INTERPRETATION IC 135-2016-4 OF
ANSI/ASHRAE STANDARD 135-2016 BACnet® -
A Data Communication Protocol for Building
Automation and Control Networks

Approval Date: June 26, 2017

Request from: Dave Robin, Automated Logic, 1150 Roberts Blvd, Kennesaw, GA 30144.

Reference: This request for interpretation refers to the requirements presented in ANSI/ASHRAE Standard 135-2016, Clause **W.5.3 The .auth Data Item**, regarding the format of public keys.

Background: The format defined for the public keys is not possible. The existing clause defines a format for "all keys", but that format is only appropriate for private keys and therefore cannot be used for the two public keys in the /.auth tree.

> ***W.5.3 The .auth Data Item***
> *The .auth data item contains information related to the server device's security. The meaning of this data is discussed in Clause W.3. All data under the /.auth path, with the exception of the "{item}-pend" items, shall be nonvolatile. All Certificates shall be X.509 certificates in binary DER format with a mediaType "application/x-x509-ca-cert" and all keys shall be in PKCS #8 binary DER format (RFC 5958) with a mediaType "application/pkcs8". The complete list of children is defined in the following table.*

The most common form of public key serialization is a DER encoding of a SubjectPublicKeyInfo structure, defined by X.509 in Section 4.1.2.7 of RFC 5280. In addition to all the places where X.509 certificates are used, this is the encoding used by the ubiquitous PEM format (i.e., "-----BEGIN PUBLIC KEY-----") defined by Section 13 of RFC 7468, and also used by the "Raw Public Key" TLS extension in Section 3 of RFC 7250.

If the keys in /.auth were defined to be CharacterString values, there could be some ambiguity about whether the PEM "-----BEGIN/END-----" wrapper should be present or not. But since they are defined to be OctetString values, the binary DER content is the only obvious choice.

Command line tools like "openssl" can read/generate/convert keys using this binary format so no custom programming is needed.

Interpretation: Since the storage of the public keys in the prescribed format is not possible, it is assumed that this is an errata/oversight and that an appropriate public key format was intended. And since there seems to be only one common and widely supported way to encode a public key, the assumption is that the format to be used for the public key values in the /.auth structure is a DER encoding of a SubjectPublicKeyInfo as defined by X.509.

Question: Is this Interpretation correct?

Answer: Yes