RuggedVPN Stable Firmware Release 28. Oktober 2016 - Version 2016100640/2016102400

Dieser Release bringt eine erhebliche Zahl von Verbesserungen im Bereich Qualität, Performanz und Stabilität. Wir empfehlen allen RuggedVPN-Nutzern, zeitnah auf diese Firmware umzusteigen. Wir empfehlen zudem allen Kunden, die noch Classic-Firmware verwenden, nun zeitnah auf diese Firmware umzusteigen, da das Ende des Supportzeitraums für die Classic-Firmware bald endet.

Sollten Sie von einer älteren Classic-Firmware umsteigen wollen, müssen Sie zunächst Ihren Router auf die letzte Classic-Firmware (Version 2015081830/2015102900, veröffentlicht am 27. November 2015) aktualisieren. Anschließend steht das Upgrade auf RuggedVPN zur Verfügung. Bitte beachten Sie dass ein Upgrade der Firmware von Classic zu RuggedVPN eine aktivierte und installierte Viprinet Lifetime Maintenance Lizenz erfordert. Weitere Informationen hierzu erhalten Sie unter https://www.viprinet.com/vlm.

Router und Hubs, die noch Classic-Firmware verwenden, können zu Routern und Hubs verbinden, die RuggedVPN-Firmware verwenden. Allerdings wird in diesem Falle ein Kompatibilitätsmodus verwendet, der den "kleinsten gemeinsamen Nenner" verwendet und daher keine gute Performanz oder Features liefert. Ein solches Setup sollte also nicht dauerhaft, sondern nur während einer Migrationsphase verwendet werden. Der Software VPN Client verwendet aktuell einen auf der Classic-Firmware basierenden Kern und nutzt daher immer den Kompatibilitätsmodus. Eine neue Version des Software VPN Clients mit RuggedVPN-Kern wird in nächster Zeit veröffentlicht werden.

Nachfolgend eine Liste aller neuen Features und Fehlerkorrekturen im Vergleich zur dritten RuggedVPN Firmware-Version (2016080240/2016080800, veröffentlicht am 11. August 2016):

Neue Funktionen

- Diese Firmware bietet keine neuen Funktionen.

Fehlerbehebungen

- Seit geraumer Zeit wurden Channel disconnects nicht sauber ausgeführt. Dies könnte zu Abstürzen führen. In weniger schlimmen Fällen traten nur SSL-Fehler auf, oder es dauerte 5 Sekunden bis der Channel abgebaut war, statt direkt sauber getrennt zu werden.
- Auf einigen Produkten (310, 2620, 2030, 5000, 5010) konnte die Hardwareverschlüsselungsengine während Channel disconnects für einen Absturz sorgen. Wir vermuten dass dieser Fehler der Grund ist, wieso Kunden mit Routern, welche unter großem Stress stehen (Satellitenverbindungen, Schiffe, Fahrzeuge mit häufigen Verbindungsneuaufbau), häufige Reboots sehen, während andere Kunden gar nicht betroffen sind.
- Module, die im Stacking-Verbund als Slave genutzt werden konnten nicht verbinden, wenn die errechnete MTU kleiner als 1500 war. Das führte dazu dass 500er Router nicht als Stacking Slave verwendet werden konnte (das für UMTS verwendete PPP-Protokoll verlangt nach MTUs kleiner 1500).
- Das automatische Kontaktieren des Lizenzservers funktioniert jetzt - zuvor musste man Lizenzen manuell neu abrufen lassen.
- Bei LTE-Modulen findet, anders in vorherigen Firmwareversionen, die Einwahl nicht mehr direkt statt, sondern wird intern ausgelöst durch eine Änderung des LTE-Profils. Wir hatten dies zuvor geändert, da es Kunden gab, die mit der Nutzung von privaten APN Profilen Probleme hatten. Das neue Verfahren hat nun aber Probleme mit einigen Providern in manchen Ländern ergeben. Das Einwahlverfahren wurde daher zurückgeändert, damit die LTE-Profile der jeweiligen Provider genutzt werden. Kunden mit privaten APNs können daher nun wieder Probleme haben. Diese können das neue "Custom WWAN Profile"-Feature nutzen, um das Problem zu umgehen.
- Spezielle Demo- und Projektrouter waren nicht in der Lage, Stacking Slave Channels zu nutzen. Das wurde korrigiert. Betroffene Kunden müssen das WAN-Modul betroffener Channel neu auswählen, nachdem diese Firmware installiert wurde.
- Die Popup-Meldungen bei Demo-Routers, in denen darauf hingewiesen wird dass eine Produktvorführung nur 14 Tage dauern darf, wurde angepasst und teilt nur korrekterweise mit, dass dafür 90 Tage zur Verfügung stehen.
- Die Art wie intern Statistiken über die Quell- und Zielhosts von Traffic-Flows gehandhabt werden wurde überarbeitet. Zuvor konnte man mit DDoS-Attacken mit gefälschten Quell-IP-Adressen das gesamte RAM des Routers aufbrauchen. Derartige Angriffe machen Viprinet-Routern nun nichts mehr aus. Zudem hat sich durch die Änderungen die Perfomanz bei Nutzung mit einer sehr hohen Zahl (1000+) von Geräten im LAN merklich verbessert.
- Bei bestimmten Arten von DoS-Angriffen konnte der Routingkern festhängen, was nach 90 Sekunden einen Neustart des Routers auslöste.
- Im Falle dass ein DDoS-Angriff vom Router erkannt wird, wird nun ein Hinweis im Log ausgegeben.
- Ein Fehler in der Speicherverwaltung bei IP-Paketen wurde korrigiert. Der Fehler konnte ein kleines Speicherleck auslösen, was im Laufe der Wochen zu eine erheblichen Größe anwachsen konnte. Der Fehler konnte zudem unter sehr speziellen Bedingungen auch zum Absturz des Routers führen.
- Das setzen einer IPv6-Adresse als Haupt-IP des LAN-Interface (anstatt die V6-Adresse als Alias hinzuzufügen) war zuvor nicht zulässig, konnte aber dennoch durchgeführt werden. Damit wurde der Router aus dem LAN-Netzwerk unerreichbar. Der Router verhindert nun, dass man auf irgendeinem Weg eine v6-Adresse als Haupt-IP angibt.

- Der Neuübertragungsmanager für QoS-Klassen mit aktivierter "Guaranteed delivery" hatte einen Speicherleck, konnte aber unter Umständen auch bewirken, dass die Hälfte aller Neuübertragungeng gar nicht stattfanden. Das bedeutete, dass ein Flow durch den Tunnel trotz garantierter Übertragung doch Paketverluste erleidigen konnte. Das konnte dramatische Konsequenzen haben: Die TCP/IP Headerkompression baut auf die garantierte Übertragung auf, es dürfen nie Pakete fehlen - geschah dies doch, wäre der entsprechende Flow steckengeblieben.
- Die Logausgabe von TCP-Sequenznummern wurde korrigiert. Zudem wurde der Loglevel für sehr häufige TCP-Protokollverstöße, welche von kaputten IP-Scannern ausgelöst werden, nicht länger als Angriff protokolliert.
- Es wird nun sichergestellt dass serverseitig hängende HTTP-Verbindungen bei Nutzung der Download-Testtools des Webinterfaces sauber abgebaut werden. Der zuvor vorhandene Bug konnte für 99% CPU-Last sorgen, wenn ein Download serverseitig hängenblieb.
- Der Routingkern konnte unter außergewöhnlichen Umständen unregelmäßig auf der Empfangsseite von Flows hängenbleiben. Der Fehler trat bei Kunden teilweise über Wochen und Monate nicht auf, um dann plötzlich für einen Tag ständig aufzutreten.
- Einzelne Flows mit aktivierter "Guaranteed Delivery" konnten bei einem Neuverbinden des VPN-Tunnels steckenbleiben, wenn sie zu diesem Zeitpunkt eine hohe Übertragungsrate aufwiesen. Hängengebliebene Flows haben in diesem Falle das RAM gefüllt, bis dem Router ggf der Speicher ausging.
- Die empfangsseitigen Timeouts für Flows ohne "Guaranteed Delivery" sorgen dafür, dass beim Neusortieren der empfangenen Paketfragmente nur so lange auf fehlende Teile gewartet wird, wie die Bündelungslatenz erwarten lassen sollte. Verstreicht diese Zeit, sollte das entsprechende Paket verworfen werden (aus Sicht des Zielsystems handelt es sich dann um einen Paketverlist). Die Filter, um die richtige maximale Wartezeit zu ermitteln, waren bisher nicht ausgereift. Insbesondere bei Bündelung von Leitungen mit sehr hoher oder schwankender Latenz (z.B. Satellit) konnten die Timeout-Werte ungünstig sein. Dies konnte dazu führen dass aufgrund von scheinbar verlorenen Paketen auf Empfangsseite ein Nutzer z.B. bei einem Download nicht die volle Bandbreite ausnutzen konnte, die der VPN-Tunnel eigentlich zur Verfügung stellt. Die Filter wurden nun komplett neu geschrieben und ausführlich getestet.
- Die Meldungen "Average latency is ... ms, maximum allowed is ... ms, no alternatives, keeping channel." wurden komplett entfernt, wie auch die zugrundeliegende Idee: Es macht keinen Sinn einen Channel kramphaft als verbunden zu forcieren, wenn die Latenz über der liegt, die vom Benutzer als maximal angebenen ist. Es ist sinnvoller in diesem Fall die Verbindung des Channels zu trennen, und die Pufferung auflaufender Pakete stattdessen auf der Tunnelebene durchzuführen.