Argon2id for KDF. KDF parameters default to RFC recommendations, but can be overridden via CLI flags `-argon-memory`, `-argon-parallelism`, and `-argon-iterations`. (#386)

- Support for curve P256 and BoringCrypto has been added. See README section "Curve P256 and BoringCrypto" for more details. (#865, #861, #769, #856, #803)
- New firewall rule `local_cidr`. This could be used to filter destinations when using `unsafe_routes`. (#507)
- Add `unsafe_route` option `install`. This controls whether the route is installed in the systems routing table. (#831)
- Add `tun.use_system_route_table` option. Set to true to manage unsafe routes directly on the system route table with gateway routes instead of in Nebula configuration files. This is only supported on Linux. (#839)
- The metric `certificate.ttl_seconds` is now exposed via stats. (#782)
- Add `punchy.respond_delay` option. This allows you to change the delay before attempting punchy.respond. Default is 5 seconds. (#721)
- Added SSH commands to allow the capture of a mutex profile. (#737)
- You can now set `lighthouse.calculated_remotes` to make it possible to do handshakes without a lighthouse in certain configurations. (#759)
- The firewall can be configured to send REJECT replies instead of the default DROP behavior. (#738)
- For macOS, an example launchd configuration file is now provided. (#762)

Changed

- Lighthouses and other `static_host_map` entries that use DNS names will now be automatically refreshed to detect when the IP address changes. (#796)
- Lighthouses send ACK replies back to clients so that they do not fall into connection testing as often by clients. (#851, #408)
- Allow the `listen.host` option to contain a hostname. (#825)
- When Nebula switches to a new certificate (such as via SIGHUP), we now rehandshake with all existing tunnels. This allows firewall groups to be updated and `pki.disconnect_invalid` to know about the new certificate expiration time. (#838, #857, #842, #840, #835, #828, #820, #807)

Fixed

- Always disconnect blocklisted hosts, even if `pki.disconnect_invalid` is not set. (#858)

- Dependencies updated and go1.20 required. (#780, #824, #855, #854)
- Fix possible race condition with relays. (#827)
- FreeBSD: Fix connection to the localhost's own Nebula IP. (#808)
- Normalize and document some common log field values. (#837, #811)
- Fix crash if you set unlucky values for the firewall timeout configuration options. (#802)
- Make DNS queries case insensitive. (#793)
- Update example systemd configurations to want `nss-lookup`. (#791)
- Errors with SSH commands now go to the SSH tunnel instead of stderr. (#757)
- Fix a hang when shutting down Android. (#772)

1.6.1 - 2022-09-26

Fixed

- Refuse to process underlay packets received from overlay IPs. This prevents confusion on hosts that have unsafe routes configured. (#741)
- The ssh `reload` command did not work on Windows, since it relied on sending a SIGHUP signal internally. This has been fixed. (#725)
- A regression in v1.5.2 that broke unsafe routes on Mobile clients has been fixed. (#729)

1.6.0 - 2022-06-30

Added

- Experimental: nebula clients can be configured to act as relays for other nebula clients. Primarily useful when stubborn NATs make a direct tunnel impossible. (#678)
- Configuration option to report manually specified `ip:port`s to lighthouses. (#650)
- Windows arm64 build. (#638)
- `punchy` and most `lighthouse` config options now support hot reloading. (#649)

Changed

- Build against go 1.18. (#656)
- Promoted `routines` config from experimental to supported feature. (#702)
- Dependencies updated. (#664)

Fixed

- Packets destined for the same host that sent it will be returned on MacOS. This matches the default behavior of other operating systems. (#501)
- `unsafe_route` configuration will no longer crash on Windows. (#648)
- A few panics that were introduced in 1.5.x. (#657, #658, #675)

Security

- You can set `listen.send_recv_error` to control the conditions in which `recv_error` messages are sent. Sending these messages can expose the fact that Nebula is running on a host, but it speeds up re-handshaking. (#670)

Removed

- `x509` config stanza support has been removed. (#685)

1.5.2 - 2021-12-14

Added

- Warn when a non lighthouse node does not have lighthouse hosts configured. (#587)

Changed

- No longer fatals if expired CA certificates are present in `pki.ca`, as long as 1 valid CA is present. (#599)
- `nebula-cert` will now enforce ipv4 addresses. (#604)

- Warn on macOS if an unsafe route cannot be created due to a collision with an existing route. (#610)
- Warn if you set a route MTU on platforms where we don't support it. (#611)

Fixed

- Rare race condition when tearing down a tunnel due to `recv_error` and sending packets on another thread. (#590)
- Bug in `routes` and `unsafe_routes` handling that was introduced in 1.5.0. (#595)
- `-test` mode no longer results in a crash. (#602)

Removed

- `x509.ca` config alias for `pki.ca`. (#604)

Security

- Upgraded `golang.org/x/crypto` to address an issue which allowed unauthenticated clients to cause a panic in SSH servers. (#603)

1.5.1 - 2021-12-13

(This release was skipped due to discovering #610 and #611 after the tag was created.)

1.5.0 - 2021-11-11

Added

- SSH `print-cert` has a new `-raw` flag to get the PEM representation of a certificate. (#483)
- New build architecture: Linux `riscv64`. (#542)
- New experimental config option `remote_allow_ranges`. (#540)
- New config option `pki.disconnect_invalid` that will tear down tunnels when they become invalid (through expiry or removal of root trust). Default is **`false`**. Note, this will not currently recognize if a remote has changed certificates since the last handshake. (#370)

- New config option `unsafe_routes.<route>.metric` will set a metric for a specific unsafe route. It's useful if you have more than one identical route and want to prefer one against the other. (#353)

Changed

- Build against go 1.17. (#553)
- Build with `CGO_ENABLED=0` set, to create more portable binaries. This could have an effect on DNS resolution if you rely on anything non-standard. (#421)
- Windows now uses the wintun driver which does not require installation. This driver is a large improvement over the TAP driver that was used in previous versions. If you had a previous version of `nebula` running, you will want to disable the tap driver in Control Panel, or uninstall the `tap0901` driver before running this version. (#289)
- Darwin binaries are now universal (works on both amd64 and arm64), signed, and shipped in a notarized zip file. `nebula-darwin.zip` will be the only darwin release artifact. (#571)
- Darwin uses syscalls and AF_ROUTE to configure the routing table, instead of using `/sbin/route`. Setting `tun.dev` is now allowed on Darwin as well, it must be in the format `utun[0-9]+` or it will be ignored. (#163)

Deprecated

- The `preferred_ranges` option has been supported as a replacement for `local_range` since v1.0.0. It has now been documented and `local_range` has been officially deprecated. (#541)

Fixed

- Valid recv_error packets were incorrectly marked as "spoofing" and ignored. (#482)
- SSH server handles single `exec` requests correctly. (#483)
- Signing a certificate with `nebula-cert sign` now verifies that the supplied ca-key matches the ca-crt. (#503)
- If `preferred_ranges` (or the deprecated `local_range`) is configured, we will immediately switch to a preferred remote address after the reception of a handshake packet (instead of waiting until 1,000 packets have been sent). (#532)

- A race condition when `punchy.respond` is enabled and ensures the correct vpn ip is sent a punch back response in highly queried node. (#566)
- Fix a rare crash during handshake due to a race condition. (#535)

1.4.0 - 2021-05-11

Added

- Ability to output qr code images in `print`, `ca`, and `sign` modes for `nebula-cert`. This is useful when configuring mobile clients. (#297)
- Experimental: Nebula can now do work on more than 2 cpu cores in send and receive paths via the new `routines` config option. (#382, #391, #395)
- ICMP ping requests can be responded to when the `tun.disabled` is **`true`**. This is useful so that you can "ping" a lighthouse running in this mode. (#342)
- Run smoke tests via `make smoke-docker`. (#287)
- More reported stats, udp memory use on linux, build version (when using Prometheus), firewall, handshake, and cached packet stats. (#390, #405, #450, #453)
- IPv6 support for the underlay network. (#369)
- End to end testing, run with `make e2e`. (#425, #427, #428)

Changed

- Darwin will now log stdout/stderr to a file when using `-service` mode. (#303)
- Example systemd unit file now better arranged startup order when using `sshd` and other fixes. (#317, #412, #438)
- Reduced memory utilization/garbage collection. (#320, #323, #340)
- Reduced CPU utilization. (#329)
- Build against go 1.16. (#381)
- Refactored handshakes to improve performance and correctness. (#401, #402, #404, #416, #451)
- Improved roaming support for mobile clients. (#394, #457)
- Lighthouse performance and correctness improvements. (#406, #418, #429, #433, #437, #442, #449)

- Better ordered startup to enable `sshd`, `stats`, and `dns` subsystems to listen on the nebula interface. (#375)

Fixed

- No longer report handshake packets as `lost` in stats. (#331)
- Error handling in the `cert` package. (#339, #373)
- Orphaned pending hostmap entries are cleaned up. (#344)
- Most known data races are now resolved. (#396, #400, #424)
- Refuse to run a lighthouse on an ephemeral port. (#399)
- Removed the global references. (#423, #426, #446)
- Reloading via ssh command avoids a panic. (#447)
- Shutdown is now performed in a cleaner way. (#448)
- Logs will now find their way to Windows event viewer when running under `-service` mode in Windows. (#443)

1.3.0 - 2020-09-22

Added

- You can emit statistics about non-message packets by setting the option `stats.message_metrics`. You can similarly emit detailed statistics about lighthouse packets by setting the option `stats.lighthouse_metrics`. See the example config for more details. (#230)
- We now support freebsd/amd64. This is experimental, please give us feedback. (#103)
- We now release a binary for `linux/mips-softfloat` which has also been stripped to reduce filesize and hopefully have a better chance on running on small mips devices. (#231)
- You can set `tun.disabled` to true to run a standalone lighthouse without a tun device (and thus, without root). (#269)
- You can set `logging.disable_timestamp` to remove timestamps from log lines, which is useful when output is redirected to a logging system that already adds timestamps. (#288)

Changed

- Handshakes should now trigger faster, as we try to be proactive with sending them instead of waiting for the next timer tick in most cases. (#246, #265)
- Previously, we would drop the conntrack table whenever firewall rules were changed during a SIGHUP. Now, we will maintain the table and just validate that an entry still matches with the new rule set. (#233)
- Debug logs for firewall drops now include the reason. (#220, #239)
- Logs for handshakes now include the fingerprint of the remote host. (#262)
- Config item `pki.blacklist` is now `pki.blocklist`. (#272)
- Better support for older Linux kernels. We now only set `SO_REUSEPORT` if `tun.routines` is greater than 1 (default is 1). We also only use the `recvmmsg` syscall if `listen.batch` is greater than 1 (default is 64). (#275)
- It is possible to run Nebula as a library inside of another process now. Note that this is still experimental and the internal APIs around this might change in minor version releases. (#279)

Deprecated

- `pki.blacklist` is deprecated in favor of `pki.blocklist` with the same functionality. Existing configs will continue to load for this release to allow for migrations. (#272)

Fixed

- `advmss` is now set correctly for each route table entry when `tun.routes` is configured to have some routes with higher MTU. (#245)
- Packets that arrive on the tun device with an unroutable destination IP are now dropped correctly, instead of wasting time making queries to the lighthouses for IP `0.0.0.0` (#267)

1.2.0 - 2020-04-08

Added

- Add `logging.timestamp_format` config option. The primary purpose of this change is to allow logging timestamps with millisecond precision. (#187)
- Support `unsafe_routes` on Windows. (#184)

- Add `lighthouse.remote_allow_list` to filter which subnets we will use to handshake with other hosts. See the example config for more details. (#217)
- Add `lighthouse.local_allow_list` to filter which local IP addresses and/or interfaces we advertise to the lighthouses. See the example config for more details. (#217)
- Wireshark dissector plugin. Add this file in `dist/wireshark` to your Wireshark plugins folder to see Nebula packet headers decoded. (#216)
- systemd unit for Arch, so it can be built entirely from this repo. (#216)

Changed

- Added a delay to punching via lighthouse signal to deal with race conditions in some linux conntrack implementations. (#210)

  See deprecated, this also adds a new `punchy.delay` option that defaults to `1s`.
- Validate all `lighthouse.hosts` and `static_host_map` VPN IPs are in the subnet defined in our cert. Exit with a fatal error if they are not in our subnet, as this is an invalid configuration (we will not have the proper routes set up to communicate with these hosts). (#170)
- Use absolute paths to system binaries on macOS and Windows. (#191)
- Add configuration options for `handshakes`. This includes options to tweak `try_interval`, `retries` and `wait_rotation`. See example config for descriptions. (#179)
- Allow `-config` file to not end in `.yaml` or `yml`. Useful when using `-test` and automated tools like Ansible that create temporary files without suffixes. (#189)
- The config test mode, `-test`, is now more thorough and catches more parsing issues. (#177)
- Various documentation and example fixes. (#196)
- Improved log messages. (#181, #200)
- Dependencies updated. (#188)

Deprecated

- `punchy`, `punch_back` configuration options have been collapsed under the now top level `punchy` config directive. (#210)

  `punchy.punch` - This is the old `punchy` option. Should we perform NAT hole punching (default false)?

`punchy.respond` - This is the old `punch_back` option. Should we respond to hole punching by hole punching back (default false)?

Fixed

- Reduce memory allocations when not using `unsafe_routes`. (#198)
- Ignore packets from self to self. (#192)
- MTU fixed for `unsafe_routes`. (#209)

1.1.0 - 2020-01-17

Added

- For macOS and Windows, build a special version of the binary that can install and manage its own service configuration. You can use this with `nebula -service`. If you are building from source, use `make service` to build this feature.
- Support for `mips`, `mips64`, `386` and `ppc64le` processors on Linux.
- You can now configure the DNS listen host and port with `lighthouse.dns.host` and `lighthouse.dns.port`.
- Subnet and routing support. You can now add a `unsafe_routes` section to your config to allow hosts to act as gateways to other subnets. Read the example config for more details. This is supported on Linux and macOS.

Changed

- Certificates now have more verifications performed, including making sure the certificate lifespan does not exceed the lifespan of the root CA. This could cause issues if you have signed certificates with expirations beyond the expiration of your CA, and you will need to reissue your certificates.
- If lighthouse interval is set to `0`, never update the lighthouse (mobile optimization).
- Various documentation and example fixes.
- Improved error messages.
- Dependencies updated.

Fixed

- If you have a firewall rule with `group: ["one-group"]`, this will now be accepted, with a warning to use `group: "one-group"` instead.
- The `listen.host` configuration option was previously ignored (the bind host was always 0.0.0.0). This option will now be honored.
- The `ca_sha` and `ca_name` firewall rule options should now work correctly.

1.0.0 - 2019-11-19

Added

- Initial public release.