ArubaOS 7.4.1.3

Release Notes

a Hewlett Packard
Enterprise company

Copyright Information

Open Source Code

This product includes code licensed under the GNU General Public License, the GNU Lesser General Public License, and/or certain other open source licenses. A complete machine-readable copy of the source code corresponding to such code is available upon request. This offer is valid to anyone in receipt of this information and shall expire three years following the date of the final distribution of this product version by Hewlett-Packard Enterprise Company. To obtain such source code, send a check or money order in the amount of US $10.00 to:

Hewlett Packard Enterprise Company

Attn: General Counsel

3000 Hanover Street

Palo Alto, CA 94304

USA

Please specify the product and version for which you are requesting source code. You may also request a copy of this source code free of charge at dl-gplquery@arubanetworks.com.

Chapter 1
Release Overview

ArubaOS 7.4.1.3 is a patch release that introduces new features/enhancements, fixes to issues identified in the previous ArubaOS releases, and outstanding known issues and limitations in the current release. For details on all the features supported on Mobility Access Switch, see the Related Documents section.

This release note contains the following chapters:

- What's New in this Release on page 9 describes the fixes, known issues, and enhancements introduced in this release.
- Upgrade Procedures on page 59 covers the procedures for upgrading a Mobility Access Switch to ArubaOS 7.4.1.3.

Supported Browsers

The following browsers are officially supported for use with the ArubaOS 7.4.1.3 Web User Interface (WebUI):

- Microsoft Internet Explorer 9.x and 10.x on Windows XP, Windows Vista, Windows 7, and Windows 8
- Mozilla Firefox 17 or higher on Windows XP, Windows Vista, Windows 7, and Mac OS
- Apple Safari 5.1.7 or higher on Mac OS

Related Documents

The following documents are part of the complete documentation suite for the Aruba Mobility Access Switch:

- *ArubaOS 7.4 User Guide*
- *ArubaOS 7.4 Command Line Reference Guide*
- *ArubaOS 7.4 Quick Start Guide*
- *Aruba S3500 Series Mobility Access Switch Installation Guide*
- *Aruba S2500 Series Mobility Access Switch Installation Guide*
- *Aruba S1500 Series Mobility Access Switch Installation Guide*

Contacting Support

Table 1: *Contact Information*

Web Site Support	
Main Site	http://www.arubanetworks.com
Support Site	http://support.arubanetworks.com
Airheads Social Forums and Knowledge Base	https://community.arubanetworks.com
North American Telephone	1-800-943-4526 (Toll Free) 1-408-754-1200
International Telephone	http://www.arubanetworks.com/support-services/support-program/contact-support
Software Licensing Site	https://licensing.arubanetworks.com/
End-of-Support Information	http://www.arubanetworks.com/support-services/end-of-life-products/
Security Incident Response Team (SIRT)	Site: http://www.arubanetworks.com/support-services/security-bulletins/ Email: sirt@arubanetworks.com

New Features and Enhancements

New features in the following categories are introduced in ArubaOS 7.4:

- Layer 2 and Layer 3 Features on page 9
- Management Features on page 16
- Security Enhancements on page 25
- QoS Enhancements on page 32
- Branch Features on page 32
- Access Point Integration Features on page 33
- Platform Features on page 34
- WebUI Enhancements

Layer 2 and Layer 3 Features

This release of ArubaOS provides support for the following Layer 2 and Layer 3 features and enhancements:

Auto Link Aggregation Control Protocol

Starting from ArubaOS 7.4.1.1, the Mobility Access Switch supports Auto Link Aggregation Control Protocol (Auto-LACP). Based on device-group configuration, the Auto-LACP forms port-channels automatically. It helps automatic detection of the neighboring devices with port-channels when AP is connected to the Mobility Access Switch.

Auto-LACP is disabled by default on the Mobility Access Switch. You can enable Auto-LACP on the Mobility Access Switch using the CLI.

APs, IAPs, and ArubaOS Versions Supporting Auto-LACP

The following table lists the supporting access points and Mobility Access Switch- ArubaOS versions categorized by controller version.

Table 2: *APs, IAPs, and ArubaOS versions supporting Auto-LACP*

Controller Version	Supported APs	Mobility Access Switch - ArubaOS Version
6.4.4.0	AP-224, AP-225, AP-274, AP-275, AP-325	7.4.1.1
6.4.3.1	AP-224, AP-225, AP-274, AP-275	7.4.1.1
6.4.3.4-4.2.1.0	IAP-225	7.4.1.1

Important Points to Remember

The following are the important points to remember regarding Auto-LACP:

- Auto-LACP is supported only for AP-224, AP-225, AP-274, AP-275, AP-325, and IAP-225 access points.
- Auto-LACP is supported on S1500, S2500/S3500 Mobility Access Switch platforms.
- All Auto-LACP port-channels share the same device group configuration for AP device type.
- Configuration is blocked for Auto-LACP port-channels and port members.

- Alteration and deletion in the Auto-LACP profile is blocked.
- The maximum number of supported port-channels on S1500 and S2500/S3500 is eight and sixty four, respectively. Auto- LACP uses the LAG IDs from these limits only.
- Auto-LACP is functional only if **device-group** configuration is enabled.

Configuring Auto-LACP

To configure Auto-LACP, you should enable device configuration and Auto-LACP on the Mobility Access Switch.

To enable device configuration, execute the following commands:

```
(host) (config) #device-group ap
(host) (device-group access-point) #enable
```

To enable Auto-LACP, execute the following commands:

```
(host) (config) #device-group ap
(host) (device-group access-point) #auto-lacp
```

Verifying Auto-LACP

To verify the Auto-LACP configuration on the Mobility Access Switch, execute the following command:

```
(host) # show device-group-config ap
device-group access-point (N/A)
-------------------------------
Parameter                         Value
---------                         -----
Enable Device Config             true
Enable Auto LACP                 true
Interface MSTP Profile            default
Interface GVRP Profile            N/A
Interface PVST Profile            default
Interface LLDP Profile            device-group-default
Interface PoE Profile             device-group-default
Interface Ethernet Link Profile   default
Interface QoS Profile             N/A
Interface Policer Profile         N/A
Interface AAA Profile             N/A
Interfaces To Shutdown            N/A
Interface MTU                     1514
Interface Ingress ACL             N/A
Interface Egress ACL              N/A
Interface Session ACL             N/A
Interface QoS Trust Mode          auto
Interface Switching Profile       default
Interface Security Profile        N/A
Interface Trusted Mode            Trusted
```

NOTE

The **Enable Device Config** and **Enable Auto LACP** parameters display a true value.

To view Auto-LACP port-channel interfaces, execute the following command:

```
(host) # show interface port-channel auto-lacp
port-channel 1 is administratively Up, Link is Up, Line protocol is Up
Hardware is Port-Channel, LACP enabled, Address is 00:0b:86:6a:bd:80
Description: Link Aggregate
Created by Auto-LACP Link Aggregate
Member port(s):
GE1/0/16 is administratively Up, Link is Up, Line protocol is Up
GE2/0/16 is administratively Up, Link is Up, Line protocol is Up
Speed: 2 Gbps
Interface index: 1442
```

```
MTU 1514 bytes
Flags: Access, Trusted
Link status last changed: 0d 00h:27m:32s ago
Last clearing of counters: 0d 00h:27m:32s ago
Statistics:
Received 15172 frames, 3183154 octets
5 pps, 7.979 Kbps
7 broadcasts, 0 runts, 10 giants, 0 throttles
15220 error octets, 0 CRC frames
3410 multicast, 11755 unicast
Transmitted 6605 frames, 721121 octets
1 pps, 844 bps
1867 broadcasts, 0 throttles
0 errors octets, 0 deferred
0 collisions, 0 late collisions
GE1/0/16:
Statistics:
Received 13470 frames, 2960782 octets
5 pps, 6.966 Kbps
2 broadcasts, 0 runts, 10 giants, 0 throttles
15220 error octets, 0 CRC frames
1713 multicast, 11755 unicast
Transmitted 2265 frames, 190374 octets
0 pps, 0 bps
1867 broadcasts, 0 throttles
0 errors octets, 0 deferred
0 collisions, 0 late collisions
GE2/0/16:
Statistics:
Received 1714 frames, 224436 octets
0 pps, 1.013 Kbps
7 broadcasts, 0 runts, 0 giants, 0 throttles
0 error octets, 0 CRC frames
1707 multicast, 0 unicast
Transmitted 4372 frames, 534065 octets
1 pps, 844 bps
0 broadcasts, 0 throttles
0 errors octets, 0 deferred
0 collisions, 0 late collisions
```

To view if a gigabit Ethernet port is a member of a port-channel, execute the following command:

```
(host) # show interface gigabitethernet 1/0/16
GE1/0/16 is administratively Up, Link is Up, Line protocol is Up
Hardware is Gigabit Ethernet, Interface is GE1/0/16, Address is 00:0b:86:6c:19:52
Port is the member of port-channel1
Encapsulation ARPA, Loopback not set
Configured: duplex (Full), Speed (1 Gbps), FC (Off), Autoneg (On)
Negotiated: duplex (Full), Speed (1 Gbps)
Interface index: 161
MTU 1514 bytes
Link flaps: 28
Flags: Port is a member of port-channel4
Link status last changed:       0d 00:15:08 ago
Last update of counters:        0d 00:00:07 ago
Last clearing of counters:      0d 01:36:13 ago
Statistics:
Received 41557 frames, 5669988 octets
6 pps, 7.912 Kbps
35928 unicast, 5605 multicast, 24 broadcast
0 runts, 19 giants, 0 throttles
28918 error octets, 0 CRC frames
Transmitted 1133140920 frames, 82607799007 octets
```

```
1 pps, 1.428 Kbps
43422 unicast, 626358842 multicast, 506738656 broadcast
0 throttles, 0 errors octets, 0 deferred
0 collisions, 0 late collisions
PoE Information:
Administratively Enable, Port status: On, Power consumption: 12000 mW
PSE port status: On
```

To view all Auto-LACP port-channel interface status in brief, execute the following command:

```
(host) # show interface brief port-channel auto-lacp
Interface      Admin   Link  Line Protocol  Speed/Duplex
---------      -----   ----  -------------  ------------
port-channel2  Enable  Up    Up             2 Gbps / Full
port-channel3  Enable  Up    Up             2 Gbps / Full
port-channel4  Enable  Up    Up             2 Gbps / Full
port-channel5  Enable  Up    Up             2 Gbps / Full
```

To view all neighboring devices, execute the following command:

```
(host) (config) #show lldp neighbor
Capability codes: (R)Router, (B)Bridge, (A)Access Point, (P)Phone, (S)Station (r)Repeater, (O)
Other
LLDP Neighbor Information
-------------------------
Local Intf  Chassis ID         Capability  Remote Intf
----------  ----------         ----------  -----------
GE0/0/1     18:64:72:c8:1e:40  B:A         18:64:72:c8:1e:40
GE0/0/3     18:64:72:c8:1e:40  B:A         18:64:72:c8:1e:40
GE0/0/6     9c:1c:12:c0:9f:e4  B:A         9c:1c:12:c0:9f:e4
GE0/0/7     9c:1c:12:c0:9f:e4  B:A         9c:1c:12:c0:9f:e4
Expiry (Secs)  System Name
-------------  -----------
103            18:64:72:c8:1e:40
102            18:64:72:c8:1e:40
107            9c:1c:12:c0:9f:e4
105            9c:1c:12:c0:9f:e4

Number of neighbors: 4
```

Triggers on Disabling Auto-LACP Port-Channels

When you disable Auto-LACP port-channels, keep the following points in mind:

- When you disable **auto-lacp** parameter from **device-group ap**, all Auto-LACP port-channels are deleted.
- When you disable **device-group ap**, all Auto-LACP port-channels are deleted.
- If an AP is deleted from one of the member interfaces or if there is an LLDP timeout, the corresponding Auto-LACP port-channel is deleted.
- In case a member is plugged out of one port and inserted into another, new port-channel will be formed with updated member interfaces.

GVRP Enhancements

Starting from ArubaOS 7.4.1, the following warning message is displayed on the Mobility Access Switch if you apply the GVRP profile on an interface without enabling global GVRP:
Warning: GVRP not enabled globally.

The sample command output with the warning message is as follows:

```
(host) (gigabitethernet "0/0/1") #switching-profile vlan10
(host) (gigabitethernet "0/0/1") #gvrp-profile vlan10
Warning: GVRP not enabled globally.
```

OSPF Enhancements

Starting from ArubaOS 7.4.1, Mobility Access Switch introduces the **show ip ospf interface brief** command to display OSPF details such as Interface, Instance, Area, IP/MASK, Cost, State, and Neighbors in a brief tabular format.

The following sample displays the output of the **show ip ospf interface brief** command in a tabular format:

```
(host) #  show ip ospf interface brief
Brief OSPF Interface Information
--------------------------------
Interface Instance Area    IP Address/Mask     Cost  State  Neighbors F/C
--------- -------- ----    ---------------     ----  -----  -------------
vlan201   0        0.0.0.0 69.1.1.1/255.255.255.0  1 DOWN   0/0
vlan202   0        0.0.0.1 79.11.1.1/255.255.255.0 1 DOWN   0/0
```

NOTE

In the **show** command output, the **Neighbors F/C** column represents fully adjacent neighbors and the total number of adjacent neighbors, respectively.

Source IP Configuration for TACACS Server

Starting from ArubaOS 7.4.1, the Mobility Access Switch introduces the **source-interface** command at the global and profile levels for the TACACS server. This command allows you to select a specific source interface IP address for the outgoing TACACS packets.

Configuring Source IP

The global source interface command is used to specify the source interface for all TACACS server request packets. If the source interface IP address is configured at the profile level, it takes precedence over the global source interface IP address.

The syntax for the global **source-interface** command is as follows:

```
(host) (config) #ip tacacs source-interface {loopback | vlan <id> [secondary <ip>]}
```

The following is a sample global **source-interface** command:

```
(host) (config) #ip tacacs source-interface vlan 55
```

The syntax for the profile-level **source-interface** command is as follows:

```
(host) (config) # aaa authentication-server tacacs <tacacs_server_name>
(host) (TACACS Server "<tacacs_server_name>") #source-interface {loopback | vlan <id>
[secondary <ip>]}
```

Some sample profile-level **source-interface** commands are as follows:

```
(host) (config) #aaa authentication-server tacacs tac1
(host) (TACACS Server "tac1") #source-interface loopback
(host) (config) #aaa authentication-server tacacs tac2
(host) (TACACS Server "tac2") #source-interface vlan 55
```

The following table describes the parameters for the **source-interface** command:

Table 3: *Parameters for the **source-interface** command*

Parameter	Description
`loopback`	Assigns the switch IP as the source IP.
`vlan <id>`	Assigns the IP address of the specified VLAN interface as the source IP.
`secondary <ip>`	Assigns a secondary source IP address in A.B.C.D format. This parameter is optional.

The following sample command configures the secondary IP address of VLAN 10 as the source interface IP address for all TACACS server request packets, provided there is no profile-level configuration:

```
(host) (config) #ip tacacs source-interface vlan 10 secondary 10.1.1.1
```

The following sample command configures the secondary IP address of VLAN 20 as the source interface IP address for a specific TACACS server:

```
(host) (config) #aaa authentication-server tacacs tac1
(host) (TACACS Server "tac1") #source-interface vlan 20 secondary 10.1.1.2
```

The following sample displays the output of the **show ip tacacs source-interface** command for the global and profile-level configurations mentioned here:

- The global source-interface is configured as vlan 55.
- The profile-level source-interfaces are configured as loopback and vlan 55 for two server profiles.

```
(host) (config) #show ip tacacs source-interface
Global TACACS source interface:
vlan: 55
ip: 55.0.0.2
loopback: disabled
Per-server client source IP addresses:
Server "tac1":  loopback enabled
Server "tac2":  vlan 55, IP 55.0.0.2
```

Support for Deleting a Switching-Profile

Starting from ArubaOS 7.4.1, the Mobility Access Switch introduces the **no switching-profile** command inside a tunnel to remove any switching-profile applied to the tunnel and point the tunnel back to the default switching-profile.

The following sample command deletes the switching-profile from the interface tunnel 50:

```
(host) (config) #interface tunnel ethernet 50
(host) (Tunnel "50") #no switching-profile
```

Verification of Switching-Profile Deletion

Execute the following **show interface tunnel** command to verify if any switching-profile applied to the tunnel is removed:

```
(host) (Tunnel "50") #show interface tunnel 50

tunnel 50 is administratively Up, Line protocol is Down
Description: GRE Interface
Source  unconfigured
Destination unconfigured
Tunnel mtu is set to 1100
Tunnel keepalive is disabled
Tunnel is an L2 GRE Tunnel
Protocol number  0
Tunnel is Trusted
Inter Tunnel Flooding is enabled
Switching-profile "default"
GRE Tunnel is up and running since:     00 00:00:00
```

The **show interface tunnel** command output displays the switching-profile as **default** when no switching-profile is applied to the interface tunnel.

Enhancements to show running-config Command

Starting from ArubaOS 7.4.0.2, the probe-profile protocol information (default value is ICMP) is displayed in the output of the **show running-config** command. The following sample displays the probe-profile protocol in the output of the **show running-config** command:

```
(host) (config) #show running-config | include icmp
Building Configuration...
netservice svc-icmp 1
ip access-list stateless icmp-acl-stateless
any any svc-icmp  permit
any any svc-icmp  permit
access-list stateless icmp-acl-stateless
protocol icm
```

Enhancements to Multicast Route Commands

Starting from ArubaOS 7.4.0.2, the counters for multicast route entries are included in the output of the following show commands:

- show ip pim mroute
- show ip pim mroute detail
- show ip pim-ssm mroute
- show ip pim-ssm mroute detail

The following sample displays the multicast counter values in the output of **show ip pim mroute** command:

```
(host) #show ip pim mroute
IP Multicast Route Table
Flags:  D - Dense, S - Sparse, s - SSM, C - Connected Receiver,
J - Join SPT, R - RP-bit set, T - SPT bit set
F - Register Flag, N - Null Register, A - Assert Winner
Total (*,G) Entries : 0
Total (S,G) Entries : 0
```

Total (*,G) Entries is the number of multicast routes to a specific group from any source.

Total (S,G) Entries is the number of multicast routes from a specific source to a specific group.

Route Monitoring

Starting from ArubaOS 7.4, Mobility Access Switch provides support for Route Monitoring. Route Monitoring enables the Mobility Access Switch to monitor the L3 uplink status using the ping probe. The ping probe destined to a server IP address is sent on the uplink interface which is under monitoring. Based on the status of the ping reply, the probe status of the interface is updated to up or down. When the probe status of the interface is down, the Mobility Access Switch removes the interface host and network routes from the routing table. When the probe status of the interface is up, interface host and network routes are added back.

By default, Route Monitoring is disabled on the Mobility Access Switch.

For more information on configuring Route Monitoring, see *ArubaOS 7.4 User Guide*.

Enhancements to Route Monitoring

Starting from ArubaOS 7.4.0.3, the output of the **show probe** command displays a new column, **Flags**. The **Flags** column indicates the causes due to which the probe status of the interface is down. The cause can be one of the following:

- IP is your own-ip
- Protocol is down for the interface

- IP not assigned for the interface
- MAC is not resolved for the route next-hop
- Route is not present for the probe destination
- URL is not resolved

If the URL is not resolved, the probe status of the interface remains as Up to ensure that the routes remain in the routing table to reach the DNS server. However, the **Sent** and **Received** columns display **N/A** to indicate that no packets are forwarded.

The following sample displays the output of the **show probe** command:

```
(host)  #show probe
IPV4 PROBE Table
----------------
Vlan    Server          Protocol  Port   Probe-State  Sent  Received  Flags
-----   ------          --------  ----   -----------  ----  --------  -----
vlan1   10.16.44.110    ICMP      N/A    Own-IP       N/A   N/A       IP is your own-ip
vlan1   10.16.52.8      ICMP      N/A    Up           2     1         N/A
vlan1   www.google.com  ICMP      N/A    Up           1     0         N/A
vlan50  10.16.52.8      ICMP      N/A    Down         N/A   N/A       Protocol is down for the
interface
Total Probe host entries: 4
```

IGMPv3 Snooping

The Mobility Access Switch provides support for IGMPv3 snooping starting from ArubaOS 7.4. IGMPv3 Snooping is used to snoop the membership reports that have group records of different types. These group records specify the source specific Multicast (SSM) traffic for a particular group.

IGMP Snooping is configured as a profile under vlan-profile and is attached to a VLAN. By default, v3 snooping is disabled and v2 snooping is enabled in an igmp-snooping profile. A new configuration command is introduced to enable v3 snooping explicitly.

For more information on IGMPv3 Snooping, see *ArubaOS 7.4 User Guide*.

Support for Static Address Resolution Protocol

Starting from ArubaOS 7.4, you can add a static Address Resolution Protocol entry on the Mobility Access Switch. You can configure a static ARP entry using the CLI. For more information, see *ArubaOS 7.4 User Guide*.

Management Features

This release of ArubaOS provides support for the following Management features:

Modification to set stacking renumber Command

Starting from ArubaOS 7.4.1.1, the **set stacking renumber** command in the Mobility Access Switch is modified to allow a user renumber any stack member except the primary and secondary stack members.

The Mobility Access Switch displays the following error message if a user tries to renumber the primary and secondary stack members:

```
ERROR: Renumber involving Primary or Backup member-id is not allowed
```

Enhancements to show memory Command

Starting from ArubaOS 7.4.1.1, an additional parameter—**dpa**—is added to the **show memory** command to display the memory information for the dpa process.

The following table provides description for the new parameter added to the **show memory** command:

Table 4: *New Parameter for the **show memory** Command*

Parameter	Description
dpa	Displays the memory information for the dpa process.

The output of the **show memory** command for the dpa process is provided in the following sample:

```
(host) #show memory dpa
Memory page usage for dpa
Task Block Usage Summary:
Min/Max Used Block Sizes    4     1024
Allocated blocks/bytes      671   52388
Free blocks/bytes           3169  78652
Total bytes                       131040
Total Block Alloc Calls     5316
Allocated Page Usage:
Page Size:                  4096
Total pages allocated       254
Total bytes allocated       1040384
task_block_malloc pages     222
task_block_alloc pages      32
task_page_alloc() page Q    0
pool_alloc_page() pages     3
Allocated MultiPage Usage:
multipage blks/pages in use 11    219
multipage allocations/frees 11    0
multipage max page request  145
multipage max reused        0
multipage Q pages/blocks    0     0
multipages broken down      0
multipages returned to OS   0
Growable arrays (GDAs, GCAs * 2, BVs)
Current growable arrays     0
Number of growths           0
Max allocation in bytes     0
Task Memory (malloc, calloc, realloc, free)
Mallocs:                      1157
Callocs:                      0
Reallocs:                     0
Reallocs for more:            0
Reallocs for less:            0
Reallocs for initial:         0
Frees:                        1019
Bytes requested:              720403
Bytes allocated:              744720
Bytes wasted:                 24160
Most outstanding allocs:      192
Largest request:              589856
Currently outstanding allocs: 138
RUSAGE Stats:
rusage: ru_maxrss 0: ix 0 id 0 is 0: times 1 0
paging: rec 2997 faults 0 nswap 0: in/out 0 0
sigs: 0 cw: 3577
```

Enhancements to show memory debug Command Output

Starting from ArubaOS 7.4.1.1, the output of the **show memory debug** command includes debug information for the following processes:

- l3m

- l2m
- dpa
- stackmgr
- cmicm
- cmica

The output of the **show memory debug** command with debug information for l3m, l2m, dpa, stackmgr, cmicm, and cmica processes is shown in the following sample:

```
(host) #show memory debug
165144
memory snapshot: Sun Jul 19 21:44:58 2015
==========================================
Memory (Kb): total: 760748, free: 387792
32432 /mswitch/bin/cfgm
14564 /mswitch/bin/auth
11156 /mswitch/bin/dpa -t 0x17d27004 -p 0xdf -f -d 0x43ef44 -m 0x208
10316 /mswitch/bin/l3m
9068 /mswitch/bin/cmicm -t 0x3e675
8700 /mswitch/bin/l2m
7456 /mswitch/bin/cmica -t 0x2f
7408 /mswitch/bin/im -t0xff1 -l0x7
7100 /mswitch/bin/stackmgr
6676 /mswitch/bin/isakmpd
5772 /mswitch/bin/profmgr -m1
5088 /mswitch/bin/snmpd
4932 /mswitch/bin/cpfw
3740 /mswitch/bin/udbserver -m1
3156 /mswitch/bin/dbsync
Querying excessively large apps
auth     : 14/18M (cur/typical)  0M over
cfgm     : 31/22M (cur/typical)  9M over
=========================================
Type           Num Allocs     Size Allocs      Peak Alloc
...
0x2b026ed0               2           8288
0x2b27287c               4             64
0x2b47c788               1           1024
0x2b6e8f00               1           3612
profmgr  :  5/ 8M (cur/typical)  0M over isakmpd  :  6/ 6M (cur/typical)  0M over
cpfw     :  4/ 2M (cur/typical)  2M over
=========================================
Type           Num Allocs     Size Allocs      Peak Allocs      Peak Size
------         ------------    -------------    -------------     -----------
default               1728          567716             1881          572552
PC             Allocs            Size
----            ------            ----
0x412fd0             32           8192
0x41a008             32           8192
0x456d6c              1           1024
0x2aaff6b0               1             56
0x2ab00b1c              13            208
0x2ab00d58               1           1168
...
0x2af1cba0               1              8
0x2b0a9200              20         283540
0x2b516fe0             232           3074
0x2b5293d4               1             12
0x2b52aa5c               1             12
0x2b58abcc               1             18
0x2b8275a0               1            128
```

```
0x2b8298e8              2             24
dbsync    :  3/ 3M (cur/typical)  0M over
snmpd     :  4/ 5M (cur/typical)  0M over
udbserver:  3/ 4M (cur/typical)  0M over
l3m       : 10/ 4M (cur/typical)  6M over
========================================
l2m       :  8/ 4M (cur/typical)  4M over
========================================
im        :  7/ 4M (cur/typical)  3M over
========================================
dpa       : 10/ 4M (cur/typical)  6M over
========================================
stackmgr :  6/ 4M (cur/typical)  2M over
cmicm     :  8/ 4M (cur/typical)  4M over
========================================
cmica     :  7/ 4M (cur/typical)  3M over
========================================
Large files under /tmp
======================
/tmp/.var/log/traces:-rw-r--r--    1 root     root      1000240 Jul 19 21:38
l3m.log.0
==============
Process Output
==============
%CPU S   PID  PPID    VSZ   RSS START CMD
3.5 S  1432  1365  53544 32436 21:37 cfgm
3.7 S  1409  1365  52676 31236 21:37 fpcli
0.0 S  1521  1409  52676 31236 21:37 fpcli
0.0 S  1522  1521  52676 31236 21:37 fpcli
3.0 S  1504  1365  50288 30720 21:37 arci-cli-helper
2.8 S  1588  1552  44940 23796 21:37 [central-cfghelp]
0.1 S  1526  1365  56264 14576 21:37 auth
0.2 S  1461  1365  47824 11168 21:37 dpa -t 0x17d27004 -p 0xdf -f -d 0x43ef44 -m 0x208
0.1 S  1546  1365  48740 10340 21:37 l3m ...
```

Troubleshooting Zero-Touch Provisioning

Starting from ArubaOS 7.4.1, Mobility Access Switch introduces the **show ztp-boot-info** command to help troubleshoot any Zero Touch Provisioning (ZTP) issues.

For more information about ZTP, see *ArubaOS 7.4 User Guide*.

The output of the **show ztp-boot-info** command displays the status of various methods of provisioning a Mobility Access Switch. The output details include TFTP configuration download status, DHCP AMP discovery status, Activate AMP discovery status in addition to DHCP options received.

The following sample output displays the details of the TFTP method of provisioning:

```
(host) (config) #show ztp-boot-info
Zero Touch Provisioning Method: TFTP
Time of Provisioning          : Jun/18/2015 06:43:07
TFTP Config Download          : Successful
DHCP AMP Discovery            : N/A
Activate AMP Discovery        : N/A
DHCP Options Received
---------------------
Option No. Option Name  Value
---------- ------------ -------
3          Router       192.168.1.2
6          DNS Server   10.13.6.110
43         VSA
60         Vendor
67         Bootfile     AW0000161.cfg
```

```
150         TFTP Server  10.16.59.60
```

The following sample output displays the details of provisioning through Activate:

```
(host) #show ztp-boot-info
Zero Touch Provisioning Method: Activate
Time of Provisioning          : N/A
TFTP Config Download          : Failed
DHCP AMP Discovery            : Failed
Activate AMP Discovery        : Failed
DHCP/Activate provisioning aborted.
DHCP Options Received
---------------------
Option No. Option Name  Value
---------- -----------  -----
3          Router       192.168.1.2
6          DNS Server   10.13.6.110
43         VSA
60         Vendor
67         Bootfile
150        TFTP Server
```

The following table describes the output parameters for the **show-ztp-boot-info** command:

Table 5: *Description for Output Parameters of* ***show-ztp-boot-info*** *Command*

Parameter	Description
`Zero Touch Provisioning Method`	Displays **TFTP** or **Activate**. **NOTE:** Whenever ZTP fails, this still shows **Activate** as the provisioning method as theMobility Access Switch keeps polling Activate in the background as long as it is in factory default.
`Time of Provisioning`	Displays the Timestamp of provisioning in Date and Time format. **NOTE:** This field displays **N/A** if not provisioned.
`TFTP Config Download`	Displays **Successful** or **Failed**. **NOTE:** If TFTP is the chosen ZTP method, it is the first method to attempt provisioning, and the output dispalys **Successful**; otherwise, the output displays **Failed**.
`DHCP AMP Discovery`	Displays **Successful** if the AMP parameters were discovered through DHCP option 43. **NOTE:** If TFTP is the chosen method of provisioning, this is not applicable.
`Activate AMP Discovery`	Displays one of the following: ● **Successful**, if the AMP parameters are received through Activate. ● **N/A**, if the method is not attempted. ● **Failed**, if provisioning fails.
`DHCP Options Received`	Displays the various DHCP options received with the name and value in tabular format.

Enhancements to Traceoptions Command

Port Options

Starting from ArubaOS 7.4.1, the **port** command under traceoptions allows you to specify the actual interface number or the port-channel instead of specifying the index number of the port. The following two options are introduced under the **port** command:

- gigabitethernet—Specify the actual interface number

- port-channel—Specify the port-channel ID

The sample **port** configuration commands are as follows:

```
(host) (traceoptions) #mstp port gigabitethernet 0/0/6
(host) (traceoptions) #mstp port port-channel 1
```

The sample **show traceoptions** command with the port name displayed as the actual interface number is provided here:

```
(host) (traceoptions) #show traceoptions
traceoptions (N/A)
------------------
Parameter                               Value
---------                               -----
Layer2 Forwarding trace flags
Layer2 Forwarding trace level           debugging
Layer2 Forwarding trace file size (Mb)  10
MSTP trace flags
MSTP trace port gigabitethernet         0/0/6
MSTP trace port port-channel            N/A
Interface manager trace flags           infrastructure configuration ethernet vlan port-
channel tunnel loopback mgmt system-information
Interface manager trace level           debug
Chassis manager trace flags             fru poe-configuration interface association debug
LLDP trace flags
dhcp_snoop trace flags
igmp-snooping trace flags
pim sparse mode trace flags
ospf trace flags
routing trace flags
igmp trace flags
vrrp trace flags
ddns trace flags
stack-manager trace flags               primary-election route system webui configuration
Stack-manager trace level               informational
rmon trace flags
rmon trace level                        errors
rmon trace file size (Mb)               10
(Host) (traceoptions) #
(Host) (traceoptions) # show running-config | include mstp
Building Configuration...
mstp port gigabitethernet "0/0/6"
interface-profile mstp-profile "default"
mode mstp
mstp
```

Filtering Options for OSPF and PIM

Starting from ArubaOS 7.4.1, the **show traceoptions** command is enhanced to filter OSPF and PIM traces by interface ID.

Users can configure OSPF VLAN ID and Tunnel ID as filters using the following CLI commands:

```
(host) (traceoptions) #ospf vlanid <ID>
(host) (traceoptions) #ospf tunlid <ID>
```

Similarly, users can configure PIM VLAN ID and Tunnel ID as filters using the following CLI commands:

```
(host) (traceoptions) #pim vlanid <ID>
(host) (traceoptions) #pim tunlid <ID>
```

The following is a sample OSPF trace VLAN ID configuration command:

```
(host) (traceoptions) #ospf vlanid 800
```

The sample output of the **show traceoptions** command for the preceding OSPF VLAN configuration is as follows:

```
(host) (traceoptions) #show traceoptions
traceoptions (N/A)
------------------
Parameter                               Value
---------                               -----
Layer2 Forwarding trace flags
Layer2 Forwarding trace level           debugging
Layer2 Forwarding trace file size (Mb)  10
MSTP trace flags
MSTP trace port gigabitethernet         N/A
MSTP trace port port-channel            N/A
Interface manager trace flags           infrastructure configuration ethernet vlan port-
channel tunnel loopback mgmt system-information
Interface manager trace level           error
Chassis manager trace flags             fru poe-configuration interface association debug
LLDP trace flags
dhcp_snoop trace flags
igmp-snooping trace flags
pim sparse mode trace flags             all
pim sparse mode trace by vlanid         0
pim sparse mode trace by tunnel id      0
ospf trace flags                        all
OSPF trace by vlanid                    800
ospf trace by tunnel id                 0
routing trace flags
igmp trace flags
vrrp trace flags
ddns trace flags
stack-manager trace flags               primary-election route system webui configuration
Stack-manager trace level               informational
rmon trace flags
rmon trace level                        errors
rmon trace file size (Mb)               10
```

NOTE

Without OSPF or PIM trace configuration, no filtering is done by VLAN ID and Tunnel ID and the output of **show traceoptions** command displays *0*, by default.

Enhancements to Reload Command

The **reload** command is enhanced with more options such as **reload in** and **reload at** to reload a switch or stack member in/at a specific time and/or date.

The following table provides description for the available reload options:

Table 6: *Parameters for the **reload** Command*

Parameter	Description	Range
in <minutes>	Reloads the stack or switch after the specified time.	0-60
at <hours, minutes, month, date>	Reloads the switch or stack at a specific time and date in the format: <hours, minutes, month, date>.	0-23, 0-60, 1-12, 1-31
cancel	Cancels the scheduled reload from the switch	

Table 6: *Parameters for the* ***reload*** *Command*

Parameter	Description	Range

`<member>`
`   in <minutes>`
	Reloads a stack member after the specified time.	0-60

`<member>`
`   at <hours, minutes, month, date>`
	Reloads a stack member at a specific time and date.	0-23, 0-60, 1-12, 1-31

The following command reloads the switch after 60 minutes:

```
(host) #reload in 60
```

The following command reloads the switch at a specific time and date:

```
(host) #reload at 1 50 7 12
```

Support for Global and ACL-Based Packet Tracing

Starting from ArubaOS 7.4.1, Mobility Access Switch introduces the following CLI commands to enable global and ACL-based packet tracing:

- pkt-trace-global enable
- pkt-trace acl <ACL-name> enable

Enabling Global Packet Tracing

Execute the following CLI command to enable global packet tracing:

```
(host) # pkt-trace-global enable
```

Execute the following CLI command to disable global packet tracing:

```
(host) # pkt-trace-global disable
```

The following table describes the parameters of the **pkt-trace-global enable** command:

Table 7: *Parameters for the* ***pkt-trace-global enable*** *command*

Parameter	Description
`trace`	Configures datapath trace options.

`trace-hex-mask`
`   <tmask>`
	Configures datapath trace mask in Hex format.

Sample Configuration

The following sample **pkt-trace global** command configures trace mask for ACL functionality:

```
(host) # pkt-trace-global enable trace-hex-mask 0 trace acl-processing
```

Enabling ACL-Based Packet Tracing

Execute the following CLI command to enable packet tracing for an ACL entry:

```
(host) # pkt-trace acl <ACL-name> enable
```

Execute the following CLI command to disable packet tracing for an ACL entry:

```
(host) # pkt-trace acl <ACL-name> disable
```

The following table describes the parameters of the **pkt-trace acl enable** command:

Table 8: *Parameters for the **pkt-trace acl enable** command*

Parameter	Description
`log`	Writes packet trace data into log file.
`trace`	Configures datapath trace options.
`trace-hex-mask <tmask>`	Configures datapath trace mask in Hex format.

Sample Configuration

The following sample **pkt-trace acl** command writes packet trace data into log file for the stated ACL bug:

```
(host) #pkt-trace acl acl-bug-58651 enable log trace acl-processing
```

Verifying Packet Tracing Configuration

The following **show** command helps verify the packet tracing configuration:

```
(host) #show datapath debug trace-buffer
Datapath Trace Buffer Entries:
MacAddr(      bb) 0x0        0x0         0x0        0x0        0xb86a1
0x6ac00000
MacAddr(      bb) 0x0        0x0         0x0        0x0        0xb86a1
0x6ac00000
MacAddr(      bb) 0x0        0x0         0x0        0x0        0xb86a1
0x6ac00000
CPDNSok(      4f) 0x0        0x1         0xa1045a2  0x37       0x1f
0x0
...
```

CAUTION

The **show** command output may not completely imply that the packet tracing configuration is set. Enabling packet tracing might impact the throughput of the system.

Support for Factory-Reset on a Detached ArubaStack Member

Starting from ArubaOS 7.4.0.3, you can reset a detached ArubaStack member that boots up as a line card to its factory defaults. This allows you to reset the password on the Mobility Access Switch if the login credentials are lost.

To reset a line card to its factory-defaults:

1. Connect a local console to the serial port on the Mobility Access Switch.
2. From the console, log in to the Mobility Access Switch using the username, **password** and the password, **forgetme!**.
3. Execute the following commands:

```
(LC-1) #restore factory_default stacking
(LC-1) #reload
```

SNMP Enhancements

This release of ArubaOS provides support for the following SNMP enhancements:

- Starting from ArubaOS 7.4.1.1, the following trap is introduced in the Mobility Access Switch for indicating the Stack MAC change:
  - **wlsxStackMacChangeTrap**—This trap is triggered when there is any change in the stack MAC.

Note: The trap is not triggered so long the system MAC is alive. Either when a Mobility Access Switch is moved out of the stack or when there is power failure, the System MAC is dead and the trap is generated.

- Starting from ArubaOS 7.4.0.3, the slot details of the ArubaStack are included in the power supply missing trap.
- Starting from ArubaOS 7.4.0.2, the following Aruba Enterprise traps for linkup/linkdown status are introduced in the Mobility Access Switch:
  - **wlsxIfLinkDownTrap**—This trap is sent when the operational state of a link transitions to down state from any other state (indicated by **ifOperStatus** object in the trap).
  - **wlsxIfLinkUpTrap**—This trap is sent when the operational state of a link transitions to up state from any other state (indicated by **ifOperStatus** object in the trap).

Security Enhancements

This release of ArubaOS provides support for the following Security Enhancements:

Authentication Survivability

The authentication survivability feature supports a survivable authentication framework against the remote link failure when working with the CPPM authentication servers. When enabled, this feature allows the Mobility Access Switches to authenticate the previously connected clients against the cached credentials if the connection to the authentication server is temporarily lost.

Starting from ArubaOS 7.4.1, Mobility Access Switch supports the following authentication methods with Authentication Survivability:

- 802.1X clients with Termination disabled/enabled: EAP-TLS with CPPM as RADIUS server.
- MAC-Based Authentication clients: PAP method. CPPM server is not mandatory in this case.

NOTE

This release of ArubaOS supports only EAP-TLS standard for Authentication Survivability.

When the authentication survivability feature is enabled, the following authentication process is used:

1. The wired client connects to a Mobility Access Switch and authenticates to the external authentication server. The external authentication server can be CPPM.
2. Upon successful authentication, the Mobility Access Switch caches the authentication credentials of the connected users for the configured duration. The cache expiry duration for authentication survivability can be set within the range of 1-72 hours, with 24 hours being the default cache timeout duration.
3. If the client roams or tries to reconnect to the Mobility Access Switch and the remote link fails due to the unavailability of the authentication server, the Mobility Access Switch uses the cached credentials in the internal authentication server to authenticate the user. However, if the user tries to reconnect after the cache expiry, the authentication fails.
4. When the authentication server is available and if the client tries to reconnect, the Mobility Access Switch detects the availability of server and allows the client to authenticate to the server. Upon successful authentication, the Mobility Access Switch cache details are refreshed.

The following attributes are supported from CPPM server along with the caching credentials:

- PW_USER_NAME
- PW_SESSION_TIMEOUT
- MS_TUNNEL_TYPE
- MS_TUNNEL_MEDIUM_TYPE
- MS_TUNNEL_PRIVATE_GROUP_ID
- ARUBA_ROLE

- ARUBA_VLAN
- ARUBA_CPPM_ROLE (Downloadable Role)
- ARUBA_ADMIN_ROLE

Important Points to Remember

- Any client connected through CPPM and authenticated through Mobility Access Switch remains authenticated with the Mobility Access Switch even if the client is removed from the CPPM server during the CPPM downtime.
- For EAP-TLS authentication, ensure that the CPPM 6.5.1.7 or later version is used for authentication.
- The cached credentials of a client will be deleted, if it fails the authentication via CPPM server. The credentials will be cached again if the subsequent authentication is successful.
- When the role download fails for a user after successful authentication from CPPM, the user will remain in initial or a previously known role even though the cache table cached the name of the role that failed to download, The role is not applied to the user when the CPPM server is down. The clients must do a fresh authentication with the cached credentials.
- When the CPPM server is unreachable , the user gets authenticated with the cached credentials only if all the following entries match the cached entries:
  - mac address
  - username
  - auth-type (EAP-TLS or PAP)

Configuring Authentication Survivability

You can enable authentication survivability on the Mobility Access Switch using the following CLI command:

```
(host) (config) #aaa auth-survivability enable
```

Execute the following command to set the duration after which the authenticated credentials in the cache must expire.

```
(host) (config) #aaa auth-survivability cache-lifetime <1-72>
```

Specify a value in hours for Cache timeout. The allowed range is 1 to 72 hours and the default value is 24 hours.

Execute the following command to specify a server certificate which will be used by the survival server to terminate EAP-TLS for 802.1X authentication.

```
(host)(config) #aaa auth-survivability server-cert ?
<server-cert-name>      Name of the Server Certificate
```

Sample Configuration

```
(host) (config) #aaa auth-survivability enable
(host) (config) #aaa auth-survivability cache-lifetime 25
```

Verifying Authentication Survivability Configuration

Execute the following command to verify the Authentication Survivability configuration on the Mobility Access Switch:

```
(host)(config) #show aaa auth-survivability
Auth-Survivability: Enabled (Running)
Survival-Server Server-Cert: server-crt
Survival-Server Cache lifetime: 72 hours
```

Viewing Survived Authentication Entries

To view the cached entries on Mobility Access Switch, use the following command:

```
(host) #show aaa auth-survivability-cache
```

```
Auth-Survivability Cached Data
------------------------------
MAC                User Name          Authenticated By  Authenticated On  Attributes                 AuthType
-----------------  ---------          ----------------  ----------------  ----------                 --------
00:00:00:01:01:01  00:00:00:01:01:01  cppm1             2015-06-11 08:02  CPPM Role(auth_surv-3167-2)  PAP
```

Clearing Cache Entries

To clear the cache entries manually, use the following commands:

```
(host) (config) #clear aaa auth-survivability-cache mac <mac address of client>
(host) (config) #clear aaa auth-survivability-cache all
```

Limitations

- 802.1X reauthentication timer value should be less than the dead interval time.

NTP Upgrade and Vulnerability Fixes

Starting from ArubaOS 7.4.1.3, the Network Time Protocol (NTP) version is upgraded from ntp-4.2.6p5 to ntp-4.2.8p4. With this upgrade, 13 NTP vulnerabilities are addressed; further, the server IP resolution and sync times are quicker than those in earlier versions.

Enhancement to show command output

The output of the **show ntp servers** command is enhanced to include 138 columns of data and 16 lines of server status as compared to 72 columns in previous ArubaOS releases.

```
(host) #show ntp servers
NTP Server Table Entries
----------------------------------------------------------------------
Flags:     * Selected for synchronization
+ Included in the final selection set
# Selected for synchronization but distance exceeds maximum
- Discarded by the clustering algorithm
= mode is client
remote                                  local
===============================================================================
*LOCAL(1)                              127.0.0.1

st   poll   reach    delay     offset     disp
==========================================================
16   1024   0        0.00000   0.000000    3.99217
```

Support for Proxy ARP

Starting from ArubaOS 7.4.1.1, the Mobility Access Switch provides proxy ARP support. If the proxy ARP feature is enabled, SOS traps all the ARP packets to the Proxy-ARP module in the control plane.

Configuring Proxy ARP

To configure a port security profile (for example, "PARP"), execute the following command:

```
(host) (config) #interface-profile port-security-profile "PARP"
```

To enable the proxy-arp, execute the following command:

```
(host) (Port security profile "PARP") #proxy-arp
```

Execute the following command to enter the port mode:

```
(host) (Port security profile "PARP") #interface gigabitethernet 0/0/10
```

To apply the port-security-profile to the interface, execute the following command:

```
(host) (gigabitethernet "0/0/10") #port-security-profile "PARP"
```

Enhancement to the show Command Output

Starting from ArubaOS 7.4.1.1, the **show interface-profile port-security-profile** command output is enhanced to include the status of the proxy ARP.

The following sample shows the output of the **show interface-profile port-security-profile** command that displays the status of the proxy ARP.

```
(host) #show interface-profile port-security-profile PARP
Port security profile "PARP"
----------------------------
Parameter                             Value
---------                             -----
IPV6 RA Guard Action                  N/A
IPV6 RA Guard Auto Recovery Time      N/A
MAC Limit                             N/A
MAC Limit Action                      N/A
MAC Limit Auto Recovery Time          N/A
Sticky MAC                            Disabled
Sticky  MAC Action                    N/A
Sticky MAC Auto Recovery Time         N/A
Trust DHCP                            No
Port Loop Protect                     N/A
Port Loop Protect Auto Recovery Time  N/A
IP Source Guard                       N/A
Dynamic Arp Inspection                N/A
Proxy Arp                            Enabled
```

The following are the commands to unconfigure proxy ARP:

```
(host) (Port security profile "PARP") #interface gigabitethernet 0/0/10
(host) (gigabitethernet "0/0/10") #no port-security-profile
(host) (gigabitethernet "0/0/10")#no interface-profile port-security-profile "PARP"
```

Support for Deleting Downloadable Roles

Starting from ArubaOS 7.4.1, Mobility Access Switch provides support for deleting downloadable roles from the CPPM server if the following conditions are met:

- No user references it
- It is in **Complete** or **Incomplete** state

Execute the following CLI command for deleting a role downloaded from the CPPM server:

```
(host) #downloadable-role-delete <role>
```

NOTE

The following error message is displayed if you try to delete a role that is not downloaded from CPPM or a non-existing role:

Invalid role <role-name>

The following sample CLI command deletes the abc_profile-3023-8 user role:

```
(host) #downloadable-role-delete  abc_profile-3023-8
```

Security Update

Starting from ArubaOS 7.4.1, the BASH access is disabled on Mobility Access Switch for security reasons.

Support for Port Bounce

Starting from ArubaOS 7.4.0.3, Mobility Access Switch provides support for the port bounce feature which enables a client to re-initiate a DHCP request when there is a VLAN change. This is achieved when a RADIUS server such as CPPM sends Disconnect-Request with a Vendor Specific Attribute (VSA 40) to the Mobility Access

Switch to trigger an interface shut down for a specified period. This allows the device to re-initiate a DHCP request for obtaining an IP address in the changed subnet.

The Disconnect-Request must include the following information:

- Calling Station-Id—MAC address of the user
- VSA—40
- Integer—0-60

VSA 40 represents Aruba-Port-Bounce-Host. The integer value indicates the time in seconds for which the Mobility Access Switch must shut the interface down. If the integer value received is 0 or a number greater than 60, the Mobility Access Switch does not shut the interface down.

NOTE

During a port bounce, the client connected to the interface is removed from the user table and is added back after the port is up.

Execute the following command to view the security logs during and after a port bounce:

```
(host) #show log security all | include Port
```

The following sample shows the output during a port bounce:

```
Press 'q' to abort.
Apr 29 06:06:19 :124004:  <DBUG> |authmgr|  Port Bounce Link Down flag set. Port =
gigabitethernet0/0/6.
Apr 29 06:06:19 :124004:  <DBUG> |authmgr|  LIF_OPER_STATE_UP. Port = gigabitethernet0/0/6.
Apr 29 06:06:20 :124004:  <DBUG> |authmgr|  Port will come up within 60 secs. link =
0x106ea2ac.
```

The following sample shows the output after a port bounce:

```
Press 'q' to abort.
Apr 29 06:06:19 :124004:  <DBUG> |authmgr|  Port Bounce Link Down flag set. Port =
gigabitethernet0/0/6.
Apr 29 06:06:19 :124004:  <DBUG> |authmgr|  LIF_OPER_STATE_UP. Port = gigabitethernet0/0/6.
Apr 29 06:06:20 :124004:  <DBUG> |authmgr|  Port will come up within 60 secs. link =
0x106ea2ac.
Apr 29 06:07:20 :124004:  <DBUG> |authmgr|  Port Bounce Link DOWN flag reset. Port =
gigabitethernet0/0/6.
```

Captive Portal Enhancements

Starting from ArubaOS 7.4.0.3, the **Authorization Required** page appearing before the actual Captive Portal login page is removed from the Mobility Access Switch.

Enhancements to Sticky MAC Configuration

Starting from ArubaOS7.4.0.2, the Mobility Access Switch allows you to configure the Sticky MAC feature with an action to take when a Sticky MAC violation occurs. The allowed actions are:

- Drop—Drops any new MAC addresses trying to connect to the interface. This is the default option.
- Shutdown—Shuts down the port on which the sticky MAC violation occurs. You can also optionally set an auto-recovery time between 0-65535 seconds for the interface to recover.

Configuring Sticky MAC Action

To enable and configure a Sticky MAC action, execute the following command:

```
(host) (config) #interface-profile port-security-profile <profile-name>
(host) (Port security profile "<profile-name>") #sticky-mac action [drop | shutdown auto-
recovery-time <1-65535>]
```

Sample Configuration

```
(host) (config) #interface-profile port-security-profile sticky
```

```
(host) (Port security profile "sticky") #sticky-mac action shutdown auto-recovery-time 10
```

Verifying Sticky MAC Configuration

Execute the following command to verify the Sticky MAC configuration:

```
(host) #show interface-profile port-security-profile <profile-name>
```

The following command verifies the sample configuration:

```
(host) #show interface-profile port-security-profile sticky
Port security profile "sticky"
------------------------------
Parameter                          Value
---------                          -----
IPV6 RA Guard Action               N/A
IPV6 RA Guard Auto Recovery Time   N/A
MAC Limit                          N/A
MAC Limit Action                   N/A
MAC Limit Auto Recovery Time       N/A
Sticky MAC                         Enabled
Sticky MAC Action                  Shutdown

Sticky MAC Auto Recovery Time      10 Seconds
Trust DHCP                         No
Port Loop Protect                  N/A
Port Loop Protect Auto Recovery Time  N/A
IP Source Guard                    N/A
Dynamic Arp Inspection             N/A
```

Enhancements to ClearPass Policy Manager (CPPM) Server Authentication

To download roles from the CPPM server, the Mobility Access Switch requires to provide the CPPM server admin credentials starting from CPPM 6.4.3. To achieve this, a new CLI command is introduced in ArubaOS 7.4.0.2. Using this command, you can configure the CPPM admin username/password, under **authentication-server** definition on the Mobility Access Switch.

Configure CPPM Server Credentials

Use the following command to configure CPPM Username/password:

```
(host) (config) #aaa authentication-server radius <server-name>
(host) (RADIUS Server "<server-name>") #cppm username <username> password <password>
```

Sample Configuration

```
(host) (config) #aaa authentication-server radius cppm1
(host) (RADIUS Server "cppm1") #host 1.1.1.1
(host) (RADIUS Server "cppm1") #key key123
(host) (RADIUS Server "cppm1") #cppm username admin password password123
(host) (RADIUS Server "cppm1") #exit
```

Verifying Configuration

The following show command displays the CPPM server credentials configured on the Mobility Access Switch:

```
(host) (config) #show aaa authentication-server radius cppm1
RADIUS Server "cppm1" (N/A)
---------------------------
Parameter                           Value
---------                           -----
Host                                1.1.1.1
Key                                 ********
CPPM credentials                    admin/********
Auth Port                           1812
Acct Port                           1813
Retransmits                         3
```

```
Timeout                              5 sec
NAS ID                               N/A
NAS IP                               N/A
Source intf                          N/A
Use MD5                              Disabled
Use IP address for calling station ID  Disabled
Mode                                 Enabled
```

Enhancements to web-server Command

As part of CVE-2014-3566 security vulnerabilities and exposures, **SSLv3** transport layer security is disabled from ArubaOS 7.4.0.1.

NOTE

Clients exclusively using SSLv3 will fail to access Captive Portal or Mobility Access Switch WebUI. It is recommended to use TLSv1, TLSv1.1, or TLSv1.2 transport layer security.

To address this, the following changes are introduced under the **web-server ssl-protocol** command.

Table 9: *Changes in the **web-server ssl-protocol** command*

Parameter	Description	Range	Default

`ssl-protocol`
`tlsv1`
`tlsv1.1`
`tlsv1.2`
	Specifies the Secure Sockets Layer (SSL) or Transport Layer Security (TLS) protocol version used for securing communication with the web server:
	● TLS v1
	● TLS v1.1
	● TLS v1.2
		—	tlsv1
			tlsv1.1
			tlsv1.2

Session ACLs on RVI

Starting from ArubaOS 7.4, you can apply session ACLs on a routed VLAN interface (RVI) of the Mobility Access Switch. For more information on configuring session ACLs on RVI, see *ArubaOS 7.4 User Guide*.

Deny Inter-User Traffic

Starting from ArubaOS 7.4, Mobility Access Switch provides support for Deny Inter-user Traffic. Deny Inter-user Traffic feature enables Mobility Access Switches to block the communication between users with the same role. For example, an organization can block communication between any two guest users. If the role has voip-profile configured, then the traffic across the VoIP users is also denied.

NOTE

The inter-user traffic denial happens only within an ArubaStack and does not span across multiple Mobility Access Switches or Aruba Stack.

By default this feature is disabled. You can configure Deny Inter-user Traffic for a maximum of seven user-roles (including CPPM downloaded roles) on a per user-role basis. For more information on configuring Deny Inter-User Traffic, see *ArubaOS 7.4 User Guide*.

Enhancements to Netdestination Alias

Starting from ArubaOS 7.4, a new netdestination alias, **localip** is introduced in the Mobility Access Switch. This is a system-defined alias which can be used as a destination alias for all the local IP addresses defined in the Mobility Access Switch.

QoS Enhancements

This release of ArubaOS provides support for the following QoS Enhancements:

Support for QoS Trust on Tunneled Node Port

Starting from ArubaOS 7.4.0.3, if **qos-trust** is enabled on a Tunneled Node port , the QoS markings (DSCP/dot1p) of the incoming packet are copied to the outer GRE header packet as well. This enables appropriate QoS treatment along the tunnel path.

Branch Features

This release of ArubaOS provides support for the following branch features:

DHCP Scope Distribution

Starting from ArubaOS 7.4.1.3, by default, only those Mobility Access Switches with configurations received from Trusted modes can access the data center.

Configurations for VPN deployments are done through Aruba Central or AirWave as these are considered as Trusted modes. If the Mobility Access Switch received the configuration from Aruba Central, the **Trusted Branch Mode** flag is set to 1. If it received the configuration from AirWave, the **Trusted Branch Mode** flag is set to 2. When this flag value is set to 1 or 2, the registration request is sent with the value for **Trusted Branch** set to **Yes**. Otherwise, the value for **Trusted Branch** is set to No and the controller will block the Mobility Access Switch.

Support for IP NAT Outside

Starting from ArubaOS 7.4, Mobility Access Switch provides support for IP Network Address Translation (NAT) outside on egress VLAN interface. The IP NAT outside feature changes the source IP of all the egressing packets to the IP of the egress VLAN interface. You can configure IP NAT outside using the CLI. For more information on configuring IP NAT Outside, see *ArubaOS 7.4 User Guide*.

Support for Dynamic Domain Name Server Client

Starting from ArubaOS 7.4, Mobility Access Switch provides support for Dynamic DNS Client. The Dynamic DNS Client enables a Mobility Access Switch to update its DHCP assigned IP address with a Dynamic DNS service provider. This helps to keep the remote devices reachable without tracking their IP address. For more information on DDNS configuration, see *ArubaOS 7.4 User Guide*.

Support for Myonlineportal.net

Starting from ArubaOS 7.4.0.1, Mobility Access Switch extends support for the **myonlineportal.net** dynamic DNS server in addition to the other servers supported in ArubaOS 7.4.

Aruba VPN Tunnel

The Mobility Access Switch at the branch acts as the VPN endpoint and the controller at the datacenter acts as the VPN concentrator. When a Mobility Access Switch is set up for VPN, it forms an IPsec tunnel to the controller to secure sensitive corporate data. IPsec authentication and authorization between the controller and the Mobility Access Switches is based on the RAP whitelist configured on the controller.

NOTE

You can configure an Aruba VPN tunnel either manually or through Zero Touch Provisioning (ZTP).

For more information on ZTP VPN and manual configuration of Aruba VPN Tunnel, see *ArubaOS 7.4 User Guide*.

Distributed, L3 DHCP Scopes

Starting from ArubaOS 7.4, Mobility Access Switch allows you to configure the DHCP address assignment for the branches connected to the corporate network through VPN. You can configure the range of DHCP IP addresses used in the branches and the number of client addresses allowed per branch. You can also specify the IP addresses that must be excluded from those assigned to clients, so that they are assigned statically. This release of Mobility Access Switch provides support for Distributed, L3 DHCP scope.

In Distributed L3 mode, DHCP server resides in the local branch on the Mobility Access Switch and each branch location is assigned a dedicated subnet. Based on the number of clients specified for each branch, the range of IP addresses is divided. Based on the IP address range and client count configuration, the DHCP server is configured with a unique subnet.

For more information on configuring Distributed L3, DHCP Scope, see *ArubaOS 7.4 User Guide*.

NAT Pools

Starting from ArubaOS 7.4, Mobility Access Switch provides support for NAT pools to protect private IPs of trusted servers behind the switch. It also gives the flexibility to support source NAT and dual NAT without using the switch IP. NAT actions can be performed only on packets processed by software. Support for applying session ACLs on RVI enables software processing of the packets that require a NAT action.

For more information on configuring NAT pools, see *ArubaOS 7.4 User Guide*.

VPN Survivability

The Mobility Access Switch provides support for a standby VPN uplink when the primary VPN uplink interface goes down. Whenever the primary uplink is detected to be down, the standby uplink is used to establish VPN.

For more information on configuring VPN Survivability, see *ArubaOS 7.4 User Guide*.

Default Route to VPN

Starting from ArubaOS 7.4, a crypto map matching all destinations can be used for customer applications that require all client generated traffic (Internet and Corporate bound) to be sent over a VPN tunnel. A branch office Mobility Access Switch has VPN tunnel which terminates on a Firewall. Any client non-corporate traffic from Mobility Access Switch is forwarded to the firewall through the VPN tunnel. This requires a default gateway route on Mobility Access Switch pointing to a VPN tunnel.

For more information about Default Route to VPN, see *ArubaOS 7.4 User Guide*.

Multiple Default Gateway Support

Default gateway is the route configured on the Mobility Access Switch to reach the upstream network. Starting from ArubaOS 7.4, Mobility Access Switch allows you to configure multiple default gateways using the metric option introduced in the CLI. Gateway with lower metric takes precedence when more than one gateways exist to a given upstream network. The second gateway with higher metric takes over when the first route is down.

For more information on multiple Default gateway support, see *ArubaOS 7.4 User Guide*

Access Point Integration Features

This release of ArubaOS provides support for the following portfolio integration features:

Configurable Rogue AP Containment

Starting from ArubaOS 7.4, the Mobility Access Switch allows you to configure the rogue AP containment using the CLI. This was enabled by default and was not configurable in ArubaOS 7.3.x versions.

You can now enable or disable rogue AP containment and configure the action to be taken on the list of MAC addresses received from IAP that are detected as rogue. The default action is to shut down the access port and

PoE on which it is detected and to discard the MAC address of the rogue AP and blacklist it if detected on a trunk port.

This feature is enabled by default. For more information on configuring Rogue AP Containment, see *ArubaOS 7.4 User Guide*.

Dynamic Port Reconfiguration

Starting from ArubaOS 7.4, Mobility Access Switch dynamically configures an interface based on the type of device connected to it. It uses LLDP to detect the type of device connected to an interface and applies a device-group configuration (a set of predefined configuration) on the interface based on the device-type.

> NOTE
>
> In this release, the Mobility Access Switch provides support only for the device-type and Aruba APs that support Aruba's proprietary LLDP TLVAP device-group.

This feature is disabled by default. For more information on reconfiguring ports dynamically, see *ArubaOS 7.4 User Guide*.

Platform Features

This release of ArubaOS provides support for the following platform features:

Restoring Factory Default Settings on S1500

Starting from ArubaOS 7.4.0.2, S1500 Mobility Access Switch allows you to use the **Mode** button to restore the switch to the factory default settings. You can enable this feature by using a CLI command on a configured S1500 Mobility Access Switch. After enabling the feature, you must push and hold the **Mode** button on the switch for about 15 seconds to reset it to the factory defaults. The Mobility Access Switch reboots after the reset.

Configuring Mode Button on S1500

Execute the following commands to enable the **Mode** button for factory reset:

```
(host) (config) #mode-button
(host) (mode-button) #enable factory-default
```

Verifying Mode Button Configuration

Use the following command to verify the **Mode** button configuration:

```
(host)  #show mode-button
mode-button (N/A)
-----------------
Parameter        Value
---------        -----
factory-default  enabled
```

Platform Enhancements

Starting from ArubaOS 7.4, the following enhancements are introduced in the Mobility Access Switch:

- DAC support on S1500 (GE Only)
- 10GBASE-ER SFP+
  - 40km over SMF
- 10GBASE-ZR SFP+
  - 80km over SMF

WebUI Enhancements

Starting from ArubaOS 7.4.1, the column header in the **Dashboard > Ports** page of the Mobility Access Switch WebUI is changed from **Total Errors** to **Total Error Frames** to indicate that the error counter refers to the frames counter.

Resolved Issues

This release of ArubaOS includes fixes for Network Time Protocol (NTP) vulnerabilities that are documented in CVE-2015-7691, CVE-2015-7692, CVE-2015-7701, CVE-2015-7702, CVE-2015-7703, CVE-2015-7704, CVE-2015-7705, CVE-2015-7848, CVE-2015-7849, CVE-2015-7850, CVE-2015-7851, CVE-2015-7852, CVE-2015-7853, CVE-2015-7854, CVE-2015-7855, and CVE-2015-7871. This release also includes fixes for CVE-2015-0286, CVE-2015-0292, and CVE-2015-1788. Additionally, this section lists the issues that are resolved until ArubaOS 7.4.1.3.

AirWave/Activate

Table 10: *Fixed AirWave/Activate Issues*

Bug ID	Description	Fixed in
108372	**Symptom**: The AirWave details obtained through DHCP options (60 and 43) were not retained by a Mobility Access Switch after a reload. **Scenario**: This issue was observed in Mobility Access Switches running ArubaOS 7.3 or later versions when in factory default settings.	7.4.0.1

Base OS Security

Table 11: *Fixed Base OS Security Issues*

Bug ID	Description	Fixed in
114450	**Symptom**: An authenticated user was not assigned the default authentication role but was assigned the initial logon role, when the role download from CPPM failed. This issue is resolved by assigning the default role to the user at the time of CPPM role download failure. **Scenario**: This issue occurred either when CPPM credentials were not given under the auth-server configuration or when there was invalid syntax in the role content. This issue was observed in S2500 and S3500 Mobility Access Switches running ArubaOS 7.4.0.2.	7.4.1.1
114452	**Symptom**: Users could not create NAT pools using a downloadable role in CPPM, though NAT pools can be referenced from CPPM. This issue is resolved by providing the IP NAT pool configuration support in Mobility Access Switch for dynamically downloadable role configuration in CPPM. **Scenario**: This issue occurred when the Mobility Access Switch tried to download NAT Pool configuration from CPPM. This issue was observed in Mobility Access Switches running ArubaOS 7.3 or later versions.	7.4.1.1
122891	**Symptom**: When user tried to upload malformed Elliptic Curve Digital Signature Algorithm (ECDSA) certificate on a Mobility Access Switch, the certmgr hung up while processing the ECParameters structure. Later, the switch had to be rebooted. This issue is resolved by implementing internal code changes that prevent infinite loop being caused by malformed ECParameters. **Scenario**: This issue was not limited to any specific platform or ArubaOS release version.	7.4.1.1
113613	**Symptom**: The **Invalid downloadable role** error messages were reported incorrectly on the Mobility Access Switch. **Scenario**: This issue was not limited to any specific Mobility Access Switch model or release version.	7.4.0.3
110867	**Symptom**: The extended ACL keyword, **established** was not effective for the traffic processed in the hardware. **Scenario**: This issue was not limited to any specific Mobility Access Switch model or release version.	7.4.0.2
105743	**Symptom**: A Mobility Access Switch crashed and rebooted due to a synchronization issue with the AAA user table. **Scenario**: This issue was not limited to any specific Mobility Access Switch model or release version.	7.4.0.1

Table 11: *Fixed Base OS Security Issues*

Bug ID	Description	Fixed in
105890	**Symptom**: The administrators were unable to login to the Mobility Access Switch using the console for a brief period. The logs indicated that the kernel killed an internal process with the following out of memory message: **nanny[1345]: <303093> <ERRS> |nanny| Out Of Memory handler killed process /mswitch/bin/aaa_proxy:1380 due to low memory. Set 1** **Scenario**: This issue was not limited to any specific Mobility Access Switch model or release version.	7.4.0.1
107099	**Symptom**: The log operator applied on an ACL was not effective when the ACL was applied to a Routed VLAN interface. **Scenario**: This issue was not limited to any specific Mobility Access Switch model or release version.	7.4.0.1
85582	**Symptom**: Quate CMS cross site scripting (XSS) vulnerabilities were noticed in the system. This issue is resolved by upgrading OpenSSL and Apache HTTP. **Scenario**: This issue was not limited to any specific Mobility Access Switch model or release version.	7.4

Captive Portal

Table 12: *Fixed Captive Portal Issues*

Bug ID	Description	Fixed in
115518	**Symptom**: Users could not access the allowed Web sites through Captive Portal as the whitelist ACLs were lost on the switch after a reboot or an internal process restart. **Scenario**: This issue was observed if Whitelist was configured in the Captive Portal profile. This issue was not limited to any specific Mobility Access Switch model or release version.	7.4.1

Configuration

Table 13: *Fixed Configuration Issues*

Bug ID	Description	Fixed in
101284	**Symptom**: The local IP address of the NTP server was displayed as 0.0.0.0 when executing the **show ntp servers** command after rebooting the Mobility Access Switch. This occurred because the NTPD was not refreshed with the switch IP address. This issue is resolved by initiating a seamless NTP restart. **Scenario**: This issue was observed only when a Mobility Access Switch running ArubaOS 7.3 or later version was rebooted after configuring the NTP servers.	7.4.1.3
112462	**Symptom**: When user tried to copy to or from the FTP server— that is, for any copy operation involving the FTP server—the password provided in the command line was viewable as clear text in the audit trail output. **Scenario**: This issue was not limited to any specific Mobility Access Switch model or ArubaOS release version.	7.4.1.1
112282	**Symptom**: A crash due to memory issues was observed in a Mobility Access Switch managed through AirWave. **Scenario**: This issue was observed in Mobility Access Switches running ArubaOS 7.3.0.1 when configured and managed through AirWave.	7.4.1
106082	**Symptom**: The CLI did not process a command that exceeded 252 characters. This issue is fixed by increasing the maximum command-line character limit to 512. **Scenario**: This issue was not limited to any specific Mobility Access Switch model or release version.	7.4.0.1
93768	**Symptom**: Multiple mirroring profiles creation was not allowed. This issue is resolved by allowing creation of multiple mirroring profiles. However, at a given time, only one mirroring profile can be applied to different interfaces. **Scenario**: This issue was not limited to a specific Mobility Access Switch model or release version.	7.4
94375	**Symptom**: Authenticated clients were unable to pass traffic causing a network outage. This issue is resolved by clearing the user table. **Scenario**: This issue was observed when MAC address of the uplink interface was learnt on untrusted interface. This issue was observed on Mobility Access Switches running ArubaOS 7.3.2.1	7.4

Data Path Agent

Table 14: *Fixed Data Path Agent Issues*

Bug ID	Description	Fixed in
105466	**Symptom**: The DPA process crashed when there were too many user updates in a fully loaded system. **Scenario**: This issue occurred when a system with untrusted users had a lot of user updates in multiple VLANs. This issue was observed in Mobility Access Switches running ArubaOS 7.3 or later versions.	7.4.0.2
99566	**Symptom**: The DPA process crashed on the Mobility Access Switch. **Scenario**: This issue occurred when a user pressed the MODE button on the front panel of the Mobility Access Switch during the boot process. This issue was observed in S1500-12P model running ArubaOS 7.3.0.1 or earlier versions.	7.4

Dynamic Host Configuration Protocol (DHCP)

Table 15: *Fixed DHCP Issues*

Bug ID	Description	Fixed in
104220	**Symptom**: The Mobility Access Switch incorrectly leased out an IP address that was used by another client. **Scenario**: This issue was observed when the DHCP server pool on the Mobility Access Switch had only one IP address to offer. This issue was not limited to any specific Mobility Access Switch model or release version.	7.4.0.3

Generic Routing Encapsulation (GRE)

Table 16: *Fixed GRE Issues*

Bug ID	Description	Fixed in
112619	**Symptom**: A GRE tunnel could not be established between a Mobility Access Switch and a controller when a non-zero value was configured for the tunnel type on the Mobility Access Switch. **Scenario**: This issue was observed when the Mobility Access Switch was trying to connect to a controller running ArubaOS 6.4 or later versions.	7.4.0.2

IPsec

Table 17: *Fixed IPsec Issues*

Bug ID	Description	Fixed in
85235	**Symptom**: Traffic outage occurred when re-keying of the VPN tunnel failed. This issue is resolved by enabling the NAT-T option in configuration because traffic is across WAN. **Scenario**: This issue was observed in a WAN deployment where VPN tunnel was established and re-keying of the tunnel was set to be executed every hour. However, VPN tunnel failed to re-key after every 6 to 8 hours causing traffic outage. But at the next re-key interval (after one hour), the re-keying was successful and allowed traffic.	7.4

Layer 2 Forwarding

Table 18: *Fixed Layer 2 Forwarding Issues*

Bug ID	Description	Fixed in
120916	**Symptom**: L2M crashed and the user was not able to access or ping the Mobility Access Switch. Also, access points connected to the Mobility Access Switch went DOWN. This issue is resolved by preventing the overflow of timestamp value. **Scenario**: This issue occurred when i) the Mobility Access Switch had been running for more than 250 days, ii) a new device—AP or phone—was added or modified after 250 days, and iii) the device that was added in the beginning was either removed or went DOWN that resulted in an LLDP request for deletion. This issue was observed in all Mobility Access Switches running ArubaOS 7.3.2.2.	7.4.1.3
109561	**Symptom**: A disruption in the network traffic was sometimes observed in an ArubaStack. Users sometimes encountered the following error message upon executing any layer 2 CLI command: **Module Layer 2 manager is busy** **Scenario**: This issue occurred if a stack member disconnected and rejoined the Aruba Stack when spanning tree was enabled. This issue was observed in an Aruba Stack running ArubaOS 7.3.x or later versions.	7.4.0.2
107450	**Symptom**: The process handling layer 2 functions crashed when the Mobility Access Switch received LLDP BPDUs with System Description exceeding 256 bytes. **Scenario**: This issue was observed in Mobility Access Switches running ArubaOS 7.4.	7.4.0.1

Link Layer Discovery Protocol (LLDP)

Table 19: *Fixed LLDP Issues*

Bug ID	Description	Fixed in
125727	**Symptom**: LLDP-enabled VoIP phones that were connected to a Mobility Access Switch on untrusted interfaces did not pass traffic intermittently. High utilization of the datapath CPUs was also observed. This issue is resolved by stopping the specific DHCP packets from being reinjected into the control plane and allowing the client to resend them. **Scenario**: This issue occurred when the DHCP packets from the VoIP phones were reinjected into the control plane, thereby chocking the communication channel between the datapath and the control plane. This issue was observed in Mobility Access Switches running ArubaOS 7.4.0.3 or ArubaOS 7.4.1.1.	7.4.1.3
121465	**Symptom**: The output of the **show neighbor-devices** command incorrectly displayed the uplink port (local interface) as the port on which CDP-capable device was connected, though there was no functionality impact due to this. The issue is resolved by pushing the CDP packets to the control plane from SOS instead of being tunneled to the uplink port. **Scenario**: This issue occurred when the CDP-capable device was connected to the controller via tunneled-node interface. This issue was observed in S1500, S2500, and S3500 Mobility Access Switch models.	7.4.1.1

Multicast

Table 20: *Fixed Multicast Issues*

Bug ID	Description	Fixed in
93220	**Symptom**: Domain login took unusually long time when the traffic went through tunnel node. This issue is resolved by changing the configuration to use TCP instead of UDP so that the server does not expect packets in sequence. **Scenario**: This issue was observed when the kerb client was connected to tunneled node port, and the client sent packets exceeding tunnel MTU. This issue was not limited to any specific Mobility Access Switch model or release version.	7.4

OSPF

Table 21: *Fixed OSPF Issues*

Bug ID	Description	Fixed in
98544	**Symptom**: OSPF convergence time was longer (about 45 seconds) in a Mobility Access Switch. For achieving faster convergence (about 25 seconds), the **hello interval** time needs to be explicitly configured to 7 seconds as compared to its default value of 10 seconds. **NOTE:** OSPF neighbors trying to form an adjacency with a Mobility Access Switch running ArubaOS 7.4.1.1 must have the value for **hello interval** time matching with that of the Mobility Access Switch. **Scenario**: This issue was observed when **OSPF** was configured on a Mobility Access Switch. This issue was not limited to any specific Mobility Access Switch model or release version. **NOTE:** The default values for **hello interval** and **dead interval** times were changed in ArubaOS 7.4.1 version, but are reverted to the original default values in ArubaOS 7.4.1.1.	7.4.1.1

For more information about the change of default values for **hello interval** and **dead interval** times in ArubaOS 7.4.1, refer to the *ArubaOS 7.4.1 Release Notes* available in the support site.

RADIUS

Table 22: *Fixed RADIUS Issues*

Bug ID	Description	Fixed in
107183	**Symptom**: The following debug message was incorrectly reported in the error logs of the Mobility Access Switch as the accounting messages were incorrectly sent for the unauthenticated users: **An internal system error has occurred at file rc_acct.c print** **Scenario**: This issue was observed in Mobility Access Switches running ArubaOS 7.3.2.1 or later versions when interim accounting was enabled in the AAA profile.	7.4.0.1

Routing

Table 23: *Fixed Routing Issues*

Bug ID	Description	Fixed in
110596	**Symptom**: The following error message appeared when executing the command, **clear ip ospf process** on the Mobility Access Switch: **Module Layer 3 Manager is busy. Please try later** **Scenario**: The issue occured if the command was executed when a default OSPF route or a router IP that conflicted with the tunnel destination IP was advertised through GRE over VPN tunnel. This issue was observed in Mobility Access Switches running ArubaOS 7.3 or later versions.	7.4.0.2
109727	**Symptom**: A Mobility Access Switch failed to respond to ARP requests for an IP used in a NAT pool even though **session-processing** was enabled on the uplink VLAN. **Scenario**: This issue was observed when a session ACL was applied on a VLAN that had a source NAT configured from a NAT pool. This issue was limited to Mobility Access Switches running ArubaOS 7.4.	7.4.0.1
109920	**Symptom**: The following error message was displayed on a Mobility Access Switch when executing any layer 3 command in the CLI: **Module Layer3 Manager is busy. Please try later** The message logs indicated that the module handling the layer 3 functions had crashed. **Scenario**: The crash occurred when a default OSPF route or a router IP that conflicted with the tunnel destination IP was advertised through GRE over VPN tunnel. This issue was observed in Mobility Access Switches running ArubaOS 7.3 or later versions.	7.4.0.1

Stacking

Table 24: *Fixed Stacking Issues*

Bug ID	Description	Fixed in
94551	**Symptom**: The output of **show stacking members** command displayed more than eight members with valid member IDs even though Mobility Access Switch supports only up to eight members in an ArubaStack. This issue is resolved by ensuring that a maximum of eight members are only allowed in an ArubaStack. **Scenario**: This issue was observed when more than eight members were added to the ArubaStack. This issue was not specific to any Mobility Access Switch model or release version.	7.4
95855	**Symptom**: The Primary member of a 5-member ArubaStack rebooted due to memory leak. **Scenario**: The issue occurred when LLDP was enabled in the ArubaStack. This issue was limited to ArubaStack with S3500 Mobility Access Switches running ArubaOS 7.3.0 or later.	7.4
103518	**Symptom**: The Mobility Access Switch displayed the **Module Layer 2 manager is busy** error message on issuing any CLI command. **Scenario**: This issue occurred during a system switchover or Layer 2 Module (L2M) process restart. This issue was observed in Mobility Access Switches running ArubaOS 7.3.2.2 or earlier versions.	7.4

Switch-Datapath

Table 25: *Fixed Switch-Datapath Issues*

Bug ID	Description	Fixed in
123171	**Symptom**: An S2500 Mobility Access Switch restarted when datapath crashed due to buffer corruption. This issue is resolved by implementing code changes that prevent buffer overflow. **Scenario**: This issue was observed when at least 34 clients and one router/gateway were connected to the Mobility Access Switch on untrusted ports. This issue was not limited to any specific Mobility Access Switch model or release version.	7.4.1.1
113397	**Symptom**: Sometimes, the clients connected to a Mobility Access Switch obtained an IP address from the initial VLAN instead of the final VLAN. **Scenario**: This issue was not limited to any specific Mobility Access Switch model or release version.	7.4.0.3
113448	**Symptom**: DHCP broadcast packets were dropped by ingress RVI ACL configured on a Mobility Access Switch. **Scenario**: This issue was observed in Mobility Access Switches running ArubaOS 7.4.0.1.	7.4.0.2
97002	**Symptom**: The Mobility Access Switch dropped packets when the traffic rate was high on the egress port due to insufficient port buffer. **Scenario**: The issue was not limited to any specific Mobility Access Switch model or release version.	7.4

Switch-Platform

Table 26: *Fixed Switch-Platform Issues*

Bug ID	Description	Fixed in
132980	**Symptom**: The Mobility Access Switch rebooted because of kernel panic. This issue is resolved by not sending the unhandled IP packets to the control plane for logging. **Scenario**: This issue occurred when unhandled IP fragments were waiting to be reassembled. This issue was not limited to any specific Mobility Access Switch model or ArubaOS release version.	7.4.1.3
112286	**Symptom**: The following error message was displayed on a Mobility Access Switch a few minutes after executing any Layer 3 show command in the CLI: **Module Layer3 Manager is busy. Please try later.** **Scenario**: This issue was observed when a large number of hosts were connected to the Mobility Access Switch. This issue was not limited to any specific Mobility Access Switch model or release version.	7.4.1
113966	**Symptom**: The Mobility Access Switch WebUI did not display all the stacking ports correctly. **Scenario**: This issue was observed in an ArubaStack running ArubaOS 7.3 or later versions.	7.4.0.3

Table 26: *Fixed Switch-Platform Issues*

Bug ID	Description	Fixed in
114628	**Symptom**: Transceivers connected to a Mobility Access Switch were not detected when the Mobility Access Switch was reloaded after an image upgrade or downgrade. **Scenario**: This issue was observed when there were multiple transceivers connected to the Mobility Access Switch running ArubaOS 7.4.0.2.	7.4.0.3
111206	**Symptom**: The process handling layer 3 functions crashed when Dyn DNS was enabled on a Mobility Access Switch. **Scenario**: This issue was observed in Mobility Access Switches running ArubaOS 7.4.0.1.	7.4.0.2
98030	**Symptom:** A stack member stopped responding and rebooted. **Scenario:** The log files for the event suggested multiple link flaps. Due to this, the Chassis Manager (CM) process missed keep-alives and removed the stack member from the ArubaStack. This issue was observed in Mobility Access Switches running ArubaOS 7.2.2.2. **NOTE:** This issue was caused due to a cabling problem at the customer site.	N/A
89131 95757 104999	**Symptom**: Crash file was unavailable for a crash due to kernel panic. This issue is resolved by adding the watchdog and Non Maskable Interrupt (NMI) support. **Scenario**: This issue occurred because of synchronization problems in the panic routine. This issue was not limited to a specific Mobility Access Switch model or release version	7.4
99562	**Symptom**: The Mobility Access Switch stopped detecting SFP/SFP+ transceivers when they were plugged out and inserted back in, or replaced. **Scenario**: This issue was observed in Mobility Access Switches running ArubaOS 7.3.1.0 or earlier versions.	7.4

WebUI

Table 27: *Fixed WebUI Issues*

Bug ID	Description	Fixed in
105975	**Symptom**: **Copy Backup** option in WebUI did not redirect to the **Copy files** page after upgrading the Mobility Access Switch from ArubaOS 7.2 to 7.3.2.2. **Scenario**: This issue occurred when ArubaOS was upgraded to 7.3 or later versions on the Mobility Access Switches.	7.4.0.1
104261	**Symptom**: The **Allowed VLAN** field under the **Configuration > Ports > Switching** tab was inaccessible through the WebUI of the Mobility Access Switch. **Scenario**: This issue occurred when the Mobility Access Switch was upgraded from ArubaOS 7.3.1.0 to ArubaOS 7.3.2.0. This issue was observed in Mobility Access Switches running ArubaOS 7.3.2.0 or later versions.	7.4

Known Issues and Limitations

The following are known issues and limitations observed in ArubaOS 7.4.1.3. Bug IDs and applicable workarounds are included.

Base OS Security

Table 28: *Known Base OS Security Issues and Limitations*

Bug ID	Description
74264	**Symptom**: A combination of CPPM and Windows Radius server for fail-through is not supported. **Scenario**: This issue is not limited to any specific Mobility Access Switch model or release version. **Workaround**: Use either CPPM servers as Primary and Backup or Windows Radius as Primary and Backup. Do not combine them.
85575	**Symptom**: User reauthentication does not happen after the reauth timer expires. This issue is observed when the EAP type is **TLS** and **EAP Termination** is enabled in the 802.1X profile. Due to this issue, periodic user reauthentication does not happen. **Scenario**: This issue is not limited to any specific Mobility Access Switch model or release version. **Workaround**: None
87971	**Symptom**: The Mobility Access Switch IP is programmed as the loopback IP automatically when there is no **ip-cp redirect address** configured. If **ip-cp redirect address** is configured and saved in the config, either system switchover or reload is invoked. After switchover or reload the configured **ip-cp redirect address** is lost. The IP address displays all 0s. **Scenario**: This issue occurs only when the **ip cp-redirect-address<ip-addr>** command is configured on Mobility Access Switches running ArubaOS 7.3. **Workaround**: If the **ip-cp redirect address** command is explicitly configured and it is lost after reload or switchover, configure the **ip cp-redirect-address<ip-addr>** command once again and save it.
90067	**Symptom**: A ClearPass Policy Manager (CPPM) **Downloadable Role** may not be properly assigned to a Mobility Access Switch user if it is not correctly configured in CPPM. **Scenario**: This issue occurs when the Mobility Access Switch is still processing the invalid **Downloadable Role** and an administrator has already modified the **Downloadable Role** in CPPM. This issue occurs on Mobility Access Switches running ArubaOS 7.3. **Workaround**: Ensure that the role definition syntax is correct in CPPM. This can be verified by testing the configuration on a test switch before configuring the role details in CPPM. If that is not possible and a **Downloadable Role** has been incorrectly defined, wait for the Mobility Access Switch to complete processing the invalid role (~3 minutes), delete the user(s) assigned to that role, update the role definition in CPPM and re-trigger authentication.
98778	**Symptom**: Client traffic is blocked because the Mobility Access Switch fails to download the CPPM role for an 802.1X authenticated user. **Scenario**: This issue occurs when **preauth** is enabled in the Mobility Access Switch. While the client passes MAC authentication and gets a role, it fails to download the CPPM role after 802.1X authentication. This is because the datapath still shows preauth ACL instead of the default MAC ACL. This issue is not limited to any specific Mobility Access Switch model or release version. **Workaround**: None

Table 28: *Known Base OS Security Issues and Limitations*

Bug ID	Description
100904	**Symptom**: When a client successfully authenticated by MAC and/or 802.1X authentication fails reauthentication, it remains in the authenticated VLAN even after it moves back to the previous role. **Scenario**: This issue is not limited to any specific Mobility Access Switch model or release version. **Workaround**: Delete the failed user entry manually.
101489	**Symptom**: When an authenticated client fails reauthentication after an EAP-start, it remains in the previously authenticated role and VLAN. **Scenario**: This issue is not limited to any specific Mobility Access Switch model or release version. **Workaround**: Delete the failed user entry manually.
120783	**Symptom**: The authentication (auth) servers that are marked as out-of-service are not cleared from the server-group table—even after executing the **auth-server** command—until the **dead timer** expires. **Scenario**: This issue is observed in S2500 and S3500 Mobility Access Switches running ArubaOS 7.4.x. **Workaround**: Execute the following command to bring the auth server back to service: `aaa inservice <server-group> <authentication-server>`

Central

Table 29: *Known Central Issues and Limitations*

Bug ID	Description
102328	**Symptom**: When the Mobility Access Switch is in managed mode, the configuration received or sent from Aruba Central are not processed and applied properly, if the size of running-config file exceeds 150KB. **Scenario**: This issue occurs when the Mobility Access Switch has a large number of profile configuration defined and managed by Aruba Central. This issue is observed on a stand-alone Mobility Access Switch running ArubaOS 7.3.2 or later versions. **Workaround**: None.
104181	**Symptom**: Users are unable to configure the Mobility Access Switch from the console for 5 to 10 mins after it loses connection from Aruba Central. **Scenario**: This issue occurs when the Mobility Access Switch in Managed mode abruptly disconnects from Aruba Central. This issue is observed on a standalone Mobility Access Switch running ArubaOS 7.3.2.2 or later versions. **Workaround**: None.

Configuration

Table 30: *Known Configuration Issues and Limitations*

Bug ID	Description
55306	**Symptom**: User is unable to delete the characters using the backspace key when the admin username is as long as the maximum characters. **Scenario**: This issue is observed when the admin username reaches the maximum limit (32 characters). This issue is not limited to any specific Mobility Access Switch model. **Workaround**: Press enter key and type the username again.
99871	**Symptom**: Sometimes, the user prompt does not appear on the Mobility Access Switch console after a reload. **Scenario**: This issue is observed only when a Mobility Access Switch running ArubaOS 7.4 is reloaded. **Workaround**: Press any key to proceed with the login.
101943	**Symptom**: Users cannot configure the banner Message of the Day (MOTD) text using the **banner motd** command in the same line. **Scenario**: This issue is not limited to any specific Mobility Access Switch model or release version. **Workaround**: Enter the banner text to be configured with a delimiter in a new line after the **banner motd** keyword.
106839	**Symptom**: The Mobility Access Switch does not preserve extra spaces in the banner text. **Scenario**: If the banner text starts with multiple spaces or has consecutive multiple spaces in between, the extra spaces are trimmed and replaced with a single space. Because of this, a banner with extra spaces in the text is not displayed properly. This issue is not limited to any specific Mobility Access Switch model, release version, or topology. **Workaround**: None.

DHCP Snooping

Table 31: *Known DHCP Snooping Issues and Limitations*

Bug ID	Description
87131	**Symptom**: When a line card member of an ArubaStack is individually rebooted, the DHCP Snooping bindings for that particular member switch are lost. **Scenario**: Reloading a line card does not trigger repopulating the DHCP Snooping database. However, the DHCP Snooping database repopulates in case of a stack or box reload. This issue occurs on Mobility Access Switches running ArubaOS 7.3. **Workaround**: None.

Data Path Agent (DPA)

Table 32: *Known DPA Issues and Limitations*

Bug ID	Description
98845	**Symptom**: The DPA process crashes on the Mobility Access Switch. **Scenario**: This issue is observed when the DPA process waits for an acknowledgment from the SOS process and times out. This issue is observed in Mobility Access Switches running ArubaOS 7.3.0.1 or later versions. **Workaround**: None.

Dynamic ARP Inspection (DAI)

Table 33: *Known DAI Issues and Limitations*

Bug ID	Description
91146	**Symptom**: An ACL matching on ARP traffic for specific source and destination pairs may not always be enforced. **Scenario**: This issue is observed only when Dynamic ARP Inspection (DAI) is enabled on the Mobility Access Switch and is not limited to any specific Mobility Access Switch model. **Workaround**: Disable DAI when using ACLs matching on ARP for specific source and destination pairs.

Generic Routing Encapsulation (GRE)

Table 34: *Known GRE Issues and Limitations*

Bug ID	Description
87459 88968	**Symptom**: L3 GRE tunnel interfaces toggle between up and down states. **Scenario**: This issue occurs when the L3 GRE tunnel forwarding rate exceeds 40 Kilo packets per second (Kpps). This issue occurs in Mobility Access Switches running ArubaOS 7.3. **Workaround**: None.

IPsec

Table 35: *Known IPsec Issues and Limitations*

Bug ID	Description
73261	**Symptom**: Site-to-site IPsec VPN with transport-mode is not functioning correctly. **Scenario**: This issue is not limited to any specific Mobility Access Switch model or release version. **Workaround**: None.
94073	**Symptom**: The IKE gets deleted when the Mobility Access Switch is used as a NAT box. **Scenario**: This issue is observed when the **session-idle-timeout** value was less than the DPD timer value. This issue is not limited to any specific Mobility Access Switch model or release version. **Workaround**: Use the **crypto-local isakmp dpd idle-timeout <idle_sec>** command to reduce the DPD time to a value lower than the **session-idle-timeout** value configured under the **firewall** command.
103560	**Symptom**: The **crypto isakmp pre-shared key** does not accept special characters to establish an IKE session. **Scenario**: This issue is not limited to any specific Mobility Access Switch model or release version. **Workaround**: None.

IPv6

Table 36: *Known IPv6 Issues and Limitations*

Bug ID	Description
57529	**Symptom**: Copy on IPv6 address does not work as this command is not recognized for IPv6. As a result, the scp/ftp/tftp copy over IPv6 address will not work. **Scenario**: This issue is not limited to any specific Mobility Access Switch model. **Workaround**: Use an IPv4 address instead of an IPv6 or use the WebUI and try the local file management.

Interface

Table 37: *Known Interface Issues and Limitations*

Bug ID	Description
105233	**Symptom**: Though Auto-LACP port-channel is formed initially—with both ports of an Instant AP (IAP) connected to a Mobility Access Switch—the port-channel does not come UP on the Mobility Access Switch after an IAP reboot. **Scenario**: This issue is observed in all IAP versions earlier than Aruba Instant 6.4.3.4_4.2.1.0. This issue is not limited to any specific Mobility Access Switch model to which the IAP is connected. **Workaround**: Execute the `clear lldp neighbor` command for the Mobility Access Switch to recover.

Layer 2 Forwarding

Table 38: *Known Layer 2 Forwarding Issues and Limitations*

Bug ID	Description
68312	**Symptom**: DHCP Offer/ACK messages are not discarded when using DHCP Trust . **Scenario**: This issue is observed when **no trust DHCP** is enabled in a port- security profile and a MAC ACL with a **permit any any rule** is applied to an interface. This issue is not limited to any specific Mobility Access Switch model. **Workaround**: Use a stateless ACL instead of a MAC ACL.
73285	**Symptom**: The Mobility Access Switch does not register a GVRP VLAN on the STP blocked ports. **Scenario**: This issue occurs when there is a change in the STP topology and the blocked ports become forward. The ports first register the VLAN and then the data traffic flow continues. Under these conditions, there is a long delay in resuming the traffic. **Workaround**: None.

Multicast

Table 39: *Known Multicast Issues and Limitations*

Bug ID	Description
63951	**Symptom**: As IPv6 on untrusted port is not supported in this release, Multicast Listener Discovery (MLD) snooping on untrusted port is ignored. Hence, MLD snooping membership table cannot be formed. **Scenario**: This issue is not limited to any specific Mobility Access Switch model. **Workaround**: None.
65314	**Symptom**: The Mobility Access Switch does not send query when there is a change in the Spanning Tree Protocol (STP) topology. This delays the formation of the MLD snooping membership table. **Scenario**: This issue is not limited to any specific Mobility Access Switch model. **Workaround**: None.
77185	**Symptom**: IGMP Snooping entries are removed in 12 seconds before expiry of the age-out timer. **Scenario**: This issue is observed when mutlicast stream is sent over 40Kpps on a L2 GRE tunnel. This issue is not limited to any specific Mobility Access Switch version. **Workaround**: Send multicast stream less than 40 Kpps over a L2 GRE tunnel.

OSPF

Table 40: *Known OSPF Issues and Limitations*

Bug ID	Description
59609	**Symptom**: Layer 3 Manager utilizes more memory and throws an error message during the removal of large number of OSPF routes. **Scenario**: This issue is observed in S3500 running ArubaOS 7.2.0.0. **Workaround**: None.
59738	**Symptom**: Loss of traffic is observed on some advertised OSPF routes. **Scenario**: This issue is observed when it reaches the route capacity limitation (1500). This issue is not limited to any specific Mobility Access Switch model. **Workaround**: None.

Port-Channel

Table 41: *Known Port-Channel Issues and Limitations*

Bug ID	Description
96779	**Symptom:** Link flap occurs when the dynamic ARP Inspection (DAI) parameter is enabled in the port-security profile of a port-channel interface. **Scenario:** This issue occurs in a port-channel interface when the **DAI** parameter is enabled in the port-security profile while the member interfaces are in Independent (LACP - I) state. Due to this issue, traffic impact is noticed for 3 seconds on member interfaces in LACP-I state. This issue is observed in all Mobility Access Switch models running ArubaOS 7.4 or later. **Workaround:** None.
104770	**Symptom:** Connectivity to devices across port channel results in extended request time out when the member port status is changed. **Scenario:** This issue is observed under the following configuration setup: ● On Mobility Access Switch, configure port channel in LACP mode. ● On Cisco switch, configure port channel. ● Configure the link between the two devices as a trunk link. This issue is observed in S2500 running ArubaOS 7.3.2.1. **Workaround:** None.

QoS

Table 42: *Known QoS Issues and Limitations*

Bug ID	Description
79774	**Symptom**: The Mobility Access Switch does not apply QoS remarking or prioritization for traffic in an L2 GRE tunnel. **Scenario**: A QoS profile configured on the interface of the Mobility Access Switch does not prioritize traffic in an L2-GRE tunnel traversing through the same interface. This issue is not limited to any specific Mobility Access Switch model. **Workaround**: None.

Routing

Table 43: *Known Routing Issues and Limitations*

Bug ID	Description
74123	**Symptom**: With Source NAT enabled, no matter what MTU value is assigned to the RVI, packets up to 1784 bytes will be source NAT'ed. Packets larger than this are dropped on the ingress RVI because fragmentation is not supported. Additionally, no matter what MTU is configured, packets leaving the egress RVI are not fragmented. **Scenario**: This issue is not limited to any specific Mobility Access Switch model or release version. **Workaround**: None.
84327	**Symptom**: Traffic continues to be routed even though the ingress Routed Virtual Interface (RVI) is administratively shutdown. **Scenario**: If any Layer 3 unicast traffic is received destined to an RVI that is in an administratively down state, the RVI will route the unicast traffic towards destination even though it is shutdown. This issue is not limited to any specific Mobility Access Switch model. **Workaround**: None.
103209	**Symptom**: The routing table sometimes contains routes for the reserved multicast IP addresses of IGMPv3. **Scenario**: This issue is observed when L3 GRE tunnel is configured with OSPF routing protocol. This issue is limited to Mobility Access Switches running ArubaOS 7.4 **Workaround**: None.
105540	**Symptom**: The peer IP route configured in the crypto map points to the default gateway even though a static route is configured for the peer IP. **Scenario**: This issue is limited to Mobility Access Switches running ArubaOS 7.4. **Workaround**: Configure a higher metric on the VLAN interface through which the peer IP is reachable.
105550	**Symptom**: Sometimes, the connected routes on a VLAN interface may not appear in the routing table after a switchover. **Scenario**: This issue is observed when the VLAN interface with a dynamic IP address is configured on a port channel. This issue is observed on Mobility Access Switches running ArubaOS 7.4. **Workaround**: None.

Security

Table 44: *Known Security Issues and Limitations*

Bug ID	Description
64356	**Symptom**: Router Advertisement (RA) messages are not dropped on untrusted interfaces. **Scenario**: This issue is not limited to any specific Mobility Access Switch model. **Workaround**: None.
67157	**Symptom**: If a phone connected to a Mobility Access Switch port using 802.1X MD5 authentication experiences an Extensible Authentication Protocol (EAP) transaction failure, the Mobility Access Switch sends an EAP-Fail packet every 5 seconds after the failure until the phone restarts 802.1X authentication. **Scenario**: This issue is not limited to any specific Mobility Access Switch model. **Workaround**: None.
67159	**Symptom**: If a phone connected to a Mobility Access Switch port using 802.1X authentication and the AAA profile bound to the interface has a user-derivation rule associated with it, the phone may exchange multiple EAP transactions with the Mobility Access Switch, but may not be able to complete the 802.1X authentication. **Scenario**: This issue is not limited to any specific Mobility Access Switch model. **Workaround**: Remove the **user-derivation-rule** from the AAA profile.
82617	**Symptom**: When Captive Portal authentication is provided by ClearPass Guest, instead of assigning a **Downloadable Role** with Captive Portal redirect, the user gets the default Captive Portal user role defied in the Captive Portal settings. **Scenario**: The issue was observed when the user table has two L3 entries for a same MAC. This issue is not limited to any specific Mobility Access Switch model. **Workaround**: Delete both the stale and valid user entry and perform Captive Portal authentication again.
84802	**Symptom**: A Cisco® IP phone that is assigned a user-role via a device-type User Derivation Rule (UDR) and also 802.1X authenticated (UDR user-role overrides 802.1X user-role), shows the authentication type as **Web** as opposed to **802.1X-Wired** after a switchover of the primary and secondary ArubaStack members. **Scenario**: This issue is not limited to any specific Mobility Access Switch model. **Workaround**: The **show user ip <A.B.C.D>** command incorrectly displays **Web** under the **Auth** column for a Cisco IP phone connected to the Mobility Access Switch. However, the switch assigns the correct role to the Cisco IP phone.
85674	**Symptom**: For some IP phones, the **show station-table** command entry displays the MAC or 802.1X default authentication role of the AAA profile. However, the **show user-table** command entry displays the initial role of the AAA profile. **Scenario**: This issue occurs when an IP phone connected to one of the ports of the Mobility Access Switch, gets an IP address before an L2 authentication completes. This issue is not limited to any specific Mobility Access Switch model. **Workaround:** None.
85682	**Symptom**: When 802.1X authentication is configured with Extensible Authentication Protocol (EAP) termination, even if the user gets blacklisted, it is still able to re-attempt authentication prior to the blacklist timer expiring. **Scenario**: This issue is observed when 802.1X authentication with EAP termination type is set to **eap-tls** and **inner-eap-type** is set to EAP-General Token Type (GTC). This is issue is not limited to any specific Mobility Access Switch model. **Workaround**: None.

SNMP

Table 45: *Known SNMP Issues and Limitations*

Bug ID	Description
82812	**Symptom**: SNMP may not respond temporarily due to a process crash. **Scenario**: This issue is observed while issuing an SNMP GetNext on the ipNetToMediaTable. This issue occurs in Mobility Access Switch running 7.2.0.0 or later and not limited to any specific model. **Workaround**: None.

Stacking

Table 46: *Known Stacking Issues and Limitations*

Bug ID	Description
92339	**Symptom**: Multicast packets in an S1500 ArubaStack are rate limited to 40kpps when IGMP-snooping is enabled on a Rendezvous Point interface. **Scenario**: This issue is limited to S1500 ArubaStack where PIM-Sparse Mode and IGMP-Snooping are enabled on the ArubaStack and affects clients that are not on the same member as that of the interface connecting to the Rendezvous Point. **Workaround**: None.

STP

Table 47: *Known STP Issues and Limitations*

Bug ID	Description
57519	**Symptom**: With Spanning Tree loopguard enabled, an interface will enter LOOP_Inc state if that interface is not receiving any more BPDU. **Scenario**: When the situation happens, restart L2M daemon (such as doing stacking primary failover) may mistakenly bring the interface back to DES/FWD state. **Workaround**: Check your network when an interface enters LOOP_Inc state. Resolve your network problem before doing stacking primary failover or L2M restart. **NOTE:** A typical problem that causes an interface not to receive BPDU happens on the fiber connection in which TX is successful but RX fails.
91798	**Symptom**: After multiple recoveries on a BPDU guard enabled interface, BPDU guard may take a long time to trigger the shutdown operation on the interface. **Scenario**: This issue is observed when a Mobility Access Switch or a connected downstream hub/switch is looped upon itself and if BPDU guard is enabled on the connected interfaces. This issue is not limited to any specific Mobility Access Switch model. **Workaround**: None.
92327	**Symptom**: In an MSTP topology, the interfaces of the Mobility Access Switches may go into an STP boundary state if the STP mode is manipulated. **Scenaro:** This issue is observed if the STP Mode is manually changed from MSTP to PVST and then changed back to MSTP in any one of the Mobility Access Switches connected in a spanning tree environment. This issue is not limited to any specific Mobility Access Switch model. **Workaround**: Remove the MSTP instance from VLAN mapping and add it back.

Switch-Datapath

Table 48: *Known Switch-Datapath Issues and Limitations*

Bug ID	Description
58584	**Symptom**: When an AP is connected to a Mobility Access Switch through a mid-span PoE injector, auto negotiation might fail. **Scenario**: This issue is not limited to any specific Mobility Access Switch model or release version. **Workaround**: Force link speed on the ports.

Switch-Platform

Table 49: *Known Switch-Platform Issues and Limitations*

Bug ID	Description
52196	**Symptom**: **Press 'q' to abort** does not work after issuing the **ping interval <delay_pkts> <host>** command. **Scenario**: This issue is not limited to any specific Mobility Access Switch model. **Workaround**: None.
65618	**Symptom**: The Mobility Access Switch does not synchronize with a Network Time Protocol (NTP) server. **Scenario**: This issue is observed when a NTP server entry is configured prior to configuring or changing the IP address of the egress Routed Virtual Interface (RVI) which is used to contact said NTP server. This issue is not limited to any specific Mobility Access Switch model. **Workaround**: First configure the IP address the RVI and then configure the NTP server address.
65807	**Symptom**: When you create an **eth** ACL with **permit any**, apply the ACL to a user-role, and send IPv6 traffic to untrusted port, the Mobility Access Switch did not create an L2 user nor forward the IPv6 traffic. ArubaOS 7.3 does not support IPv6 on untrusted port. **Scenario**: This issue is not limited to any specific Mobility Access Switch model. **Workaround**: None.
68091	**Symptom**: An interface is operationally down. **Scenario**: This issue occurs when an Ethernet OAM failure may still transmit data and other control packets. **Workaround**: Enable STP on the interface or configure the link as a port-channel member.
86723	**Symptom**: Copying files from any source to an external USB flash drive or the local flash drive using the CLI does not show the transfer progress and there is no option to abort the transfer. **Scenario**: This issue is not limited to any specific Mobility Access Switch model. **Workaround**: None.
86853	**Symptom**: Copying a raw image from a USB connected to the primary stack member copies the image only on primary and not all stack members. **Scenario:** This issue occurs on Mobility Access Switches running ArubaOS 7.3. **Workaround**: None.
86857	**Symptom**: Users cannot exit from Quick-Setup in the CLI using CTRL+C. **Scenario**: This issue is observed in an ArubaStack when the console port is redirected from a secondary or line card member. This issue is not limited to any specific Mobility Access Switch model. **Workaround**: Connect the console port to the primary member of the ArubaStack if using Quick-Setup.

Table 49: *Known Switch-Platform Issues and Limitations*

Bug ID	Description
90167	**Symptom**: AP-220 Series and AP-130 Series may not get powered up when connected to a Mobility Access Switch. **Scenario**: This issue is observed when both ethernet ports of the access point are connected to the PoE ports of the same Mobility Access Switch. This issue is limited to PoE models of Mobility Access Switch. **Workaround**: Remove the **poe-profile** (i.e. disable PoE) from one of the two ports of the Mobility Access Switch that are connected to the access point.
90231	**Symptom**: Cisco IP phones utilizing pre-standard PoE (also known as legacy power) may lose power after being operational for a long time. **Scenario**: This issue is limited to PoE models of the Mobility Access Switch. **Workaround**: Disconnect the phone for a few minutes and reconnect it.
99827	**Symptom:** Sometimes, the following I2C error messages are observed in the output of **show log system** command due to an internal processor issue: - **Mar 4 08:22:17 KERNEL: 2:i2c_xls_wait_for_idle: i2c line is busy (status: 0003)** - **Mar 4 08:22:17 KERNEL: 2:Unable to select i2c mux channel 6** - **Mar 4 08:22:17 KERNEL: 2:Hard reset to i2c mux on bus 0 address 0x70** - **Mar 4 08:22:17 KERNEL: 2:Unable to access hw sensor on bus 9 address 0x2d** **Scenario:** This issue is very rarely observed and is not limited to any specific Mobility Access Switch model or release version. **Workaround:** Reload the box.
103600	**Symptom**: Uplink port status LED remains in **On** state even after the link is locally shutdown with 1G SFP. **Scenario**: This issue is observed only when a 7205 Mobility Controller is connected to the uplink port of the Mobility Access Switch. **Workaround**: None.
103713	**Symptom**: Kernel panic is observed in the tar logs of one of the members of the ArubaStack. **Scenario**: This issue is observed when an IGMPv2 client joins and disconnects from a group where IGMPv3 is enabled. This issue is limited to ArubaStack running ArubaOS 7.4. **Workaround**: None.
103793	**Symptom**: All APs associated to a Mobility Access Switch go down, and the system status LED on the Mobility Access Switch turns blinking amber indicating a major alarm. **Scenario**: This issue is observed during lightening, thunder storm, or if another PSE is providing inline power to the Mobility Access Switch. This issue is observed in S2500 and S1500-24/48P Mobility Access Switches running ArubaOS 7.2.2 or earlier versions. **Workaround**: Upgrade the Mobility Access Switch to ArubaOS 7.3.2.1 to benefit from many PoE features intruduced in this release version.
105354	**Symptom:** A Mobility Access Switch stops responding and reboots. The log files for the event listed the reason as **Hard Watchdog Reset**. **Scenario:** This issue is observed in S3500 running ArubaOS 7.3.1.0. **Workaround:** None.

WebUI

Table 50: *Known WebUI Issues and Limitations*

Bug ID	Description
106087	**Symptom**: Copying an image using TFTP from the WebUI does not upgrade the image on an ArubaStack. **Scenario**: This issue occurs only when the TFTP copy is tried from the WebUI for an ArubaStack running ArubaOS 7.3.x or later versions. **Workaround**: Copy the image using the **copy tftp** command in the CLI.
107809	**Symptom**: The following error message appears when downloading logs from the Mobility Access Switch using the WebUI: **can't query: TimeoutError: DOM Exception 23** **Scenario**: This issue occurs only when Safari is used as the browser for the WebUI. This issue is limited to Mobility Access Switches running ArubaOS 7.4 or later versions. **Workaround**: Use browsers such as Google Chrome or Mozilla Firefox to access the WebUI.

Other Limitations

The limitations applicable to ArubaOS release versions are as follows:

Maximum DHCP Leases Per Platform

The following table provides the maximum number of DHCP leases supported per Mobility Access Switch platform:

Table 51: *DHCP Lease Limits*

Switch Platform	Maximum number of DHCP Leases Supported
S1500	512
S2500	512
S3500	512

Exceeding these limits may result in excessive CPU utilization and unpredictable negative impact on the switch operations.

Issues Under Investigation

The following are the issues observed in ArubaOS 7.4.1.3 and are under investigation. The associated Bug IDs are included.

DPA

Table 52: *DPA Issues Under Investigation*

Bug ID	Description
128218	**Symptom**: When a Mobility Access Switch is configured as tunneled node and multiple ports are mapped with the same tunneled node profile—each node pointing to primary and the backup controller—all the tunneled nodes go DOWN whenever there is network fluctuation. The status of the tunneled nodes change to `in progress´ and the nodes fail to reestablish the connection.

Stacking

Table 53: *Stacking Issues Under Investigation*

Bug ID	Description
99121	**Symptom**: Error octets are seen in Received Statistics (Rx counters) on the stack ports of S2500 and S3500 Mobility Access Switches.
133881	**Symptom**: In a 5-member stack, while trying to save the running configuration on a primary switch—S3500 Mobility Access Switch that runs ArubaOS 7.4.1.0—the operation fails with the following error: **May not have enough memory for operation**.

Chapter 3
Upgrade Procedures

This chapter details the Mobility Access Switch software upgrade procedures. To optimize your upgrade experience and ensure a successful upgrade, read all the information in this chapter before upgrading and follow all the procedures carefully.

Topics in this chapter include:

Important Points to Remember

You should create a permanent list of the following information for future use:

- Best practice is to upgrade during a maintenance window. This will limit the troubleshooting variables.
- Resolve any existing issues (consistent or intermittent) before you upgrade.
- List the devices in your infrastructure that are used to provide your wireless users with connectivity (Core switches, radius servers, DHCP servers, firewall, for example).
- Always upgrade the non-boot partition first. If something happens during upgrade, you can switch back to the boot partition. Upgrading the non-boot partition gives you a smoother downgrade path should it be required.
- If you have removed the default stacking interfaces (ports 0/1/2 and 0/1/3) from 7.0.x but plan to use them for stacking purposes after upgrading to ArubaOS 7.3, you must reconfigure them for stacking.

Before You Upgrade

Run the following checklist before installing a new image on the Mobility Access Switch:

- Ensure that you have at least 60 MB of free flash space (**show storage** command).
- Run the **tar crash** command to ensure that there are no "process died" files clogging up memory and FTP/TFTP the files to another storage device. To clean up any crash core file, use the **tar clean crash** command.
- Remove all unnecessary saved files from flash (**delete filename** command).

Save Your Configuration

Before upgrading, save your configuration and back up your Mobility Access Switch data files. Saving your configuration will retain the admin and enable passwords in the proper format.

Saving the Configuration in the WebUI

1. Click the **Configuration** tab.
2. Click the **Save Configuration** button at the top of the screen.

Saving the Configuration in the CLI

Enter the following command in either the enable or configuration mode:

```
(host) #write memory
```

Upgrading to ArubaOS 7.4.1.3

Read all the following information before you upgrade. Download the latest software image from the Aruba Customer Support web site.

There are three ways to upgrade your software image:

- Upgrading from the WebUI on page 60
- Upgrading from the Command Line Interface on page 60
- Upgrading from your USB using the LCD on page 61

CAUTION

If you are upgrading from 7.0.x to 7.3 and are going to create a stack, each Mobility Access Switch in the stack must be upgraded to ArubaOS 7.3 before forming the stack.

Upgrading from the WebUI

The following steps describe how to install the Aruba software image from a PC or workstation using the WebUI on the Mobility Access Switch. You can also install the software image from a TFTP or FTP server using the same WebUI page.

1. Upload the new software image to a PC or workstation on your network.
2. Log in to the WebUI from the PC or workstation.
3. Navigate to the **Maintenance > Image Management** page. Select the **Upgrade using local file** option, then click **Browse** to navigate to the image file on your PC or workstation.
4. Determine which partition will be used to hold the new software image. Best practice is to load the new image onto the non-boot partition. To see the current boot partition, navigate to the **Maintenance > Boot Parameters** page.
5. Select **Yes** in the **Reboot after upgrade** field to reboot after upgrade.
6. Click **Upgrade Image**. The image, once copied to the ArubaStack primary, will be pushed down to every stack member.
7. When the software image is uploaded to the Mobility Access Switch, a popup appears. Click **OK** to reload the entire stack. The boot process starts automatically within a few seconds.
8. When the boot process is complete, log in to the WebUI and navigate to the **Monitoring > Summary** page to verify the upgraded code version.
9. Select the **Configuration** tab.
10. Click **Save Configuration** at the top of the screen to save the new configuration file header.

Upgrading from the Command Line Interface

The following steps describe how to install the ArubaOS software image using the CLI on the Mobility Access Switch. You need a FTP/TFTP server reachable from the Mobility Access Switch you are upgrading.

1. Upload the new software image to your FTP/TFTP server on your network.
2. Execute the **ping** command to verify the network connection from the target Mobility Access Switch to the FTP/TFTP server:

   ```
   (host) # ping <tftphost>
   ```

NOTE

A placeholder file with the destination filename and proper write permissions must exist on the FTP/TFTP server prior to executing the copy command.

3. Determine which partition to load the new software image. Best practices is to load the new image onto the backup partition (the non-boot partition). To view the partitions, use the **show image version** command.
4. Use the **copy** command to load the new image onto the Mobility Access Switch. The image, after being copied to the stack Primary, will be pushed down to every stack member:

   ```
   (host) # copy ftp: <ftphost> <ftpusername> <image filename> system: partition 1
   ```

   or

   ```
   (host) # copy tftp: <tftphost> <image filename> system: partition 1
   ```

NOTE

When using the **copy** command to load a software image, the specified partition automatically becomes active (default boot partition) the next time the Mobility Access Switch is rebooted. There is no need to manually select the partition.

5. Execute the **show image version member all** command to verify if the new image is loaded:

   ```
   (host) #show image version member all
   ```

6. Reload the entire stack:

   ```
   (host) # reload
   ```

7. Execute the **show version member all** command to verify if the reload and upgrade is complete.

   ```
   (host) #show version member all
   ```

8. Execute the **write memory** command to save the new configuration file header.

Upgrading from your USB using the LCD

CAUTION

If you are upgrading from ArubaOS 7.0.2.0 to ArubaOS 7.1.0.0 or greater, you cannot upgrade from an external USB device using the LCD screen. Use either the WebUI or the CLI to complete your upgrade.

The Mobility Access Switch is equipped with an LCD panel that displays a variety of information about the status of the Mobility Access Switch and provides a menu that allows you to do basic operations such as initial setup and reboot. The LCD panel displays two lines of text.

Use the upper right **Menu** button to navigate through LCD functions and the lower right **Enter** button to select (or enter) an LCD function. The active line, in the LCD panel, is indicated by an arrow.

Use a USB device to transfer the upgrade image:

1. Create a folder named arubaimage on your USB device.
2. Using your laptop, copy the new image from the support site to your USB device's folder **arubaimage**.

NOTE

You must download the new image to the **arubaimage** folder or the image will not properly upload to the Mobility Access Switch.

3. Insert your USB device into the rear USB port (next to the console port) of your Mobility Access Switch.
4. Press the **Menu** button until you reach the **Maintenance** function.
5. Press the **Enter** button to enter the maintenance function.
6. Press the **Enter** button at **Upgrade Image** function.
7. Press the **Menu** button to locate the partition you want to upgrade.

   ```
   partition 0
   partition 1
   ```

   Then press the **Enter** button to select the partition to upgrade.

Always upgrade the non-boot partition first. Upgrading the non-boot partition gives you a smoother downgrade path should it be required.

8. Press the **Enter** button again to confirm the partition you are upgrading (or press the Menu button to exit).

```
y: Enter button
n: Menu button
```

9. The LCD displays an a upgrade in process acknowledgement:

```
Upgrading...
```

   When the upgrade is complete, the LCD displays the message:

```
Reload to boot from new image
```

When loading a software image, the specified partition automatically becomes active (default boot partition) the next time the Mobility Access Switch is rebooted. There is no need to manually select the partition.

10. From the command line, execute **show image version member all** to view the partitions:
11. Execute the following command to reload the stack:

```
(host) # reload
```

12. Execute the **show version member all** command to verify if the reload and upgrade is complete.

```
(host) #show version member all
```

13. Execute the **write memory** command to save the new configuration file header.

After completing the upgrade, your system will create a configuration file called **default.cfg.<timestamp>**. This file is your configuration at the time of upgrade. Another file called **default.cfg** is created , which is your configuration post-upgrade.

Downgrading after an Upgrade

If necessary, you can roll-back to the previous version of ArubaOS on your Mobility Access Switch using the procedure given below.

Note the following points before downgrading ArubaOS:

- Save your configuration file before and after completing your downgrade
- MSTP will be disabled upon downgrading.

Before you reboot the Mobility Access Switch with the pre-upgrade software version, you must perform the following steps:

1. Set the Mobility Access Switch to boot with the previously-saved configuration file. By default, ArubaOS creates a file called **original.cfg** upon upgrade. This file can be used instead of a previously-saved configuration file in case you did not save your configuration before upgrade.
2. Use the **dir** command to confirm your saved configuration files or **original.cfg**.

```
(host)#dir
-rw-r--r-- 1 root root 3710 Nov 7 14:35 default.cfg
-rw-r--r-- 2 root root 3658 Nov 7 14:35 default.cfg.2011-11-07_1
-rw-r--r-- 2 root root 3658 Nov 7 14:35 original.cfg
```

3. Use the boot **config-file <filename>** command to select the configuration file you will boot from after downgrading.

```
(host)#boot config-file original.cfg
```

4. Confirm that you have selected the correct file using the **show boot** command.

```
(host)#show boot
Config File: original.cfg
Boot Partition: PARTITION 0
```

5. Set the Mobility Access Switch to boot from the system partition that contains the previously running image.
6. Execute the **write memory** command after the downgrade to save your configuration

Before You Call Your Support Provider

Before you place a call to Technical Support, follow the steps listed below:

1. Provide a detailed network topology (including all the devices in the network between the user and the Mobility Access Switch with IP addresses and Interface numbers if possible).
2. Provide the Mobility Access Switch logs and output of the **show tech-support** command.
3. Provide the syslog file of the Mobility Access Switch at the time of the problem.

   Best practices strongly recommend that you consider adding a syslog server if you do not already have one to capture from the Mobility Access Switch.
4. Let the support person know if this is a new or existing installation. This helps the support team to determine the troubleshooting approach, depending on whether you have:
   - an outage in a network that worked in the past
   - a network configuration that has never worked
   - a brand new installation
5. Let the support person know if there are any recent changes in your network (external to the Mobility Access Switch) or any recent changes to your Mobility Access Switch configuration.
6. If there was a configuration change, list the exact configuration steps and commands used.
7. Provide the date and time (if possible) when the problem first occurred.
8. If the problem is reproducible, list the exact steps taken to re-create the problem.
9. Provide the Mobility Access Switch site access information, if possible.