CHIPKIN

CAS BACnet Cli
CAS-1000-18

Chipkin - Enabling Integration

salesgroup1@chipkin.com

Tel: +1 866 383 1657

Software Version: 1
Document Revision: 1

TABLE OF CONTENTS

1 Description

The CAS BACnet CLI is a powerful command-line utility designed for Windows and Linux platforms. It provides a means for discovering, reading, writing, and commanding BACnet IP devices from the command line or scripting.

This tool is intended for advanced users who are familiar with BACnet protocols and command-line tools and wish to automate interactions with BACnet devices for purposes like regression testing, unit testing, and scripting device interactions.

Key Features

- **Automation-Friendly:** Ideal for integrating into scripts to automate BACnet interactions.
- **Testing and Diagnostics:** Designed for use in unit tests and regression tests of BACnet server interfaces.
- **Comprehensive Command Set:** Includes commands for device discovery, object manipulation (read,write), time synchronization, subscriptions, etc…
- **Cross-Platform Support**: Available for both Windows and Linux.

2 Common Use Case

These are some examples of common use cases for the CAS BACnet Cli. The tool is designed for use within any scripting languages such as bash, powershell, python, node, etc….

2.1 Unit and Regression testing

One of the primary uses of CAS BACnet CLI is for scripting unit and regression tests for BACnet server interfaces.

Unit and regression testing help identify bugs early, improve code quality, and ensure that new changes don't break existing functionality. These tests enhance development efficiency, reduce maintenance costs, and increase confidence in code changes.

Example simple write, read test scenario:

- Send **a write property** command to set a property (e.g., Present Value) of a device.
- Immediately perform a **read property** command to confirm that the value was set correctly.

Example commendable object write, read test scenario

- Send a **write property** command to commendable property (e.g., Present Value) with a priority of 8
- Send a **read property** command to confirm that the value matches the value at priority 8
- Send a **write property** command to commendable property (e.g., Present Value) with a priority of 12
- Send a **read property** command to confirm that the value matches the value at priority 8
- Send a **write property** command to commendable property (e.g., Present Value) with a priority of 8 and a value of NULL
- Send a **read property** command to confirm that the value matches the value at priority 12
- Send a **write property** command to commendable property (e.g., Present Value) with a priority of 12 and a value of NULL
- Send a **read property** command to confirm that the value matches the value for the relinquish default value

This kind of testing ensures that the system behaves correctly under different scenarios, and the automation of these tests simplifies the process.

2.2 Automation of BACnet Device Interactions

Automating interactions with devices is straightforward with CAS BACnet CLI. This can save time and improve reliability by reducing the need for manual intervention.

Example Script:

1. Write to set the motor state to "Stop".
2. Write to set the motor speed.
3. Write to set the motor direction.
4. Write to set the motor state to "Start".

This enables a single command to control the motor, eliminating the need for multiple manual write commands.

2.3 Logging Values Over Time

For monitoring and logging, a scheduled task (Windows) or cron job (Linux) can be used to periodically execute a script that reads values from BACnet devices and stores them in a CSV file or inserts them into a database. These logs can later be analyzed, converted into graphs, or used for tracking changes over time.

3 Supported Operating Systems

This tool should function on most Windows and Linux operating systems. Below is a table of the operating systems and versions that this tool has been tested on.

- Windows - Windows 10, 11
- Linux - Ubuntu 20, Ubuntu 22

4 Supported Output formats

The CAS BACnet Cli tool supports many output formats for different scenarios.

- XML - Machine readable format. Useful for scripting or webhooks.
- Values only - Only the values, useful for injection into other scripts or tools.
- Human readable - (Default) Full text description, useful for a user to read.

Example of human readable outputs

```
> CASBACnetCli whois

   Connection String  | SNET |      SADR | Device ID |  Vendor | Max APDU |  Seg |
----------------------*------*-----------*-----------*---------*----------*------*
   192.168.3.74:47808 |      |           |    389998 |     389 |     1476 |   No |
  192.168.3.102:47808 |      |           |       389 |      37 |     1458 |   No |
  192.168.3.163:47808 |      |           |   4194303 |     113 |     1476 |   No |
   192.168.1.77:47808 |   15 |    05ef89 |    389001 |     389 |     1476 |   No |
   192.168.1.77:47808 |   15 |    05ef90 |    389002 |     389 |     1476 |   No |
   192.168.1.77:47808 |   15 |    05ef91 |    389003 |     389 |     1476 |   No |

6 I-Am messages received in response to Who-IS message
```

Example of XML outputs

```
> CASBACnetCli whois -xml
```

```
<!-- CAS BACnet Stack v4.3.8.0 -->
<BACnetPacket networkType='IP'>
  <BVLL function='originalBroadcastNPDU' />
  <NPDU control='0x20' version='1'>
    <DestinationNetwork>65535</DestinationNetwork>
    <DestinationAddress length='0' />
    <HopCount>255</HopCount>
  </NPDU>
  <UnconfirmedRequestPDU serviceChoice='iAm'>
    <IAmRequest>
      <IAmDeviceIdentifier datatype='12' objectInstance='389998' objectType='8'>device,
```

```
389998</IAmDeviceIdentifier>
      <MaxAPDULengthAccepted datatype='2' value='1476'>1476</MaxAPDULengthAccepted>
      <SegmentationSupported datatype='9' value='3'>noSegmentation</SegmentationSupported>
      <VendorId datatype='2' value='389'>389</VendorId>
    </IAmRequest>
  </UnconfirmedRequestPDU>
</BACnetPacket>
```

5 Commands

This is a list of the CAS BACnet Cli commands available.

Usage:

```
CASBACnetCli <command> [<args>...]
CASBACnetCli help <command>
```

General Options:

```
--xml                          Prints the responses as XML instead of human
readable
--verbose                      Prints the responses in a human readable
format. Default
--value                        Prints the response's value only
--timeout=<time>               The APDU timeout in seconds [default: 3]
--port=<port>                  Destination UDP port [default: 47808]
--ip=<ipaddress>               Destination ip address.
--dnet=<destination-network>   Destination network on foreign network.
--dadr=<destination-address>   Destination address (in HEX) on foreign
network.
--deviceid=<device-id>         Searches the 'whois' table for --dent, --dadr,
--ip, --port
--clientDeviceID=<clientID>    The command line tool's device ID. [default:
389998]
```

5.1 Build Create Object

The “buildcreateobject” command records a file, combination of BACnetPropertyIdentifier and value. This command does not send any messages onto the network.

The “sendbuiltcreateobject” command is used to send the recorded combination created by “buildcreateobject” as a single “CreateObjectMessage”. After this command is successfully executed the recorded combinations created by “buildcreateobject” are cleared.

The reset command clears the recorded combination created by “buildcreateobject” command.

Note:

- For a list of the enumerations for <object-type> run
  - `CASBACnetCli enum object-type`
- For a list of the enumerations for <property-identifier> run
  - `CASBACnetCli enum property-identifier`
- For a list of the enumerations for <datatype> run
  - `CASBACnetCli enum datatype`

5.1.1 Usage

```
CASBACnetCli buildcreateobject <property-identifier> <value> [--
datatype=<datatype>]

CASBACnetCli sendbuiltcreateobject <object-type> [<object-instance>]
[options]

CASBACnetCli reset
```

5.1.2 Arguments

<object-type>

Object type combined with Object instance is part of the 'Object Identifier' parameter, of type BACnetObjectIdentifier.

<object-instance>

Object instance combined with Object type is part of the 'Object Identifier' parameter, of type BACnetObjectIdentifier.

<property-identifier>

This parameter, of type BACnetPropertyIdentifier, shall provide the means of uniquely identifying the property to be initialized on creation of this object.

<value>

This parameter in combination with the <property-identifier>, shall be used to initialize properties on the creation of this object.

--datatype=<datatype>

[Optional] Defines the data type of the <value> parameter.

5.1.3 Examples

The following series of commands build a series of combinations of BACnetPropertyIdentifier, and values. In this case a on creation of the analog-output (1) with an object instance of 800. The initial value of object-name (77), and present-value (85) wil be set.

```
CASBACnetCli buildcreateobject 77 "New analog output name" --datatype=7

CASBACnetCli buildcreateobject 85 99.6 --datatype=4 --priority=8

CASBACnetCli sendbuiltcreateobject 1 800
```

5.2 Build Read Property

The “buildreadproperty” command records to a file, combination of BACnetObjectIdentifier and BACnetPropertyIdentifier. This command does not send any messages onto the network.

The sendbuiltreadproperty command is used to send the recorded combination created by “buildreadproperty” as a single ReadPropertyMuliple message. After this command is successfully executed the recorded combination created by “buildreadproperty” is cleared.

The reset command clears the record combination created by “buildreadproperty” command.

Note:

- For a list of the enumerations for <object-type> run
  - `CASBACnetCli enum object-type`
- For a list of the enumerations for <property-identifier> run
  - `CASBACnetCli enum property-identifier`

5.2.1 Usage:

```
CASBACnetCli buildreadproperty  <object-type> <object-instance> <property-identifier>
                                [<property-array-index>] [options]
CASBACnetCli sendbuiltreadproperty [options]
CASBACnetCli reset
```

5.2.2 Arguments

<object-type>

Object type combined with Object instance is part of the 'Object Identifier' parameter, of type BACnetObjectIdentifier.

<object-instance>

Object instance combined with Object type is part of the 'Object Identifier' parameter, of type BACnetObjectIdentifier.

<property-identifier>

This parameter, of type BACnetPropertyIdentifier, shall provide the means of uniquely identifying the property to be read and returned by this service.

<property-array-index>

[Optional] If the property identified above is of datatype array, this optional parameter of type Unsigned shall indicate the array index of the element of the property referenced by this service. If the 'Property Array Index' is omitted, this shall mean that the entire array shall be referenced.

5.2.3 Examples

The following series of commands build a series of combinations of BACnetObjectIdentifier and BACnetPropertyIdentifier. In this case a request for object-name (77), status-flags (111), description (28) on a device (8) with an object instance of 389000. Then sends the Read property Multiple message with the previous series of combination of BACnetObjectIdentifier and BACnetPropertyIdentifier as a single message.

```
CASBACnetCli buildreadproperty 8 389000 77
CASBACnetCli buildreadproperty 8 389000 111
CASBACnetCli buildreadproperty 8 389000 28
CASBACnetCli sendbuiltreadproperty
```

5.3 Build Write Property

The “buildwriteproperty” command records a file, combination of BACnetObjectIdentifier, BACnetPropertyIdentifier and value. This command does not send any messages onto the network.

The sendbuiltwriteproperty command is used to send the recorded combination created by “buildwriteproperty” as a single WritePropertyMuliple message. After this command is successfully executed the recorded combination created by “buildwriteproperty” are cleared.

The reset command clears the recorded combination created by “buildwriteproperty” command.

Note:

- For a list of the enumerations for <object-type> run
  - `CASBACnetCli enum object-type`
- For a list of the enumerations for <property-identifier> run
  - `CASBACnetCli enum property-identifier`

5.3.1 Usage:

```
CASBACnetCli buildwriteproperty <object-type> <object-instance> <property-identifier> <value>
                                [--priority=<priority-level>] [--datatype=<datatype>] [options]

CASBACnetCli sendbuiltwriteproperty [options]

CASBACnetCli reset
```

5.3.2 Arguments

<object-type>

Object type combined with Object instance is part of the 'Object Identifier' parameter, of type BACnetObjectIdentifier.

<object-instance>

Object instance combined with Object type is part of the 'Object Identifier' parameter, of type BACnetObjectIdentifier.

<property-identifier>

This parameter, of type BACnetPropertyIdentifier, shall provide the means of uniquely identifying the property to be written to.

<value>

This parameter in combination with the <property-identifier>, shall be used to set the value of this object.

--priority=<priority-level>

--datatype=<datatype>

[Optional] Defines the data type of the <value> parameter.

5.3.3 Examples

The following series of commands build a series of combinations of BACnetObjectIdentifier, BACnetPropertyIdentifier, and values. In this case a write request for object-name (77), present-value (85) on an analog-output (1) with an object instance of 800. Then sends the write property multiple messages with the previous series of combination of BACnetObjectIdentifier BACnetPropertyIdentifier, and value as a single message.

```
CASBACnetCli buildwriteproperty 1 800 77 "New analog output name" --
datatype=7

CASBACnetCli buildwriteproperty 1 800 85 99.6 --datatype=4 --priority=8

CASBACnetCli sendbuiltwriteproperty
```

5.4 Cancel Subscribe COV

Send a Cancel Subscribe COV message to cancel an existing subscription. The <object-type> <object-instance> and <subscriber-process-identifier> of the previous subscription is required.

Note:

- For a list of the enumerations for <object-type> run "CASBACnetCli enum object-type"

5.4.1 Usage

```
CASBACnetCli cancelsubscribecov <object-type> <object-instance>

<subscriber-process-identifier> [options]
```

5.4.2 Arguments

<object-type>

Object type combined with Object instance is part of the 'Object Identifier' parameter, of type BACnetObjectIdentifier.

<object-instance>

Object instance combined with Object type is part of the 'Object Identifier' parameter, of type BACnetObjectIdentifier.

<subscriber-process-identifier>

This parameter, of type Unsigned32, shall convey a numeric "handle" meaningful to the subscriber. This handle shall be used to match future re-subscriptions and cancellations from the subscriber with the COV context that exists within the COVserver device and with confirmed or unconfirmed COV notifications to identify the process within the COV-client that should receive them.

5.4.3 Examples

Send a Cancel Subscribe COV message to cancel an existing subscription for an Analog input (0) with an object instance of 800 and a subscriber process identifier of 101.

```
CASBACnetCli cancelsubscribecov 0 800 101
```

5.5 Create Object

Sends a CreateObject message to create a new instance of an object. This service may be used to create instances of both standard and vendor specific objects.

Note:

- For a list of the enumerations for <object-type> run
  - `CASBACnetCli enum object-type`

5.5.1 Usage

```
CASBACnetCli createobject <object-type> [<object-instance>] [options]
```

5.5.2 Arguments

<object-type>

Object type combined with Object instance is part of the 'Object Identifier' parameter, of type BACnetObjectIdentifier.

<object-instance>

[Optional] Object instance combined with Object type is part of the 'Object Identifier' parameter, of type BACnetObjectIdentifier.

5.5.3 Examples

Sends a create object message asking a remote device to create an Analog input (0) object.

```
CASBACnetCli createobject 0
```

Sends a create object message asking a remote device to create a Analog input (0) object with a object instance of 800

```
CASBACnetCli createobject 0 800
```

5.6 Delete Object

Sends a DeleteObject message to a device to delete an existing object. Although this service is general in the sense that it can be applied to any object type, it is expected that most objects in a control system cannot be deleted by this service because they are protected as a security feature. There are some objects, however, that may be created and deleted dynamically. Group objects and Event Enrollment objects are examples.

Note:

- For a list of the enumerations for <object-type> run
  - `CASBACnetCli enum object-type`

5.6.1 Usage

```
CASBACnetCli deleteobject <object-type> <object-instance> [options]
```

5.6.2 Arguments

<object-type>

Object type combined with Object instance is part of the 'Object Identifier' parameter, of type BACnetObjectIdentifier.

<object-instance>

Object instance combined with Object type is part of the 'Object Identifier' parameter, of type BACnetObjectIdentifier.

5.6.3 Examples

Sends a delete object message to a device in an attempt to delete an analog input (0) with a object instance of 800

```
CASBACnetCli deleteobject 0 800
```

5.7 Enum

Prints the enumerated values for the parameters used by the CASBACnetCli application.

5.7.1 Usage

```
CASBACnetCli enum
CASBACnetCli enum <enum-name>
CASBACnetCli enum <enum-name> <enum-value>
```

5.7.2 Arguments

<enum-name>

[Optional] The name of the enumeration to print. Options include object-type,

property-identifier, datatype

<enum-value>

[Optional] The value of a defined enumeration to print. This is useful for

finding a single enumeration in a large list.

5.7.3 Examples

```
CASBACnetCli enum                          # Prints all enumerations
CASBACnetCli enum object-type              # Prints the object type
enumerations
CASBACnetCli enum object-type 14           # Prints the object type
enumeration with a value
                                           # of 14 (multiStateOutput)
CASBACnetCli enum property-identifier      # Prints all the property-
identifier enumerations
CASBACnetCli enum property-identifier 85   # Prints the object type
enumeration with a value
                                           # of 85 (presentValue)
```

5.8 Explore

Discover and create a report of a device

5.8.1 Usage

```
CASBACnetCli explore [<object-instance>] [options]
```

5.9 I Am

Sends a IAM message for the CASBACnetCli

5.9.1 Usage

```
CASBACnetCli iam [options]
```

5.9.2 Examples

```
CASBACnetCli iam  # Sends a Iam message onto the local network.
```

5.10 Read Property

Sends a read property request for a current value of an object's property. This service allows read access to any property of any object, whether a BACnet-defined object or not. If <property-identifier> parameter is one of the special property identifiers ALL (8), REQUIRED (105), or OPTIONAL (80) then the service will be ReadPropertyMultiple instead of ReadProperty

Note:

- For a list of the enumerations for <object-type> run
  - `CASBACnetCli enum object-type`
- For a list of the enumerations for <property-identifier> run
  - `CASBACnetCli enum property-identifier`

5.10.1 Usage

```
CASBACnetCli readproperty <object-type> <object-instance> <property-
identifier>
                          [<property-array-index>] [options]
```

5.10.2 Arguments

<object-type>

Object type combined with Object instance is part of the 'Object Identifier' parameter, of type BACnetObjectIdentifier.

<object-instance>

Object instance combined with Object type is part of the 'Object Identifier' parameter, of type BACnetObjectIdentifier.

<property-identifier>

This parameter, of type BACnetPropertyIdentifier, shall provide the means of uniquely identifying the property to be read and returned by this service.

<property-array-index>

[Optional] If the property identified above is of datatype array, this optional parameter of type Unsigned shall indicate the array index of the element of the property referenced by this service. If the 'Property Array Index' is omitted, this shall mean that the entire array shall be referenced.

5.10.3 Examples

Sends a read property request for the object-name (77) from a device (8) with a instance of 389000

```
CASBACnetCli readproperty 8 389000 77
```

Sends a read property request for the object-list (76) with an array index 19 from a device (8) with an instance of 389000.

```
CASBACnetCli readproperty 8 389000 76 19
```

Sends a read property request for the present-Value (85) from an analog input (0) with an instance of 800

```
CASBACnetCli readproperty 0 800 85
```

5.11 Read Range

The most commonly used read range types are :

- **all** - Sends a read range message for all values.
- **position** - based on position
- **sequencenumber** - based on sequencenumber
- **time** - based on time

See 'CASBACnetCli help readrange <type>' for more information on a specific read range type.

5.11.1 Usage

```
CASBACnetCli readrange <type> [<args>...]
CASBACnetCli help readrange <type>
```

5.12 Register Foreign Device

5.12.1 Usage

```
CASBACnetCli registerforeigndevice <lifetime> [options]
```

5.12.2 Arguments

<lifetime>

This parameter, of type Unsigned, shall convey the desired lifetime in seconds of

the foreign registration.

5.12.3 Examples

Sends a foreign device registration message with a lifetime of 300 seconds (5 mins) to a specific BACnet BBMD gateway at IP 192.168.1.25

```
CASBACnetCli registerforeigndevice 300 --ip=192.168.1.26
```

5.13 Reinitialize Device

5.13.1 Usage

```
CASBACnetCli reinitializedevice <state> [<password>] [options]
```

5.14 Reset

The “buildreadproperty” command records to a file, a combination of BACnetObjectIdentifier and BACnetPropertyIdentifier. This command does not send any messages onto the network.

The sendbuiltreadproperty command is used to send the recorded combination created by “buildreadproperty” as a single ReadPropertyMuliple message. After this command is successfully executed the recorded combination created by “buildreadproperty” are cleared.

The reset command clears the record combination created by the “buildreadproperty” command.

Note:

- For a list of the enumerations for <object-type> run
  - `CASBACnetCli enum object-type`
- For a list of the enumerations for <property-identifier> run
  - `CASBACnetCli enum property-identifier`

5.14.1 Usage

```
CASBACnetCli buildreadproperty  <object-type> <object-instance> <property-identifier>
                                [<property-array-index>] [options]
CASBACnetCli sendbuiltreadproperty [options]
CASBACnetCli reset
```

5.14.2 Arguments

<object-type>

Object type combined with Object instance is part of the 'Object Identifier' parameter, of type BACnetObjectIdentifier.

<object-instance>

Object instance combined with Object type is part of the 'Object Identifier' parameter, of type BACnetObjectIdentifier.

<property-identifier>

This parameter, of type BACnetPropertyIdentifier, shall provide the means of uniquely identifying the property to be read and returned by this service.

<property-array-index>

[Optional] If the property identified above is of datatype array, this optional parameter of type Unsigned shall indicate the array index of the element of the property referenced by this service. If the 'Property Array Index' is omitted, this shall mean that the entire array shall be referenced.

5.14.3 Examples

The following series of commands build a series of combinations of BACnetObjectIdentifier and BACnetPropertyIdentifier. In this case a request for object-name (77), status-flags (111), description (28) on a device (8) with an object instance of 389000. Then sends the Read property Multiple message with the previous series of combination of BACnetObjectIdentifier and BACnetPropertyIdentifier as a single message.

```
CASBACnetCli buildreadproperty 8 389000 77
CASBACnetCli buildreadproperty 8 389000 111
CASBACnetCli buildreadproperty 8 389000 28
CASBACnetCli sendbuiltreadproperty
```

5.15 Send Built Create Object

The “buildcreateobject” command records a file, combination of BACnetPropertyIdentifier and value. This command does not send any messages onto the network.

The sendbuiltcreateobject command is used to send the recorded combination created by “buildcreateobject” as a single CreateObjectmessage. After this command is successfully executed the recorded combinations created by “buildcreateobject” are cleared.

The reset command clears the recorded combination created by “buildcreateobject” command.

Note:

- For a list of the enumerations for <object-type> run
  - `CASBACnetCli enum object-type`
- For a list of the enumerations for <property-identifier> run
  - `CASBACnetCli enum property-identifier`
- For a list of the enumerations for <datatype> run
  - `CASBACnetCli enum datatype`

5.15.1 Usage

```
CASBACnetCli buildcreateobject <property-identifier> <value> [--
datatype=<datatype>]
CASBACnetCli sendbuiltcreateobject <object-type> [<object-instance>] [options]
CASBACnetCli reset
```

5.15.2 Arguments

<object-type>

Object type combined with Object instance is part of the 'Object Identifier' parameter, of type BACnetObjectIdentifier.

<object-instance>

Object instance combined with Object type is part of the 'Object Identifier' parameter, of type BACnetObjectIdentifier.

<property-identifier>

This parameter, of type BACnetPropertyIdentifier, shall provide the means of uniquely identifying the property to be initialized on creation of this object.

<value>

This parameter in combination with the <property-identifier>, shall be used to initialize properties on the creation of this object.

--datatype=<datatype>

[Optional] Defines the data type of the <value> parameter.

5.15.3 Examples

The following series of commands build a series of combinations of BACnetPropertyIdentifier, and values. In this case a on creation of the analog-output (1) with an object instance of 800. The initial value of object-name (77), and present-value (85) will be set.

```
CASBACnetCli buildcreateobject 77 "New analog output name" --datatype=7
CASBACnetCli buildcreateobject 85 99.6 --datatype=4 --priority=8
CASBACnetCli sendbuiltcreateobject 1 800
```

5.16 Send Built Read Property

The “buildreadproperty” command records to a file, a combination of BACnetObjectIdentifier and BACnetPropertyIdentifier. This command does not send any messages onto the network.

The sendbuiltreadproperty command is used to send the recorded combination created by “buildreadproperty” as a single ReadPropertyMuliple message. After this command is successfully executed the recorded combination created by “buildreadproperty” is cleared.

The reset command clears the record combination created by the “buildreadproperty” command.

Note:

- For a list of the enumerations for <object-type> run
  - `CASBACnetCli enum object-type`
- For a list of the enumerations for <property-identifier> run
  - `CASBACnetCli enum property-identifier`

5.16.1 Usage

```
CASBACnetCli buildreadproperty  <object-type> <object-instance> <property-identifier>
                                [<property-array-index>] [options]
CASBACnetCli sendbuiltreadproperty [options]
```

```
CASBACnetCli reset
```

5.16.2 Arguments

<object-type>

Object type combined with Object instance is part of the 'Object Identifier' parameter, of type BACnetObjectIdentifier.

<object-instance>

Object instance combined with Object type is part of the 'Object Identifier' parameter, of type BACnetObjectIdentifier.

<property-identifier>

This parameter, of type BACnetPropertyIdentifier, shall provide the means of uniquely identifying the property to be read and returned by this service.

<property-array-index>

[Optional] If the property identified above is of datatype array, this optional parameter of type Unsigned shall indicate the array index of the element of the property referenced by this service. If the 'Property Array Index' is omitted, this shall mean that the entire array shall be referenced.

5.16.3 Examples

The following series of commands build a series of combinations of BACnetObjectIdentifier and BACnetPropertyIdentifier. In this case a request for object-name (77), status-flags (111), description (28) on a device (8) with an object instance of 389000. Then sends the Read property Multiple message with the previous series of combination of BACnetObjectIdentifier and BACnetPropertyIdentifier as a single message.

```
CASBACnetCli buildreadproperty 8 389000 77
CASBACnetCli buildreadproperty 8 389000 111
CASBACnetCli buildreadproperty 8 389000 28
CASBACnetCli sendbuiltreadproperty
```

5.17 Send Built Write Property

The “buildwriteproperty” command records a file, combination of BACnetObjectIdentifier, BACnetPropertyIdentifier and value. This command does not send any messages onto the network.

The sendbuiltwriteproperty command is used to send the recorded combination created by “buildwriteproperty” as a single WritePropertyMuliple message. After this command is successfully executed the recorded combination created by “buildwriteproperty” is cleared.

The reset command clears the recorded combination created by “buildwriteproperty” command.

Note:

- For a list of the enumerations for <object-type> run
  - `CASBACnetCli enum object-type`
- For a list of the enumerations for <property-identifier> run
  - `CASBACnetCli enum property-identifier`

5.17.1 Usage

```
CASBACnetCli buildwriteproperty <object-type> <object-instance> <property-identifier> <value>
                                [--priority=<priority-level>] [--datatype=<datatype>] [options]
CASBACnetCli sendbuiltwriteproperty [options]
CASBACnetCli reset
```

5.17.2 Arguments

<object-type>

Object type combined with Object instance is part of the 'Object Identifier' parameter, of type BACnetObjectIdentifier.

<object-instance>

Object instance combined with Object type is part of the 'Object Identifier' parameter, of type BACnetObjectIdentifier.

<property-identifier>

This parameter, of type BACnetPropertyIdentifier, shall provide the means of uniquely identifying the property to be written to.

<value>

This parameter in combination with the <property-identifier>, shall be used to set the value of this object.

--priority=<priority-level>

--datatype=<datatype>

[Optional] Defines the data type of the <value> parameter.

5.17.3 Examples

The following series of commands build a series of combinations of BACnetObjectIdentifier, BACnetPropertyIdentifier, and values. In this case a write request for object-name (77), present-value (85) on an analog-output (1) with an object instance of 800. Then sends the write property multiple messages with the previous series of combination of BACnetObjectIdentifier BACnetPropertyIdentifier, and value as a single message.

```
CASBACnetCli buildwriteproperty 1 800 77 "New analog output name" --datatype=7
CASBACnetCli buildwriteproperty 1 800 85 99.6 --datatype=4 --priority=8
CASBACnetCli sendbuiltwriteproperty
```

5.18 Subscribe COV

Sends a SubscribeCOV message to subscribe for the receipt of notifications of changes that may occur to the properties of a particular object. Certain BACnet standard objects may optionally support COV reporting. If a standard object provides COV reporting, then changes of value of specific properties of the object, in some cases based on programmable increments, trigger COV notifications to be sent to one or more subscriber clients.

Note:

- For a list of the enumerations for <object-type> run
  - `CASBACnetCli enum object-type`

5.18.1 Usage

```
CASBACnetCli subscribecov <object-type> <object-instance> <lifetime>
                          <subscriber-process-identifier> [--enable-
confirmed-notifications]
                          [options]
```

5.18.2 Arguments

<object-type>

Object type combined with Object instance is part of the 'Object Identifier' parameter, of type BACnetObjectIdentifier.

<object-instance>

Object instance combined with Object type is part of the 'Object Identifier' parameter, of type BACnetObjectIdentifier.

<lifetime>

This parameter, of type Unsigned, shall convey the desired lifetime of the subscription in seconds. A value of zero shall indicate an indefinite lifetime, without automatic cancellation. A non-zero value shall indicate the number of seconds that may elapse before the subscription shall be automatically cancelled.

<subscriber-process-identifier>

This parameter, of type Unsigned32, shall convey a numeric "handle" meaningful to the subscriber. This handle shall be used to match future re-subscriptions and cancellations from the subscriber with the COV context that exists within the COVserver device and with confirmed or unconfirmed COV notifications to identify the process within the COV-client that should receive them.

--enable-confirmed-notifications

[Optional] This parameter is a flag, shall convey whether the COV-server device shall issue ConfirmedCOVNotifications (TRUE) or UnconfirmedCOVNotifications (FALSE) when changes occur.

5.18.3 Examples

Send a Subscribe COV message for an Analog input (0) with an object instance of 800 and a lifetime of 300 seconds (5 mins). The subscriber process identifier is 101.

```
CASBACnetCli subscribecov 0 800 300 101
```

Send a Subscribe COV message for an Analog input (0) with an object instance of 800 and a lifetime of 300 seconds (5 mins). The subscriber process identifier is 101. The responding device uses ConfirmedCOVNotifications instead of UnconfirmedCOVNotifications.

```
CASBACnetCli subscribecov 0 800 300 101 --enable-confirmed-notifications
```

5.19 Subscribe COV Property

Sends a SubscribeCOVProperty message to subscribe for the receipt of notifications of changes that may occur to the properties of a particular object. Certain BACnet standard objects may optionally support COV reporting. If a standard object provides COV reporting, then changes of value of specific properties of the object, in some cases based on programmable increments, trigger COV notifications to be sent to one or more subscriber clients.

Note:

- For a list of the enumerations for <object-type> run
  - `CASBACnetCli enum object-type`

5.19.1 Usage

```
CASBACnetCli subscribecovproperty <object-type> <object-instance> <property-identifier>
                                  <lifetime> <subscriber-process-identifier>
                                  [--enable-confirmed-notifications]
[options]
```

5.19.2 Arguments

<object-type>

Object type combined with Object instance is part of the 'Object Identifier' parameter, of type BACnetObjectIdentifier.

<object-instance>

Object instance combined with Object type is part of the 'Object Identifier' parameter, of type BACnetObjectIdentifier.

<property-identifier>

This parameter, of type BACnetPropertyIdentifier, shall provide the means of uniquely identifying the property to be subscribed to.

<lifetime>

This parameter, of type Unsigned, shall convey the desired lifetime of the subscription in seconds. A value of zero shall indicate an indefinite lifetime, without automatic cancellation. A non-zero value shall indicate the number of seconds that may elapse before the subscription shall be automatically cancelled.

<subscriber-process-identifier>

This parameter, of type Unsigned32, shall convey a numeric "handle" meaningful to the subscriber. This handle shall be used to match future re-subscriptions and cancellations from the subscriber with the COV context that exists within the COVserver device and with confirmed or unconfirmed COV notifications to identify the process within the COV-client that should receive them.

--enable-confirmed-notifications

[Optional] This parameter is a flag, shall convey whether the COV-server device shall issue ConfirmedCOVNotifications (TRUE) or UnconfirmedCOVNotifications (FALSE) when changes occur.

5.19.3 Examples

Send a Subscribe COV Property message for an Analog input (0) with an object instance of 800 and a lifetime of 300 seconds (5 mins). The subscriber process identifier is 101.

```
CASBACnetCli subscribecovproperty 0 800 85 300 101
```

Send a Subscribe COV Property message for an Analog input (0) with an object instance of 800 and a lifetime of 300 seconds (5 mins). The subscriber process identifier is 101. The responding device uses ConfirmedCOVNotifications instead of UnconfirmedCOVNotifications.

```
CASBACnetCli subscribecovproperty 0 800 85 300 101 --enable-confirmed-
notifications
```

5.20 Time Synch

Sends a time synchronization or UTC time synchronization message to a device. if no parameters are defined then local computer time is used. If only some of the parameters are defined then defaults will be used for the remaining parameters.

5.20.1 Usage

```
CASBACnetCli timesynch [options]

CASBACnetCli timesynch <year> <month> <day> <weekday> <hour> <minute>
<second>
                        <hundrethSeconds> [options]

CASBACnetCli utctimesynch [options]

CASBACnetCli utctimesynch <year> <month> <day> <weekday> <hour> <minute>
<second>
                           <hundrethSeconds> [options]
```

5.20.2 Examples

Sends a time synchronization message using the computer's local date/time.

```
CASBACnetCli timesynch
```

Sends a time synchronization message with the date/time set to 2018-Oct-03 2:30:00.0 pm

```
CASBACnetCli timesych 2018 11 03 3 14 30 0 0
```

5.21 UTC Time Synch

Sends a time synchronization or UTC time synchronization message to a device. if no parameters are defined then local computer time is used. If only some of the parameters are defined then defaults will be used for the remaining parameters.

5.21.1 Usage

```
CASBACnetCli timesynch [options]

CASBACnetCli timesynch <year> <month> <day> <weekday> <hour> <minute> <second>
<hundrethSeconds>

                        [options]

CASBACnetCli utctimesynch [options]

CASBACnetCli utctimesynch <year> <month> <day> <weekday> <hour> <minute> <second>
<hundrethSeconds>

                           [options]
```

5.21.2 Examples

Sends a time synchronization message using the computer's local date/time.

```
CASBACnetCli timesynch
```

Sends a time synchronization message with the date/time set to 2018-Oct-03 2:30:00.0 pm

```
CASBACnetCli timesych 2018 11 03 3 14 30 0 0
```

5.22 Who Has

Sends a Who-has message onto the network. This message is used to determine the device object identifier, the network address, vendor ID, and MAX APDU accepted given Object_Name or a given Object_Identifier. A I-Have message is sent in response by a device that matches the search criteria in a Who-has message.

Note:

- For a list of the enumerations for <object-type> run
  - `CASBACnetCli enum object-type`

5.22.1 Usage

```
CASBACnetCli whohas <object-name> [options]

CASBACnetCli whohas <object-type> <object-instance> [options]
```

5.22.2 Arguments

<object-name>

The 'Object Name' parameter, of type CharacterString, shall convey the value of the Object_Name property of the object that is to be located. If the 'Object Name' parameter is present, then only those devices that contain an object with an Object_Name property value matching the 'Object Name' parameter shall respond with an I-Have response.

<object-type>

Object type is part of the The 'Object Identifier' parameter, of type BACnetObjectIdentifier, shall convey the Object_Identifier of the object that is to be located. If the 'Object Identifier' parameter is present, then only those devices that contain an object with an Object_Identifier property value matching the 'Object Identifier' parameter shall respond with an I-Have response.

<object-instance>

Object instance is part of the The 'Object Identifier' parameter of type BACnetObjectIdentifier. See <object-type> for more information.

5.22.3 Examples

Search for devices with objects that have the Object name of "Current temperature"

```
CASBACnetCli whohas "Current temperature"
```

Search for a device (8) with an instance number of 389001

```
CASBACnetCli whohas 8 389001
```

Search for a device with an analog input (0) with with an instance number of 801

```
CASBACnetCli whohas 0 801
```

5.23 Who Is

Sends a Who-is message onto the network. This message is used to determine the device object identifier, the network address, vendors ID, and MAX APDU accepted. The Who-is message can optionally be restricted only discover devices with a device object identifier within a certain range using the [<low-limit>] [<high-limit>] optional parameters.

Note:

- If <low-limit> is defined but <high-limit> is not defined, then <high-limit> will equal to <low-limit>

5.23.1 Usage

```
CASBACnetCli whois [<low-limit>] [<high-limit>] [options]
```

5.23.2 Arguments

<low-limit>

[Optional] Device Instance Range Low Limit. This parameter is an unsigned integer in the range 0 - 4194303. In conjunction with the 'Device Instance Range High Limit' parameter, it defines the devices that are qualified to respond with an I-Am service request. Device Object_Identifier instance number falls within the range greater than or equal to 'Device Instance Range Low Limit' and less than or equal to 'Device Instance Range High Limit' shall be qualified to respond.

<high-limit>

[Optional] Device Instance Range High Limit. This parameter is an unsigned integer in the range 0 - 4194303. In conjunction with the 'Device Instance Range Low Limit' parameter, it defines the devices that are qualified to respond with an I-Am service request. See <low-limit> for more information.

5.23.3 Examples

Sends a whois discovery message on the network for all devices regardless of their device object identifier

```
CASBACnetCli whois
```

Sends a whois discovery message on the network for a device with device object instance of 389000. This is equivalent to calling "CASBACnetCli whois 389000 389000"

```
CASBACnetCli whois 389000
```

Sends a ranged whois discovery message on the network looking for devices with device instances between 1000 and 2000

```
CASBACnetCli whois 1000 2000
```

5.24 Write Property

The WriteProperty service is used by a client BACnet-user to modify the value of a specified property of a BACnet object. This service potentially allows write access to any property of any object, whether a BACnet-defined object or not. Some implementers may wish to restrict write access to certain properties of certain objects. In such cases, an attempt to modify a restricted property shall result in the return of an error of 'Error Class' PROPERTY and 'Error Code' WRITE_ACCESS_DENIED.

Note:

- For a list of the enumerations for <object-type> run
  - `CASBACnetCli enum object-type`
- For a list of the enumerations for <property-identifier> run
  - `CASBACnetCli enum property-identifier`
- For a list of the enumerations for <datatype> run
  - `CASBACnetCli enum datatype`

5.24.1 Usage

```
CASBACnetCli writeproperty <object-type> <object-instance> <property-
identifier> <value>
                            [--datatype=<datatype>] [--priority=<priority-
level>] [options]
```

5.24.2 Arguments

<object-type>

Object type combined with Object instance is part of the 'Object Identifier' parameter, of type BACnetObjectIdentifier.

<object-instance>

Object instance combined with Object type is part of the 'Object Identifier' parameter, of type BACnetObjectIdentifier.

<property-identifier>

This parameter, of type BACnetPropertyIdentifier, shall provide the means of uniquely identifying the property to be written by this service. Because this service is intended to write a single property of a single object, the value of this parameter shall not be one of the special property identifiers ALL (8), REQUIRED (105), or, OPTIONAL(80)

<value>

If access to the specified property of the specified object is successful, this parameter shall be used to replace the value of the property at the time of access. It shall be of any datatype that is valid for the property being modified. If the datatype parameter is not defined then the datatype will attempt to be discovered.

--datatype=<datatype>

[Optional] Defines the data type of the <value> parameter.

--priority=<priority-level>

[Optional] This parameter shall be an integer in the range 1-16, which indicates the priority assigned to this write operation. If an attempt is made to write to a commendable property without specifying the priority, a default priority of 16 (the lowest priority) shall be assumed. If an attempt is made to write to a property that is not commendable with a specified priority, the priority shall be ignored.

5.24.3 Examples

Sends a write property message to update the present value (85) to 99.6 of a analogInput (0) with an object instance of 800. The datatype is assumed to be Real (4) and the priority is defaulted to the lowest value of 16.

```
CASBACnetCli writeproperty 0 800 85 99.6
```

Sends a write property message to update the present value (85) to 99.6 of a analogInput (0) with an object instance of 800. The datatype is set to Real (4) and the priority is defaulted to the lowest value of 16.

```
CASBACnetCli writeproperty 0 800 85 99.6 --datatype=4
```

Sends a write property message to update the present value (85) to 99.6 of a analogInput (0) with an object instance of 800. The datatype is set to Real (4) and the priority is set to 8

```
CASBACnetCli writeproperty 0 800 85 99.6 --datatype=4 --priority=8
```

Sends a write property message to update the object name (77) to "new name" of a device (8) with an object instance of 389000. The datatype is set to CharacterString (4)

```
CASBACnetCli writeproperty 8 389000 77 "new name" --datatype=7
```

6 Revision History

This table summarizes the update history for this document. Please contact Chipkin for an updated version of this document if required.

DATE	RESP	DOC. REV.	COMMENT
2025-Feb-26		1	Document created