BSR/ASHRAE Addendum *by* to
ANSI/ASHRAE Standard 135-2016

Public Review Draft

Proposed Addendum *by* to Standard 135-2016, BACnet® - A Data Communication Protocol for Building Automation and Control Networks

First Public Review (June 2019)
(Draft shows Proposed Changes to Current Standard)

This draft has been recommended for public review by the responsible project committee. To submit a comment on this proposed standard, go to the ASHRAE website at www.ashrae.org/standards-research--technology/public-review-drafts and access the online comment database. The draft is subject to modification until it is approved for publication by the Board of Directors and ANSI. Until this time, the current edition of the standard (as modified by any published addenda on the ASHRAE website) remains in effect. The current edition of any standard may be purchased from the ASHRAE Online Store at www.ashrae.org/bookstore or by calling 404-636-8400 or 1-800-727-4723 (for orders in the U.S. or Canada).

This standard is under continuous maintenance. To propose a change to the current standard, use the change submittal form available on the ASHRAE website, www.ashrae.org.

The appearance of any technical data or editorial material in this public review document does not constitute endorsement, warranty, or guaranty by ASHARE of any product, service, process, procedure, or design, and ASHRAE expressly disclaims such.

ASHRAE, 1791 Tullie Circle, NE, Atlanta GA 30329-2305

[This foreword and the "rationales" on the following pages are not part of this standard. They are merely informative and do not contain requirements necessary for conformance to the standard.]

FOREWORD

The purpose of this addendum is to present a proposed change for public review. These modifications are the result of change proposals made pursuant to the ASHRAE continuous maintenance procedures and of deliberations within Standing Standard Project Committee 135. The proposed changes are summarized below.

135-2016*by*-1. Remove Clause 24, Network Security, p. 3

In the following document, language to be added to existing clauses of ANSI/ASHRAE 135-2016 and Addenda is indicated through the use of *italics*, while deletions are indicated by ~~strikethrough~~. Where entirely new subclauses are proposed to be added, plain type is used throughout. Only this new and deleted text is open to comment at this time. All other material in this document is provided for context only and is not open for public review comment except as it relates to the proposed changes.

The use of placeholders like X, Y, Z, X1, X2, N, NN, x, n, ?, etc., should not be interpreted as literal values of the final published version. These placeholders will be assigned actual numbers/letters only after final publication approval of the addendum.

135-2016*by*-1. Remove Clause 24, Network Security

Rationale

Clause 24, Network Security has very limited industry acceptance and is unlikely to be implemented with the pending addition of BACnet/SC (Addendum bj) to the standard. Keeping Clause 24 in the standard, once BACnet/SC is published, will cause confusion in the industry.

[Change **Clause 4.3**, p. 16]

4.3 Security

The principal security threats to BACnet systems are people who, intentionally or by accident, modify a device's configuration or control parameters. Problems due to an errant computer are outside the realm of security considerations. One important place for security measures is the operator-machine interface. Since the operator-machine interface is not part of the communication protocol, vendors are free to include password protection, audit trails, or other controls to this interface as needed. In addition, write access to any properties that are not explicitly required to be "writable" by this standard may be restricted to modifications made only in virtual terminal mode or be prohibited entirely. This permits vendors to protect key properties with a security mechanism that is as sophisticated as they consider appropriate. ~~BACnet also defines services that can be used to provide peer entity, data origin, and operator authentication. See Clause 24.~~

[Change **Clause 5.1**, p. 17]

5.1 The Application Layer Model

…
The Application Entity is itself made up of two parts: the BACnet User Element and the BACnet Application Service Element (ASE). The BACnet ASE represents the set of functions or application services specified in Clauses 13 through 17~~ and Clause 24~~. The BACnet User Element carries out several functions in addition to supporting the local API. It represents the implementation of the "service procedure" portion of each application service. It is responsible for maintaining information about the context of a transaction, including generating invoke IDs and remembering which invoke ID goes with which application service request (response) to (from) which device. It is also responsible for maintaining the time-out counters that are required for the retrying of a transmission. The BACnet User Element also presides over the mapping of a device's activities into BACnet objects.

Information exchanged between two peer application processes is represented in BACnet as an exchange of abstract service primitives, following the ISO conventions contained in the OSI technical report on service conventions, ISO TR 8509. These primitives are used to convey service-specific parameters that are defined in Clauses 13 through 17~~ and Clause 24~~. Four service primitives are defined: request, indication, response, and confirm. The information contained in the primitives is conveyed using a variety of protocol data units (PDUs) defined in this standard. In order to make clear which BACnet PDU is being used, the notation will be as follows:
…

[Change **Clause 6.2.4**, p. 60]

6.2.4 Network Layer Message Type

…
X'09': Disconnect-Connection-To-Network
X'0A': *removed*~~Challenge-Request~~
X'0B': *removed*~~Security-Payload~~
X'0C': *removed*~~Security-Response~~
X'0D': *removed*~~Request-Key-Update~~
X'0E': *removed*~~Update-Key-Set~~
X'0F': *removed*~~Update-Distribution-Key~~
X'10': *removed*~~Request-Master-Key~~
X'11': *removed*~~Set-Master-Key~~

X'12': What-Is-Network-Number
…

[Change **Clause 6.4.4**, p. 63]

6.4.4 Reject-Message-To-Network

…
4: The message is too long to be routed to this DNET.

5. ~~The source message was rejected due to a BACnet security error and that error cannot be forwarded to the source device. See Clause 24.12.1.1 for more details on the generation of Reject Message To Network messages indicating this reason.~~*This rejection reason value has been removed.*

6. The source message was rejected due to errors in the addressing. The length of the DADR or SADR was determined to be invalid.

[Change **Clauses 6.4.11 to 6.4.18**, pp. 64-65]

6.4.11 ~~Challenge-Request~~Deleted Clause

~~This message is indicated by a Message Type of X'0A'. It is described in Clause 24~~
This clause has been removed.

6.4.12 ~~Security-Payload~~*Deleted Clause*

~~This message is indicated by a Message Type of X'0B'. It is described in Clause 24.~~
This clause has been removed.

6.4.13 ~~Security-Response~~*Deleted Clause*

~~This message is indicated by a Message Type of X'0C'. It is described in Clause 24.~~
This clause has been removed.

6.4.14 ~~Request-Key-Update~~*Deleted Clause*

~~This message is indicated by a Message Type of X'0D'. It is described in Clause 24.~~
This clause has been removed.

6.4.15 ~~Update-Key-Set~~*Deleted Clause*

~~This message is indicated by a Message Type of X'0E'. It is described in Clause 24.~~
This clause has been removed.

6.4.16 ~~Update-Distribution-Key~~*Deleted Clause.*

~~This message is indicated by a Message Type of X'0F'. It is described in Clause 24.~~
This clause has been removed.

6.4.17 ~~Request-Master-Key~~Deleted Clause

~~This message is indicated by a Message Type of X'10'. It is described in Clause 24.~~
This clause has been removed.

6.4.18 ~~Set-Master-Key~~*Deleted Clause*

~~This message is indicated by a Message Type of X'11'. It is described in Clause 24.~~
This clause has been removed.

[Change **Clause 12.49**, pp. 464-466]

12.49 ~~Network Security Object Type~~*Deleted Clause*

[Delete all of Clause 12.49 and replace with:]
This clause has been removed.

[Change **Table 12-71**, p. 518]

…

[1] Required to be writable in routers~~, secure devices,~~ and any other device that requires knowledge of the network number for proper operation.

…

[Change **Clause 12.56.12**, p. 526]

12.56.12 Network_Number

…
This property shall be writable in routers~~, secure devices,~~ and any other device that requires knowledge of the network number for proper operation. Routers are permitted to refuse a value of 0. In that case, the write request shall result in an error response with 'Error Class' of PROPERTY and an 'Error Code' of VALUE_OUT_OF_RANGE.
…

[Change **Clause 24**, pp. 878-931]

24 ~~NETWORK SECURITY~~*DELETED CLAUSE*

[Delete all of Clause 24 and replace with:]

This clause has been removed.

[Change productions in **Clause 21, p. 782]**

BACnetObjectType ::= ENUMERATED { -- see below for numerical order
…
network-port (56),
~~network-security (38),~~
notification-class (15),
…
-- see credential-data-input (37),
~~-- see network-security (38),~~
-- *(38) removed,*
-- see bitstring-value (39),
…

BACnetObjectTypesSupported ::= BIT STRING {
…
credential-data-input (37),
~~network-security (38),~~
-- *(38) removed,*
bitstring-value (39),
…

BACnetPropertyIdentifier ::= ENUMERATED { -- see below for numerical order
…
bacnet-ipv6-udp-port (438),
bacnet-ipv6-multicast-address (440),~~ base-device-security-policy (327),~~
bbmd-accept-fd-registrations (413),

…
direct-reading (156),
~~distribution-key-revision (328),~~
~~do-not-hide (329),~~
door-alarm-state (226),
…
is-utc (344),
~~key-sets (330),~~
landing-call-control (471),
…
last-credential-removed-time (280),
~~last-key-server (331),~~
last-notify-record (173),
…
negative-access-rules (288),
~~network-access-security-policies (332),~~
network-interface-name (424),
…
output-units (82),
~~packet-reorder-time (333),~~
passback-mode (300),
…
schedule-default (174),
secured-status (235),
~~security-pdu-timeout (334),~~
~~security-time-window (335),~~
segmentation-supported (107),
…
supported-formats (304),
~~supported-security-algorithms (336),~~
system-status (112),
…
update-interval (118),
~~update-key-set-timeout (337),~~
update-time (189),
…
-- see verification-time (326),
~~-- see base-device-security-policy (327),~~
~~-- see distribution-key-revision (328),~~
~~-- see do-not-hide (329),~~
~~-- see key-sets (330),~~
~~-- see last-key-server (331),~~
~~-- see network-access-security-policies (332),~~
~~-- see packet-reorder-time (333),~~
~~-- see security-pdu-timeout (334),~~
~~-- see security-time-window (335),~~
~~-- see supported-security-algorithms (336),~~
~~-- see update-key-set-timeout (337),~~
-- *(327) removed,*
-- *(328) removed,*
-- *(329) removed,*
-- *(330) removed,*
-- *(331) removed,*
-- *(332) removed,*
-- *(333) removed,*
-- *(334) removed,*

```
        --                                  (335) removed,
        --                                  (336) removed,
        --                                  (337) removed,
        -- see backup-and-restore-state     (338),
…

BACnetPropertyStates ::= CHOICE {
…
        notify-type                         [25] BACnetNotifyType,
        security-level                      [26] BACnetSecurityLevel,
        --                                  [26] removed
        shed-state                          [27] BACnetShedState,
…

BACnetNetworkSecurityPolicy ::= SEQUENCE {
        port-id                             [0] Unsigned8,
        security-level                      [1] BACnetSecurityPolicy
        }

BACnetKeyIdentifier ::= SEQUENCE {
        algorithm                           [0] Unsigned8,
        key-id                              [1] Unsigned8
        }

BACnetSecurityKeySet ::= SEQUENCE {
        key-revision                        [0] Unsigned8,          -- 0 if key set is not configured
        activation-time                     [1] BACnetDateTime,     -- UTC time, all wild if unknown
        expiration-time                     [2] BACnetDateTime,     -- UTC time, all wild if infinite
        key-ids                             [3] SEQUENCE OF BACnetKeyIdentifier
        }

BACnetSecurityLevel ::= ENUMERATED {
        incapable                           (0),        -- indicates that the device is configured to not use security
        plain                               (1),
        signed                              (2),
        encrypted                           (3),
        signed-end-to-end                   (4),
        encrypted-end-to-end                (5)
        }

BACnetSecurityPolicy ::= ENUMERATED {
        plain-non-trusted                   (0),
        plain-trusted                       (1),
        signed-trusted                      (2),
        encrypted-trusted                   (3)
        }
```

[Change **Clause J.2.13**, p. 1027]

J.2.13 Secure-BVLL: Purpose*Deleted Clause*

[Delete all of Clause J.2.13 and replace with:]
This clause has been removed.

[Change **Clause K.6**, p. 1076]

K.6 ~~Network Security BIBBs~~*Deleted Clause*

[Delete all of Clause K.6 and replace with:]
This clause has been removed.

[Change **Annex S**, p. 1154]

ANNEX S ~~*EXAMPLES OF SECURE BACnet MESSAGES (INFORMATIVE)*~~ *Removed*

[Delete all of Annex S and replace with:]
(This annex has been removed from the standard.)

[Change **Clause U.2.12**, p. 1179]

U.2.12 ~~Secure BVLL: Purpose~~*Deleted Clause*

[Delete all of Clause U.2.12 and replace with:]
This clause has been removed.

[Change **Clause W.3.3**, p. 1192]

W.3.3 Internal Authorization Server

…
For the "Resource Owner Password Credentials Grant" type, the server shall support at least one configurable user name and user password pair, at "/.auth/int/user" and "/.auth/int/pass", with a storage minimum of 16 bytes and 32 bytes respectively. This pair shall authorize any scope presented by the client, including the "auth" scope. Access tokens issued for this pair shall set the user-id field to 0 and the user-role field to 1. See *Clause W.3.X* ~~Clause 24.2.11~~ for definition of user-id and user-role. Support for additional usernames, passwords, and a corresponding limiting of scope is optional and is a local matter.

For the "Client Credentials Grant" type, the server shall support at least one configurable client id and client secret pair at "/.auth/int/id" and "/.auth/int/secret", with a storage minimum of 16 bytes and 32 bytes respectively. This pair shall authorize any scope presented by the client, except the "auth" scope. Access tokens issued for this pair shall set the user-id field to 0 and the user-role field to 0. See *Clause W.3.X* ~~Clause 24.2.11~~ for definition of user-id and user-role. Support for additional client ids and secrets, and a corresponding limiting of scope is optional and is a local matter.
…

[Change **Clause W.3.7**, p. 1197]

W.3.7 Access Token Format

...

Requirements for the JWT 'claims' object:

> …
>
> The "sub" member is optional. If present, it shall consist of a space separated concatenation of the string representations of the numeric "user-id"' and "user-role" fields as defined in *Clause W.3.X* ~~Clause 24.2.11~~. If absent, it is assumed to have the value "0 0". The use of "user-id" and "user-role" for authorization decisions in the resource server is a local matter. It is generally expected that authorization decisions have already been made by the authorization server and are represented by the "scope" member. Therefore the presence of "user-id" and "user-role" in the token is generally only for auditing purposes in the resource server. However, they may also be provided by the authorization server for use by "downstream" communications by ~~gateways to Clause 24~~ *other* secured networks and devices which may require this information.
> …

[Add new **Clause W.3.X**, p. 1199]

W.3.X User and Role Identification

Auditing and local access policies are aided by knowing the identity of the user or entity that initiates an action and what roles they play. Basic identification of this information can be provided by a pair of numbers known as User ID and User Role. See Clause W.3.7 for example usage.

[Note to reviewer: the following text is copied here, rearranged, but substantively unchanged, from Clause 24.2.11]

User ID values are positive integers that represent unique human users or processes within a BACnet system. User ID 0 is reserved to indicate that the user is unknown; it is commonly used in conjunction with User Role 0 or 1.

User Role values are positive integers used to group access rights. Example roles could be: HVAC operator, technician, etc. User Roles 0 and 1 are reserved to mean "the system itself". User Role 0 is used for programmed device-to-device communication that is not initiated by human action. User Role 1 is used for device-to-device communication that is initiated by an "unknown human", such as the changing of a setpoint based on button presses on a thermostat. Other User Role values may also be used for device-to-device communication to indicate a particular subsystem that is performing the action, but those values are not restricted by this standard and are taken from the same set of numbers as are used for roles for human users and groups. The values 0 and 1 are the only ones that are reserved specifically for this purpose and shall not be assigned to human user roles.

[Note to reviewer, Clause 24 required uniqueness "across all BACnet devices", however, "all" was not clearly defined. Therefore, the words "in a given security context" are added below.]

Assignment of the values for User ID and Role is based on local site policy, but they should be unique across all BACnet devices in a given security context, such that User ID 1234, for example, means the same regardless of its source or destination.

[Add a new entry to **History of Revisions**, p. 1349]

HISTORY OF REVISIONS

. . .	. . .	**. . .**
1	X	**Addendum *by* to ANSI/ASHRAE Standard 135-2016** Approved by ASHRAE on MONTH DAY, 20XX; and by the American National Standards Institute on MONTH DAY, 20XX. 1. Remove Clause 24, Network Security