Configure MAC

Feature History for MAC

This table provides release and related information for features explained in this module.

These features are available on all releases subsequent to the one they were introduced in, unless noted otherwise.

Release	Feature Name and Description	Supported Platform
Cisco IOS XE 17.18.1	A MAC address is a Layer 2 address used to identify individual devices on a local area network (LAN). It is a globally unique identifier hardcoded into the network adapter by the manufacturer.	Cisco C9350 Series Smart Switches Cisco C9610 Series Smart Switches

MAC Address and MAC Address Table Management

MAC Address Table Management (MATM) refers to the processes by which a Layer 2 switch builds, maintains, and uses its MAC address table to efficiently forward Ethernet frames.

A Layer 2 switch's primary function is to forward frames based on MAC addresses. To do this efficiently, the switch maintains a MAC address table that maps MAC addresses to specific switch ports and VLANs. MATM encompasses the mechanisms of how entries are added, updated, and removed from this table.

Key MATM processes:

- **MAC address learning:**
  - When a switch receives a frame on a port, it inspects the **source MAC address** of the frame.
  - If the source MAC address is not already in the MAC address table, or if it is in the table but associated with a different port, the switch adds or updates the entry, associating the MAC address with the incoming port and VLAN.

    - This allows the switch to know which port to use when sending frames *to* that MAC address in the future.

- **Frame forwarding:**
    - When a switch receives a frame, it inspects the **destination MAC address**.
    - The switch then looks up this destination MAC address in its MAC address table.
    - **Known destination:** If the destination MAC address is found and associated with a specific port (different from the incoming port), the switch forwards the frame only out of that specific port. This is called **unicast forwarding**.
    - **Unknown destination (flooding):** If the destination MAC address is not found in the table, the switch floods the frame out of all ports within the same VLAN, except the port on which it was received. This ensures the frame reaches its intended destination, and the destination's response will allow the switch to learn its MAC address.
    - **Same port (filtering):** If the destination MAC address is found and associated with the same port on which the frame was received, the switch drops (filters) the frame. This prevents unnecessary traffic from being sent back out the same segment.
    - **Broadcast/multicast:** Broadcast frames (destination MAC FF:FF:FF:FF:FF:FF) and unknown multicast frames are always flooded out of all ports within the same VLAN, except the incoming port.

- **MAC address aging:**
    - MAC address entries in the table are not permanent. Each entry has an associated aging timer.
    - If the switch does not receive any frames from a particular source MAC address on a specific port within the aging time, that entry is removed from the MAC address table.
    - Aging ensures that the table remains current and does not become cluttered with stale entries from devices that are no longer connected or have moved to a different port. The default aging time is typically 300 seconds (5 minutes).

Benefits of MATM:

- **Efficient frame delivery:** Frames are sent only to the necessary destination port, reducing unnecessary traffic on other segments.
- **Reduced collisions (in half-duplex environments):** By segmenting collision domains, switches improve network performance.
- **Enhanced security:** Limits the exposure of traffic to only the relevant devices, although not a security feature in itself.
- **Dynamic adaptation:** The table automatically updates as devices connect, disconnect, or move within the network.

Restrictions and limitations of MAC and MATM

Both MAC addresses and MATM processes have inherent restrictions and limitations:

MAC address limitations:

- **Flat address space:** MAC addresses provide no hierarchical information for routing beyond the local segment. They do not indicate network location or subnet, making them unsuitable for large-scale routing.
- **Burned-in nature:** MAC addresses are generally hardcoded into the network interface controller (NIC) and are not easily changed by end-users. While they can be spoofed (changed programmatically), this is typically a violation of network policy.
- **Security vulnerabilities:** MAC addresses are not inherently secure. They are susceptible to spoofing, where a malicious actor impersonates another device by using its MAC address, and to MAC flooding attacks, which can overwhelm a switch's MAC address table.
- **Scalability in broadcast domains:** While unique, managing an extremely large number of MAC addresses within a single broadcast domain (VLAN) can lead to very large CAM tables, consuming significant switch memory and potentially impacting lookup performance.

MATM limitations:

- **CAM table size limits:** Switches have finite Content Addressable Memory (CAM) table sizes. If the number of learned MAC addresses exceeds this limit, the switch may resort to flooding unknown unicast frames out of all ports in a VLAN, effectively behaving like a hub for those frames. This impacts performance and can be exploited for security breaches.
- **MAC flooding attacks:** Malicious actors can deliberately send frames with many unique source MAC addresses to rapidly fill the CAM table, forcing the switch into a flooding state.
- **MAC address flapping:** A MAC address learned on multiple ports (or rapidly moving between ports) indicates a network loop or misconfiguration. While switches detect and log these events, constant flapping can lead to CAM table instability, CPU utilization spikes, and service disruptions.
- **Aging time impact:** An aging time that is too short can cause frequent re-learning of MAC addresses, increasing switch overhead. An aging time that is too long can keep stale entries in the table, leading to frames being sent to incorrect ports until the entry ages out or is re-learned.
- **No Layer 3 awareness:** MATM operates purely at Layer 2. It does not understand IP addresses, higher-layer protocols, or network topology beyond direct connectivity.

Types of MAC addresses

This section describes different classifications of MAC addresses and various methods for managing MAC address entries within the switch's table.

Types of MAC addresses:

- **Unicast MAC addresses:** These are unique identifiers assigned to a single network interface. They are the most common type and are used for one-to-one communication. The least significant bit of the first octet is 0.
- **Multicast MAC addresses:** These addresses are used for one-to-many communication within a local network segment. Frames sent to a multicast MAC address are received by all devices configured to listen for that specific multicast group. The least significant bit of the first octet is 1. IPv4 multicast addresses typically map to MAC addresses in the 01:00:5E:XX:XX:XX range.
- **Broadcast MAC addresses:** The broadcast MAC address is FF:FF:FF:FF:FF:FF. Frames sent to this address are delivered to all devices within the same broadcast domain (VLAN).

- **Universally administered addresses (UAAs):** These are the most common type of MAC address. They are burned into the NIC by the manufacturer and are globally unique. The OUI identifies the manufacturer.
- **Locally administered addresses (LAAs):** These are MAC addresses that can be manually configured or assigned by a network administrator, overriding the burned-in address. LAAs are not globally unique and must be managed carefully to avoid conflicts within a network.

Types of MAC address table entries (MATM methods):

- **Dynamic MAC addresses:** These are the most common type of entry. The switch learns them automatically when it receives a frame from a source MAC address on a specific port. Dynamic entries have an aging timer and are removed if no traffic is seen from that MAC address within the aging period.
- **Static MAC addresses:** These entries are manually configured by an administrator to permanently associate a MAC address with a specific port and VLAN. Static entries do not age out and remain in the CAM table until they are manually removed. They are useful for critical devices that should always be accessible on a specific port.
- **Sticky MAC addresses:** These combine aspects of dynamic and static learning. When enabled on a port, the switch dynamically learns MAC addresses and converts them into sticky entries. These sticky entries are then stored in the running configuration. If the running configuration is saved to the startup configuration, these entries persist across reboots. If a device associated with a sticky MAC address moves to another port, it triggers a security violation.
- **Secure MAC addresses (Port Security):** These are MAC addresses that are learned or configured under the port security feature. Port security allows you to limit the number of MAC addresses that can be learned on a port and define actions to take if a violation occurs (e.g., shutdown the port, restrict traffic). Secure MAC addresses can be dynamic, static, or sticky.

Configuration steps for MAC

-

Configuring global MAC address table aging time

Sticky MAC addresses are a port security feature that prevents unauthorized devices from connecting to a switch port. This feature controls which specific devices are allowed on certain ports.

Port security with the sticky feature enabled allows a switch port to learn MAC addresses dynamically. The switch then stores these learned MAC addresses in its running configuration. This makes the MAC addresses persistent.

If a device with a new MAC address attempts to connect to a port with sticky MAC addresses, a security violation occurs. A security violation triggers predefined actions. These actions include:

- **Shutdown:** The port enters an error-disabled state. Manual intervention is required to reactivate the port.
- **Restrict:** The switch drops packets from the unauthorized source MAC address. A violation counter increments, and a Simple Network Management Protocol (SNMP) trap message is sent to the administrator. Authorized traffic continues to flow.
- **Protect:** The switch drops packets from the unauthorized source MAC address. No log messages, SNMP traps, or violation counter increments occur.

The feature also supports configuring a maximum number of sticky MAC addresses per port. Reaching this limit with a new MAC address triggers a violation.

Benefits

- **Enhanced security:** Prevents unauthorized device access to the network.
- **Network stability:** Reduces disruptions from unauthorized MAC address changes.
- **Simplified management:** Automates MAC address learning and persistence, reducing manual configuration.
- **Persistence:** Learned MAC addresses remain associated with the port across reboots (if saved to startup configuration).

Procedure

	Command or Action	Purpose
Step 1	enable **Example:** `Switch> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 3	interface type number **Example:** `Switch(config)# interface GigabitEthernet0/2`	Specifies the interface for port security, and enters interface configuration mode.
Step 4	switchport mode access **Example:** `Switch(config-if)# switchport mode access`	Configures the interface as an access port. Port security is typically configured on access ports.
Step 5	switchport port-security **Example:** `Switch(config-if)# switchport port-security`	Enables port security on the interface.
Step 6	switchport port-security maximum value **Example:** `Switch(config-if)# switchport port-security maximum 5`	Configures the maximum number of secure MAC addresses allowed on the port. For value, the range is typically 1 to a maximum supported by the switch model.
Step 7	switchport port-security mac-address sticky **Example:** `Switch(config-if)# switchport port-security mac-address sticky`	Enables sticky learning on the port. Dynamically learned MAC addresses are converted to sticky secure MAC addresses

	Command or Action	Purpose
Step 8	switchport port-security violation {protect | restrict | shutdown} **Example:** `Switch(config-if)# switchport port-security violation shutdown`	Configures the action to take when a security violation occurs. * protect: Drops packets from unknown sources until a secure MAC address is removed. * restrict: Drops packets from unknown sources, sends an SNMP trap, and increments a violation counter. * shutdown: Shuts down the interface, sends an SNMP trap, and increments a violation counter. The interface remains shut down until manually re-enabled or error-disabled recovery is configured.
Step 9	end **Example:** `Switch(config-if)# end`	Returns to priviledged EXEC mode
Step 10	copy running-config startup-config **Example:** `Switch# copy running-config startup-config`	Saves the running configuration to the startup configuration

Configuring a static MAC address entry

Sticky MAC addresses are a port security feature that prevents unauthorized devices from connecting to a switch port. This feature controls which specific devices are allowed on certain ports.

Port security with the sticky feature enabled allows a switch port to learn MAC addresses dynamically. The switch then stores these learned MAC addresses in its running configuration. This makes the MAC addresses persistent.

If a device with a new MAC address attempts to connect to a port with sticky MAC addresses, a security violation occurs. A security violation triggers predefined actions. These actions include:

- **Shutdown:** The port enters an error-disabled state. Manual intervention is required to reactivate the port.
- **Restrict:** The switch drops packets from the unauthorized source MAC address. A violation counter increments, and a Simple Network Management Protocol (SNMP) trap message is sent to the administrator. Authorized traffic continues to flow.
- **Protect:** The switch drops packets from the unauthorized source MAC address. No log messages, SNMP traps, or violation counter increments occur.

The feature also supports configuring a maximum number of sticky MAC addresses per port. Reaching this limit with a new MAC address triggers a violation.

Benefits

- **Enhanced security:** Prevents unauthorized device access to the network.
- **Network stability:** Reduces disruptions from unauthorized MAC address changes.
- **Simplified management:** Automates MAC address learning and persistence, reducing manual configuration.
- **Persistence:** Learned MAC addresses remain associated with the port across reboots (if saved to startup configuration).

Procedure

	Command or Action	Purpose
Step 1	enable **Example:** `Switch> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 3	interface type number **Example:** `Switch(config)# interface GigabitEthernet0/2`	Specifies the interface for port security, and enters interface configuration mode.
Step 4	switchport mode access **Example:** `Switch(config-if)# switchport mode access`	Configures the interface as an access port. Port security is typically configured on access ports.
Step 5	switchport port-security **Example:** `Switch(config-if)# switchport port-security`	Enables port security on the interface.
Step 6	switchport port-security maximum value **Example:** `Switch(config-if)# switchport port-security maximum 5`	Configures the maximum number of secure MAC addresses allowed on the port. For value, the range is typically 1 to a maximum supported by the switch model.
Step 7	switchport port-security mac-address sticky **Example:** `Switch(config-if)# switchport port-security mac-address sticky`	Enables sticky learning on the port. Dynamically learned MAC addresses are converted to sticky secure MAC addresses
Step 8	switchport port-security violation {protect | restrict | shutdown}	Configures the action to take when a security violation occurs.

	Command or Action	Purpose
	Example: `Switch(config-if)# switchport port-security violation shutdown`	* protect: Drops packets from unknown sources until a secure MAC address is removed. * restrict: Drops packets from unknown sources, sends an SNMP trap, and increments a violation counter. * shutdown: Shuts down the interface, sends an SNMP trap, and increments a violation counter. The interface remains shut down until manually re-enabled or error-disabled recovery is configured.
Step 9	end **Example:** `Switch(config-if)# end`	Returns to priviledged EXEC mode
Step 10	copy running-config startup-config **Example:** `Switch# copy running-config startup-config`	Saves the running configuration to the startup configuration

Configuring port security with sticky MAC addresses

Sticky MAC addresses are a port security feature that prevents unauthorized devices from connecting to a switch port. This feature controls which specific devices are allowed on certain ports.

Port security with the sticky feature enabled allows a switch port to learn MAC addresses dynamically. The switch then stores these learned MAC addresses in its running configuration. This makes the MAC addresses persistent.

If a device with a new MAC address attempts to connect to a port with sticky MAC addresses, a security violation occurs. A security violation triggers predefined actions. These actions include:

- **Shutdown:** The port enters an error-disabled state. Manual intervention is required to reactivate the port.
- **Restrict:** The switch drops packets from the unauthorized source MAC address. A violation counter increments, and a Simple Network Management Protocol (SNMP) trap message is sent to the administrator. Authorized traffic continues to flow.
- **Protect:** The switch drops packets from the unauthorized source MAC address. No log messages, SNMP traps, or violation counter increments occur.

The feature also supports configuring a maximum number of sticky MAC addresses per port. Reaching this limit with a new MAC address triggers a violation.

Benefits

- **Enhanced security:** Prevents unauthorized device access to the network.
- **Network stability:** Reduces disruptions from unauthorized MAC address changes.

- **Simplified management:** Automates MAC address learning and persistence, reducing manual configuration.
- **Persistence:** Learned MAC addresses remain associated with the port across reboots (if saved to startup configuration).

Procedure

	Command or Action	Purpose
Step 1	enable **Example:** `Switch> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 3	interface type number **Example:** `Switch(config)# interface GigabitEthernet0/2`	Specifies the interface for port security, and enters interface configuration mode.
Step 4	switchport mode access **Example:** `Switch(config-if)# switchport mode access`	Configures the interface as an access port. Port security is typically configured on access ports.
Step 5	switchport port-security **Example:** `Switch(config-if)# switchport port-security`	Enables port security on the interface.
Step 6	switchport port-security maximum value **Example:** `Switch(config-if)# switchport port-security maximum 5`	Configures the maximum number of secure MAC addresses allowed on the port. For value, the range is typically 1 to a maximum supported by the switch model.
Step 7	switchport port-security mac-address sticky **Example:** `Switch(config-if)# switchport port-security mac-address sticky`	Enables sticky learning on the port. Dynamically learned MAC addresses are converted to sticky secure MAC addresses
Step 8	switchport port-security violation {protect | restrict | shutdown} **Example:** `Switch(config-if)# switchport port-security violation shutdown`	Configures the action to take when a security violation occurs. * protect: Drops packets from unknown sources until a secure MAC address is removed.

	Command or Action	Purpose
		* restrict: Drops packets from unknown sources, sends an SNMP trap, and increments a violation counter. * shutdown: Shuts down the interface, sends an SNMP trap, and increments a violation counter. The interface remains shut down until manually re-enabled or error-disabled recovery is configured.
Step 9	end **Example:** `Switch(config-if)# end`	Returns to priviledged EXEC mode
Step 10	copy running-config startup-config **Example:** `Switch# copy running-config startup-config`	Saves the running configuration to the startup configuration

Configuring MAC address move notification

Sticky MAC addresses are a port security feature that prevents unauthorized devices from connecting to a switch port. This feature controls which specific devices are allowed on certain ports.

Port security with the sticky feature enabled allows a switch port to learn MAC addresses dynamically. The switch then stores these learned MAC addresses in its running configuration. This makes the MAC addresses persistent.

If a device with a new MAC address attempts to connect to a port with sticky MAC addresses, a security violation occurs. A security violation triggers predefined actions. These actions include:

- **Shutdown:** The port enters an error-disabled state. Manual intervention is required to reactivate the port.
- **Restrict:** The switch drops packets from the unauthorized source MAC address. A violation counter increments, and a Simple Network Management Protocol (SNMP) trap message is sent to the administrator. Authorized traffic continues to flow.
- **Protect:** The switch drops packets from the unauthorized source MAC address. No log messages, SNMP traps, or violation counter increments occur.

The feature also supports configuring a maximum number of sticky MAC addresses per port. Reaching this limit with a new MAC address triggers a violation.

Benefits

- **Enhanced security:** Prevents unauthorized device access to the network.
- **Network stability:** Reduces disruptions from unauthorized MAC address changes.
- **Simplified management:** Automates MAC address learning and persistence, reducing manual configuration.

- **Persistence:** Learned MAC addresses remain associated with the port across reboots (if saved to startup configuration).

Procedure

	Command or Action	Purpose
Step 1	enable **Example:** `Switch> enable`	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 3	interface type number **Example:** `Switch(config)# interface GigabitEthernet0/2`	Specifies the interface for port security, and enters interface configuration mode.
Step 4	switchport mode access **Example:** `Switch(config-if)# switchport mode access`	Configures the interface as an access port. Port security is typically configured on access ports.
Step 5	switchport port-security **Example:** `Switch(config-if)# switchport port-security`	Enables port security on the interface.
Step 6	switchport port-security maximum value **Example:** `Switch(config-if)# switchport port-security maximum 5`	Configures the maximum number of secure MAC addresses allowed on the port. For value, the range is typically 1 to a maximum supported by the switch model.
Step 7	switchport port-security mac-address sticky **Example:** `Switch(config-if)# switchport port-security mac-address sticky`	Enables sticky learning on the port. Dynamically learned MAC addresses are converted to sticky secure MAC addresses
Step 8	switchport port-security violation {protect | restrict | shutdown} **Example:** `Switch(config-if)# switchport port-security violation shutdown`	Configures the action to take when a security violation occurs. * protect: Drops packets from unknown sources until a secure MAC address is removed. * restrict: Drops packets from unknown sources, sends an SNMP trap, and increments a violation counter.

	Command or Action	Purpose
		* shutdown: Shuts down the interface, sends an SNMP trap, and increments a violation counter. The interface remains shut down until manually re-enabled or error-disabled recovery is configured.
Step 9	end **Example:** `Switch(config-if)# end`	Returns to priviledged EXEC mode
Step 10	copy running-config startup-config **Example:** `Switch# copy running-config startup-config`	Saves the running configuration to the startup configuration