



Morse Micro
reaching farther™

Centralized Authentication Control

APPNOTE-25

Copyright © 2025

Table of Contents

1 Scope	3
2 Theory of Operation	4
3 How to Configure	5
Enabling CAC on an AP	5
Enabling CAC on stations	5
Further CAC configuration (AP)	5
Default configuration	6
Sample configuration for a bandwidth-constrained network	6
Considerations	7
Thin LMAC	7
4 How to Verify	8
AP	8
AP-side log messages	8
Station	9
5 Revision History	10

1 Scope

Centralised Authentication Control (CAC) can be used to reduce the time taken for a large number of stations to associate, by throttling the number of stations that attempt to associate simultaneously, and thereby minimizing contention of time-sensitive association and authentication frame exchanges.

CAC is recommended for use in moderate (more than 40 stations) to large HaLow networks, particularly when operating in low capacity environments, for example in regulatory domains that only allow 1 MHz or 2 MHz bandwidth, or that have duty cycle limitations.

2 Theory of Operation

The following explanation is taken from *Wi-Fi HaLow for the Internet of Things: An Up-to-date Survey on 802.11ah*.

In Centralized Authentication Control (CAC), the AP dynamically changes the portion of stations that are allowed to send AuthReq messages. Specifically, the AP sets a threshold and broadcasts it to all stations by sending beacon frames. The beacon frame is a management frame and contains the information about the network, it is transmitted periodically by the AP to announce the presence of a wireless network and to synchronize all the stations of the network. When a station is initialized, it generates a random value from the interval $[0, 1022]$, and tries to send an AuthReq to the AP if the random value is smaller than the threshold obtained from the received beacon.

Otherwise, it postpones authentication/association until the next beacon arrives. The threshold should be adjusted dynamically by the AP to limit the number of stations accessing the channel in one beacon interval, and make sure all stations can associate as fast as possible.

3 How to Configure

Enabling CAC on an AP

Centralised Authentication Control can reduce the time taken for a large number of stations to associate, by throttling the number of stations that attempt to associate simultaneously, and thereby minimizing contention of time-sensitive association and authentication frame exchanges.

To enable CAC, add **cac '1'** to the **wifi-iface** section.

Enabling CAC on stations

CAC is enabled on stations by adding **cac=1** to the network block in **wpa_supplicant_s1g.conf**.

```
...
network={
    ssid=<SSID>
    ...
    cac=1
}
```

Any stations that do not have CAC configured will be able to associate with an AP that has CAC enabled, and will not be constrained by CAC restrictions imposed by the AP.

Further CAC configuration (AP)

If over-the-air capacity is limited, such as when using 1MHz bandwidth and/or a low duty cycle (for example, in JP, EU and IN regulatory domains), it may also be necessary to modify the CAC thresholds, by using the **morse_cli cac** command.

```
$ morse_cli cac --help
cac [-v] [{get|set|enable|disable}] [-d <ARFS>,<percent>]...
    [-i <ARFS>,<percent>]...
Configure Centralized Authentication Control
{get|set|enable|disable}    Subcommand
    get - get configured rules
    set - set rules (default)
    enable|disable - for internal use by wpa_supplicant only
-d, --decrease=<ARFS>,<percent>
    Auth Req Frames per Sec above which to decrease threshold by <percent>
    Decrease rules must be specified in descending ARFS order
    (match highest first)
-i, --increase=<ARFS>,<percent>
    Auth Req Frames per Sec below which to increase threshold by <percent>
    Increase rules must be specified in ascending ARFS order
    (match lowest first)
    Up to 8 decrease or increase rules can be configured
-v, --verbose                Verbose output
```

Default configuration

Use the `morse_cli cac get` command to see the current configuration. These are the default settings.

```
root@mazda1:~$ morse_cli cac get
When ARFS is greater than 16, decrease threshold by 25%
When ARFS is greater than 12, decrease threshold by 12%
When ARFS is greater than 10, decrease threshold by 6%
When ARFS is less than 4, increase threshold by 25%
When ARFS is less than 6, increase threshold by 12%
When ARFS is less than 8, increase threshold by 6%
```

Sample configuration for a bandwidth-constrained network

These are more aggressive settings, which may be suitable for a large or bandwidth-constrained network.

```
root@mazda1:~$ morse_cli cac -v -d5,60 -d4,50 -d3,40 -i2,12 -i3,6
When ARFS is greater than 5, decrease threshold by 60%
When ARFS is greater than 4, decrease threshold by 50%
When ARFS is greater than 3, decrease threshold by 40%
When ARFS is less than 2, increase threshold by 12%
```

When ARFS is less than 3, increase threshold by 6%

Considerations

Thin LMAC

Thin LMAC is required for associating more than 200 stations, and is described in *MM APPNOTE 20*. That document also contains useful information for improving performance in moderate to large scale networks.

4 How to Verify

AP

Confirm CAC is enabled by using the following command. A value of 1 indicates that CAC is enabled.

```
# cat /sys/module/morse/parameters/enable_cac  
1
```

AP-side log messages

Info level (6) or debug level (7) driver debugging can be enabled with the following commands.

```
# phy=$(ls -d /sys/kernel/debug/ieee80211/phy*)  
# echo 6 >$phy/morse/logging/cac
```

or

```
# echo 7 >$phy/morse/logging/cac
```

Whenever the CAC threshold changes (i.e. when many stations attempt to associate at the same time, up until most stations have associated and a threshold is no longer being set), messages such as the following are written to the AP's syslog.

```
# dmesg|grep -w CAC  
[ 205.593166] morse_sdio mmc1:0001:2: CAC ARFS=13 period=600 adjust=-2  
      idx=14 threshold=896  
[ 206.523164] morse_sdio mmc1:0001:2: CAC ARFS=11 period=800 adjust=-1  
      idx=13 threshold=832  
[ 206.963164] morse_sdio mmc1:0001:2: CAC ARFS=11 period=400 adjust=-1  
      idx=12 threshold=768  
[ 208.073159] morse_sdio mmc1:0001:2: CAC ARFS=2 period=1000 adjust=4  
      idx=16 threshold=1024
```

The fields in these messages are:

- **ARFS**: authentication request frames received in this period

- **period:** period over which the ARFS were received (max will be 1 second)
- **adjust:** change to the threshold index
- **idx:** new index, which is used to set the new threshold
- **threshold:** threshold now being advertised in beacons and probe responses

Station

When CAC is enabled on both the AP and STAs, and the AP starts advertising a CAC threshold (the above log messages are seen), messages like the following can be seen in the **wpa_supplicant** log if INFO level debug messages are enabled.

```
SVT_mercedes27_10.53.53.37# grep CAC /var/log/morse_wpa_supplicant.log
202.791642: wlan1: Auth disallowed by CAC for 0c:bf:74:59:bf:11,
           threshold=896 random=936
205.889910: wlan1: Auth allowed by CAC for 0c:bf:74:59:bf:11,
           threshold=1023 random=936
```

The fields in these messages are:

- **threshold:** threshold from the most recently received beacon or probe response
- **random:** the current random value used by this STA

5 Revision History

Release Number	Release Date	Release Notes
01	2024-03-18	Initial version



Morse Micro Pty. Ltd. - HQ

Level 8, 10-14 Waterloo Street,
Surry Hills, NSW 2010, AUSTRALIA

Morse Micro Inc. – USA

40 Waterworks Way
Irvine, CA 92618, UNITED STATES

Office Locations:

Bangalore, India
Cambridge, UK
Hangzhou, China
Irvine, CA - USA
Picton, NSW - Australia
Portola Valley, CA - USA
Shenzhen, China
Taipei, Taiwan

sales@morsemicro.com

www.morsemicro.com

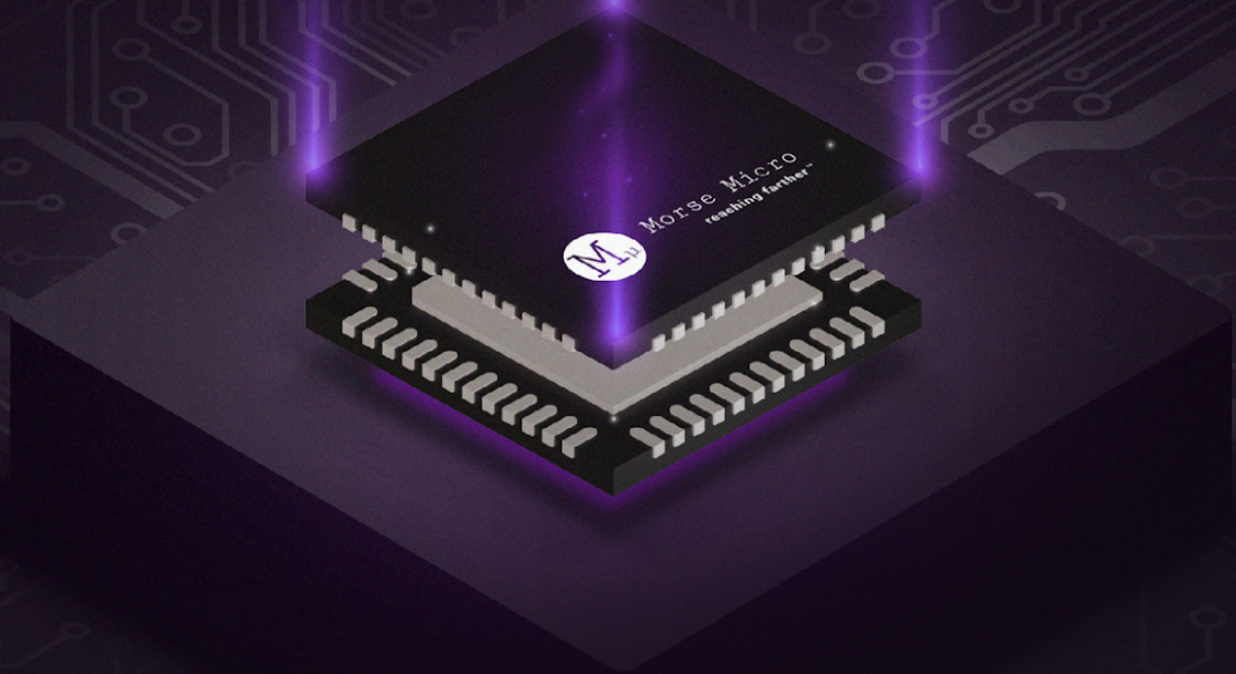
© 2016–2025 Morse Micro Pty. Ltd.

All rights reserved. Morse Micro and the Morse Micro logo are trademarks of Morse Micro Pty. Ltd. All other trademarks and service marks are the property of their respective owners.

Morse Micro makes no warranty, representation, or guarantee regarding the information contained herein or the suitability of its products and services for any particular purpose, nor does Morse Micro assume any liability whatsoever arising out of the application or use of any product or circuit. The products sold hereunder and any other products sold by Morse Micro have been subject to limited testing and should not be used in conjunction with mission-critical equipment or applications. Any performance specifications are believed to be reliable but are not verified, and Buyer must conduct and complete all performance and other testing of the products, alone and together with, or installed in, any end-products. Buyer shall not rely on any data and performance specifications or parameters provided by Morse Micro. It is the Buyer's responsibility to independently determine suitability of any products and to test and verify the same. The information provided by Morse Micro hereunder is provided "as is, where is" and with all faults, and the entire risk associated with such information is entirely with the Buyer. Morse Micro does not grant, explicitly or implicitly, to any party any patent rights, licenses, or any other IP rights, whether with regard to such information itself or anything described by such information. Information provided in this document is proprietary to Morse Micro, and Morse Micro reserves the right to make any changes to the information in this document or to any products and services at any time without notice.

About Morse Micro

Morse Micro is a leading Wi-Fi HaLow fabless semiconductor company based in Sydney, with global offices. As the world's premier Wi-Fi HaLow company, we pioneer next-gen IoT wireless connectivity solutions. Morse Micro is now sampling its Wi-Fi CERTIFIED HaLow MM6108 production silicon: the fastest, smallest, lowest power and longest-range Wi-Fi HaLow chip available in the market.



Morse Micro
reaching farther™