

TECHNISCHES MERKBLATT

Broker hinter NAT anbinden

Für den Kunden, der einen lokalen Broker anbindet

Läuft Ihr MQTT-Broker auf einem aus dem Internet nicht erreichbaren Rechner, kann der gehostete Analyzer nicht hineinverbinden. Stattdessen baut Ihr Rechner einen **ausgehenden** SSH-Tunnel auf. Er geht durch fast jede Firewall/NAT nach außen, ist verschlüsselt und öffnet bei Ihnen **keinen eingehenden Port**. Ihr Broker wird nur innerhalb unseres Servers sichtbar, ausschließlich für den Analyzer.

Drei Dinge passieren einmalig, in dieser Reihenfolge: Sie erzeugen einen Schlüssel, schicken uns den öffentlichen Teil und nach unserer Freischaltung starten Sie den Tunnel.

Funktionsweise

Ihr Rechner	logiq-vda5050.com
Broker auf localhost:1883	SSH (User: logiqtun)
ssh -R (ausgehend, Port 22)	
+----->	der Analyzer erreicht Ihren Broker
	intern, auf dem Ihnen zugewiesenen Port

Der Tunnel wird von Ihrer Seite aufgebaut, es muss also nichts zu Ihnen hin geöffnet werden; SSH verschlüsselt die gesamte Strecke.

Schritt 1: Schlüssel erzeugen

Ein eigener SSH-Schlüssel, mit leerer Passphrase, damit der Tunnel unbeaufsichtigt laufen kann. OpenSSH ist in aktuellem Windows, Linux und macOS eingebaut.

Linux / macOS

```
ssh-keygen -t ed25519 -f ~/.ssh/logiq_tunnel -N "" -C "ihr-broker"
```

Windows (PowerShell)

```
ssh-keygen -t ed25519 -f "$env:USERPROFILE\.ssh\logiq_tunnel" -C "ihr-broker"
```

Bei *Enter passphrase* zweimal Enter drücken (leer lassen). Fehlt `ssh`, den **OpenSSH-Client** unter Einstellungen → Apps → Optionale Features installieren.

Schritt 2: Öffentlichen Schlüssel senden

Schicken Sie den Inhalt der `.pub`-Datei, nur den öffentlichen Teil, niemals die Datei ohne Endung.

Linux / macOS

```
cat ~/.ssh/logiq_tunnel.pub
```

Windows (PowerShell)

```
Get-Content "$env:USERPROFILE\.ssh\logiq_tunnel.pub"
```

Kopieren Sie die Zeile (beginnt mit `ssh-ed25519 ...`) in Ihre Antwort an uns. Wir schalten sie frei und weisen Ihnen einen Port zu, die Beispiele unten nutzen **19883**.

Schritt 3: Tunnel starten

Ersetzen Sie `localhost:1883` durch Ihren Broker so, wie Sie ihn vom eigenen Rechner aus erreichen (läuft der Broker auf demselben Rechner, passt `localhost:1883`). `19883` durch den Ihnen zugewiesenen Port ersetzen.

Linux / macOS (autossh, Auto-Reconnect)

Zuerst `autossh` installieren (`apt install autossh`, `brew install autossh` oder `dnf install autossh`), dann:

```
autossh -M 0 -f -N \  
  -o "ServerAliveInterval 30" -o "ServerAliveCountMax 3" \  
  -o "ExitOnForwardFailure yes" \  
  -i ~/.ssh/logiq_tunnel \  
  -R 172.17.0.1:19883:localhost:1883 \  
  logiqtun@logiq-vda5050.com
```

Windows (PowerShell, Testlauf)

Läuft im Vordergrund; Fenster offen lassen. Keine Fehlermeldung heißt: Tunnel steht.

```
ssh -N -o "ServerAliveInterval=30" -o "ServerAliveCountMax=3" -o "ExitOnForwardFailure=yes" \  
-o "StrictHostKeyChecking=accept-new" -i "$env:USERPROFILE\.ssh\logiq_tunnel" -R \  
172.17.0.1:19883:localhost:1883 logiqtun@logiq-vda5050.com
```

Sagen Sie uns kurz Bescheid, wenn er läuft, dann sehen wir Ihren Broker sofort im Analyzer.

Schritt 4: Dauerhaft betreiben

Für eine dauerhafte Verbindung, die einen Neustart übersteht und sich selbst neu verbindet.

Linux (systemd-Dienst)

Als `/etc/systemd/system/logiq-tunnel.service` speichern (`USER` durch Ihren Login ersetzen), dann `systemctl enable --now logiq-tunnel`.

```
[Unit]
Description=LogIQ broker reverse tunnel
After=network-online.target
Wants=network-online.target

[Service]
User=USER
ExecStart=/usr/bin/autossh -M 0 -N \
  -o "ServerAliveInterval 30" -o "ExitOnForwardFailure yes" \
  -i /home/USER/.ssh/logiq_tunnel \
  -R 172.17.0.1:19883:localhost:1883 \
  logiqtun@logiq-vda5050.com
Restart=always
RestartSec=10

[Install]
WantedBy=multi-user.target
```

Windows (Skript + Aufgabenplanung)

Diese Schleife als `%USERPROFILE%\logiq-tunnel.ps1` speichern:

```
$key = "$env:USERPROFILE\.ssh\logiq_tunnel"
while ($true) {
  ssh -N -o "ServerAliveInterval=30" -o "ServerAliveCountMax=3" -o
  "ExitOnForwardFailure=yes" -o "StrictHostKeyChecking=accept-new" -i $key -R
  172.17.0.1:19883:localhost:1883 logiqtun@logiq-vda5050.com
  Start-Sleep -Seconds 10
}
```

Dann bei der Anmeldung starten lassen (einmal in PowerShell):

```
schtasks /Create /TN "LogIQ Broker Tunnel" /SC ONLOGON /RL LIMITED /F ^
/TR "powershell -NoProfile -WindowStyle Hidden -ExecutionPolicy Bypass -File
%USERPROFILE%\logiq-tunnel.ps1"
```

Für einen Server ohne aktive Anmeldung die Aufgabe auf *Unabhängig von der Benutzeranmeldung ausführen* setzen.

Was wir einstellen

Im Analyzer zeigt Ihr Broker-Profil auf Host `172.17.0.1`, Port `19883`, mit **TLS aus**, die Strecke verschlüsselt bereits der SSH-Tunnel. Verlangt Ihr Broker Benutzer und Passwort, sagen Sie uns Bescheid; wir hinterlegen es verschlüsselt.

Sicherheit

- Genutzt wird nur **ausgehendes** SSH (Port 22). In Ihrem Netz wird nichts geöffnet, weitergeleitet oder exponiert und der Broker ist nie aus dem Internet erreichbar.
- Der Schlüssel kann **nur diesen einen** Tunnel-Port öffnen, keine Shell, kein Dateizugriff, keine anderen Ports. Sie können den Tunnel jederzeit stoppen (Fenster schließen oder Dienst beenden).

Fragen zur Anbindung: support@logiq-vda5050.com