BLUEDOT

OPERATIONS NOTE

# Trusting the BNG TLS Certificate

## Operations Note — Resolving the Browser Warning

*Why the browser warns about the HTTPS connection to the QuantumTouch management interface, and the three ways to resolve it — import a certificate you already own, import the controller's local CA into your OS trust store, or accept the warning — in order from easiest to most involved.*

| | |
|---|---|
| **Document Type** | BlueDot Operations Note |
| **Component** | QuantumTouch — management TLS |
| **Audience** | Operators & Support Engineers |
| **See Also** | *Let's Encrypt on the BNG* |
| **Last Updated** | 28 June 2026 |
| **Classification** | **BlueDot Proprietary** |
| **Date** | October 2026 |

© 2026 BlueDot · All rights reserved.

# Contents

Why the warning appears ..... 3
Option 1 — Import a certificate you already own (recommended) ..... 3
Option 2 — Import the controller's local CA into your OS trust store ..... 3
- macOS ..... 4
- Windows ..... 4
- Ubuntu / Debian ..... 4

Option 3 — Accept the browser warning directly ..... 5
Footnote — What is in the certificate's SAN? ..... 5

What this covers: why your browser warns about the HTTPS connection to the BNG management interface, and the three ways to resolve it — from easiest to most involved.

Last updated: 2026-06-28

# Why the warning appears

QuantumTouch serves its management UI on port 8443 over HTTPS. The certificate is either self-signed or signed by the controller's own local CA — a private certificate authority that was created at install time and exists only on this box. Because your browser has never seen that CA before, it cannot verify the certificate chain, so it shows an "untrusted" or "potential security risk" warning.

The connection is still encrypted. The warning means the browser cannot confirm who issued the certificate, not that the traffic is unprotected.

# Option 1 — Import a certificate you already own (recommended)

If your customer or organization has a TLS certificate (from a public CA like DigiCert, Sectigo, or Let's Encrypt) for the hostname or IP of this box, you can load it directly through the UI:

1. Log in to the BNG management interface.
2. Navigate to **About** → **Manage TLS cert** → **Import existing** tab.
3. Paste or upload your certificate (PEM format) and private key.
4. Click **Apply**. The service reloads the certificate in-place — no restart needed.

After import, any browser that already trusts the issuing CA will connect without a warning. This is the right approach for production deployments where a certificate exists.

# Option 2 — Import the controller's local CA into your OS trust store

If you do not have an external certificate, you can add the controller's own CA to the trust stores of the machines your operators use. Once the OS trusts that CA, every certificate the controller issues (including future reissues) will be trusted automatically.

The CA certificate lives on the BNG box at:

```
/var/lib/quantum-controller/tls/ca-cert.pem
```

Copy it down to your workstation first:

```
scp lanner@192.168.X.Y:/var/lib/quantum-controller/tls/ca-cert.pem ./quantum-controller-ca.pem
```

Then import it into the trust store for your OS:

## macOS

1. Open **Keychain Access** (Applications → Utilities → Keychain Access).
2. Select the **System** keychain in the left panel (not Login, not Local Items).
3. Drag and drop `quantum-controller-ca.pem` onto the Keychain Access window.
4. Double-click the newly added certificate, expand the **Trust** section, and set "When using this certificate" to **Always Trust**.
5. Close the window and enter your macOS password when prompted.

Alternatively, from a terminal:

```
sudo security add-trusted-cert -d -r trustRoot -k /Library/Keychains/System.keychain quantum-controller-ca.pem
```

## Windows

1. Press `Win+R`, type `mmc.exe`, press Enter.
2. In the Microsoft Management Console: **File** → **Add/Remove Snap-in** → select **Certificates** → **Add** → choose **Computer account** → **Next** → **Local computer** → **Finish** → **OK**.
3. Expand **Certificates (Local Computer)** → **Trusted Root Certification Authorities** → **Certificates**.
4. Right-click the **Certificates** folder → **All Tasks** → **Import**.
5. Follow the wizard and select `quantum-controller-ca.pem` when asked for the file. Ensure it is placed in "Trusted Root Certification Authorities".

## Ubuntu / Debian

```
sudo cp quantum-controller-ca.pem /usr/local/share/ca-certificates/quantum-controller.crt
sudo update-ca-certificates
```

Note the `.crt` extension — `update-ca-certificates` requires it. Restart your browser after running the update.

# Option 3 — Accept the browser warning directly

For one-off access, lab use, or testing, you can accept the browser's security exception:

- **Chrome / Edge**: click **Advanced** → **Proceed to \<address> (unsafe)**.
- **Firefox**: click **Advanced** → **Accept the Risk and Continue**.

The exception is stored per browser profile and is tied to the specific IP address and port. If you access the box through a different hostname, a VPN tunnel, or a new browser profile, you will need to accept the warning again.

This option is not recommended for production use because it trains operators to ignore certificate warnings and does not carry over to other machines or tools (such as curl scripts or monitoring agents) that also connect to the API.

# Footnote — What is in the certificate's SAN?

If you need to check exactly which hostnames and IPs the current certificate covers (its Subject Alternative Names), run this from any machine that can reach the box:

```
openssl s_client -connect 192.168.X.Y:8443 -servername bng < /dev/null \
  | openssl x509 -text -noout \
  | grep -A1 'Subject Alternative Name'
```

Replace `192.168.X.Y` with the management IP of the box. The output will show the DNS names and IP addresses the certificate is valid for. If the address you are using is not listed, even a fully-trusted CA will produce a mismatch warning.