

Advanced Settings for Access Control

Advanced access control policy settings typically require little or no modification. The default settings are appropriate for most deployments.

- Requirements and prerequisites for advanced settings, on page 1
- Configure advanced settings for the access control policy, on page 2

Requirements and prerequisites for advanced settings

This reference provides the requirements and prerequisites needed to configure advanced settings for access control policy, including model support, domain compatibility, and user role permissions.

Model support

Any

Supported domains

Any

User roles

- Administrator
- Access Admin
- Network Admin
- You can define custom user roles to differentiate between the intrusion configuration in access control policy and rules and the rest of the access control policy and rules. Using these permissions, you can separate the responsibilities of your network administration team and your intrusion administration teams. The existing pre-defined user roles that included the Modify Access Control Policy permission support all sub-permissions; you need to create your own custom roles if you want to apply granular permissions. The granular permissions are:
  - **Policies** > **Security policies** > **Access Control** and choose the **Access Control Policy** > **Modify Access Control Policy** > **Modify Threat Configuration** allows the selection of intrusion policy, variable set, and file policy in a rule, the configuration of the advanced options for network analysis and intrusion policies, the configuration of the Security Intelligence policy for the access control

policy, and intrusion actions in the policy default action. If a user has this option only, the user can modify no other part of the policy or rule.

- **Modify Remaining Access Control Policy Configuration** controls the ability to edit all other aspects of the policy.

Configure advanced settings for the access control policy

Configure advanced settings to customize access control policy behavior for specific security requirements and performance optimization.

Advanced access control policy settings typically require little or no modification. The default settings are appropriate for most deployments. Note that many of the advanced preprocessing and performance options in access control policies may be modified by rule updates as described in *Update Intrusion Rules* in the *Cisco Secure Firewall Management Center Administration Guide* .

Caution See Configurations that restart the snort process when deployed or activated for a list of advanced setting modifications that restart the Snort process, temporarily interrupting traffic inspection. Whether traffic drops during this interruption or passes without further inspection depends on how the assigned device handles traffic. Refer to Snort restart traffic behavior.

Before you begin

Inheriting settings from a parent policy:

- If the access control policy has a base policy, you can elect to inherit settings from the base policy. Select **Inherit from (*base policy*)** for each setting group where you want to use the parent policy's settings. If you are allowed to configure unique settings for the policy, you must deselect the option to make your edits.
- If **View** ( ) appears instead, settings are inherited from an ancestor policy, or you do not have permission to modify the settings. In this case, inheritance has been configured so that these settings are locked. The settings are read-only.

Follow these steps to configure advanced settings for the access control policy:

Procedure

Step 1 Choose **Policies** > **Security policies** > **Access Control**.

Step 2 Create or edit an access control policy.

Step 3 Select **Advanced Settings** from the **More** drop-down arrow at the end of the packet flow line.

Step 4 For each settings group, click **Edit** ( ) and configure the settings as needed.

For each feature group, a separate dialog box is opened where you can make your changes. Click **OK** to save any changes.

- **General Settings**—These settings apply broadly to the policy, including URL filtering options. For more information, see General settings, on page 5.
- **Identity Policy Settings**—Select the policy to be used to implement user identity discovery. You must implement an identity policy to get user or user group information in connection events, or to write access control rules based on users or groups. For more information, see Identity policies.
- **Decryption Policy Settings**—Select the policy to be used when decrypting connections. You must decrypt traffic to apply inspection to encrypted connections.
- **TLS Server Identity Discovery**—Whether to allow the firewall to extract certificate details such as Common Name (CN), Organization, or Subject Alternative Names (SANs) even when TLS 1.3 encryption would normally hide them. This improves policy accuracy without requiring a decryption rule; the original client connection remains encrypted. For more information, see TLS server identity discovery, on page 6.
- **Prefilter Policy Settings**—Select the policy to be used for statically offloading large flows, implementing early connection blocks, or rezoning plain-text tunnel traffic.
- **Network Analysis and Intrusion Policies**—Advanced network analysis and intrusion policy settings allow you to:
  - Specify the intrusion policy and associated variable set that are used to inspect packets that must pass before the system can determine exactly how to inspect that traffic.
  - Change the access control policy's default network analysis policy, which governs many preprocessing options.
  - Use custom network analysis rules and network analysis policies to tailor preprocessing options to specific security zones, networks, and VLANs.
- **Threat Defense Service Policy**—Use the service policy to apply services to specific traffic classes. For example, you can use a service policy to create a timeout configuration that is specific to a particular TCP application, as opposed to one that applies to all TCP applications. The service policy rules are applied after the access control rules. For more information, see Service Policies.
- **Files and Malware Settings**—Tune file and malware inspection performance and storage provides information on performance options for file control and malware defense.
- **Threat Detection**—Configure the portscan detector to detect and prevent portscan activity in all types of traffic to protect networks from eventual attacks. Portscan traffic can be detected efficiently in both allowed and denied traffic..
- **Elephant Flow Settings**—Elephant flows are large, long duration, and fast flows that can cause duress for Snort cores. There are two actions that you can apply on elephant flows to reduce system stress, CPU hogging, packet drops, and so on. These actions are:
  - Bypass any or all applications—This action bypasses flow from Snort inspection.
  - Throttle—This action applies dynamic rate limit policy (10% reduction) on elephant flows.

  For more information, see Configure elephant flow detection.
- **Intelligent Application Bypass Settings**—(Use **Elephant Flow Settings** instead of this option.) Intelligent application bypass (IAB) is an expert-level configuration that specifies applications to bypass or test for bypass if traffic exceeds a combination of inspection performance and flow thresholds.

IAB settings are applicable for Snort2 devices or pre 7.2.0 Snort3 devices. For more information, see Intelligent application bypass, on page 7.

- **Transport/Network Layer Preprocessor Settings**—Advanced transport and network preprocessor settings apply globally to all networks, zones, and VLANs where you deploy your access control policy. For more information, see Transport/network layer preprocessor settings, on page 13.
- **Detection Enhancement Settings**—Detection enhancement settings determine whether adaptive profiles are used for application detection and intrusion rules in the access control policy. Typically, the system uses the static settings in your network analysis policy to preprocess and analyze traffic. With adaptive profile updates, the system can adapt processing behavior using host information either detected by network discovery or imported from a third party. For more information, see Detection enhancement settings, on page 14.
- **Performance Settings**—Settings for improving the performance of your system as it analyzes traffic for attempted intrusions. These settings are very advanced and should be left as defaults by most users. See Performance settings, on page 17.
- **Latency-Based Performance Settings**—Settings specific to latency-based performance. These settings are very advanced and should be left as defaults by most users. See Latency-based performance settings, on page 19.
- **Shadow Traffic**—The Shadow traffic dashboard enhances the visibility of traffic originating from unsanctioned privacy technologies. This type of traffic is specifically designed to evade traditional network monitoring and analysis by advanced firewalls. There is also a shadow traffic type attribute added to connection and unified events. You can disable this option if you do not need visibility on traffic that might contain unsanctioned content. For more information, see *The Shadow Traffic Widget* in the *Cisco Secure Firewall Management Center Administration Guide*
- **Advanced Logging**—Enable this feature to enrich connection logs with application data and forward the generated logs to the syslog destinations. Application logging leverages existing deep packet inspection capabilities to extract application data and enables you to enhance network monitoring and gain deeper insights into network traffic. This feature applies to Snort 3 Firewall Threat Defense devices.

  For more information about the application logging, see *.Application-Aware Event Logging* in the *Cisco Secure Firewall Management Center Administration Guide*



Note Application logging can cause performance drop within network if used without filters configured in the access control rule. Filter specific traffic types using the access control rules to reduce the volume of logged traffic.

Step 5 Click **Save** to save the changes to the access control policy.

- To get back to the rules list, click **Access Control** in the packet flow line.
- You must deploy the configuration to apply changes to the managed devices.

The advanced settings for the access control policy are configured and saved. The changes will take effect after deployment to the managed devices.

General settings

General settings are advanced configuration options that

- control URL logging and character limits for connection events,
- manage user bypass timeouts for interactive blocking,
- determine URL cache lookup behavior and traffic inspection settings, and
- enable threat intelligence and DNS reputation enforcement features.

General advanced settings configuration options

These are the general advanced settings that you can configure for an access control policy:

- **Maximum URL characters to store in connection events**—Sets the maximum character length of each URL requested by users in end-of-connection events. Disabling or limiting the number of stored URL characters might improve system performance. The default is 1024. The range is 0 to 4096.

  Set the length to 0 to disable URL logging. Storing zero characters does not affect URL filtering. The system filters traffic based on requested URLs even though the system does not record them.

- **Allow an Interactive Block to bypass blocking for (seconds)**—Sets the time allowed for browsing after the user bypasses a URL filtering block. After the timeout expires, the user must bypass the block again. The default is 600 seconds (10 minutes). The range is 0 to 31536000 (8760 hours).

  Setting this value to 0 means the interactive block response is displayed once and the user bypass never expires.

- **Retry URL cache miss lookup**—This setting determines what the system does when it needs to look up a URL's category and reputation in the cloud.

  The first time the system encounters a URL that does not have a locally stored category and reputation, it looks up that URL in the cloud and adds the result to the local data store. This allows faster processing of that URL in the future.

  By default, this setting is enabled. The system momentarily delays the traffic while it checks the cloud for the URL's reputation and category, and uses the cloud verdict to handle the traffic.

  If you disable this setting, when the system encounters a URL that is not in its local cache, the traffic is immediately passed and handled according to the rules configured for uncategorized and reputationless traffic.

  In passive deployments, the system does not retry the lookup, as it cannot hold packets.

- **Enable reputation enforcement on DNS traffic**—Whether to have the system evaluate domain category and reputation early in URL transactions, when the browser looks up the domain name to get the IP address. Enable this option to improve URL filtering performance and efficacy. The default is enabled. For details and additional instructions, see DNS filtering and subtopics.

- **Inspect traffic during policy apply**—Whether to inspect traffic when you deploy configuration changes unless specific configurations require restarting the Snort process. The default is enabled.

  When this option is enabled, resource demands could result in a small number of packets dropping without inspection. Additionally, deploying some configurations restarts the Snort process, which interrupts traffic inspection. The assigned device determines whether traffic drops during this interruption or passes without further inspection. See Snort restart scenarios for more information.

TLS server identity discovery

TLS server identity discovery is an access control policy advanced setting that enables early application detection and URL categorization for encrypted traffic.

TLS server identity discovery configuration

The latest version of the Transport Layer Security (TLS) protocol 1.3, defined by RFC 8446, is the preferred protocol for many web servers to provide secure communications. Because the TLS 1.3 protocol encrypts the server's certificate for additional security, and the certificate is needed to match application and URL filtering criteria in access control rules, the Firepower System provides a way to extract the server certificate *without* decrypting the entire packet.

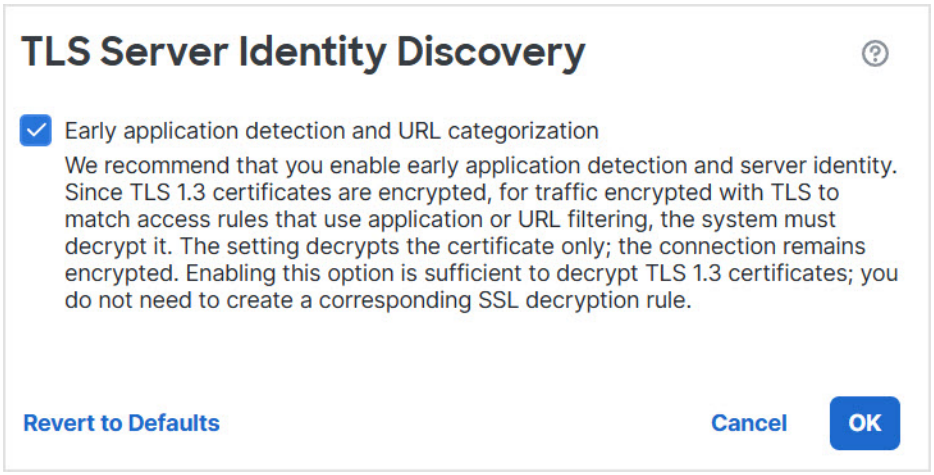
You can enable this feature, referred to as *TLS server identity discovery*, when you configure advanced settings for an access control policy. Certain features are not supported, such as STARTTLS traffic, the HTTP CONNECT method, and in a network where another device is already decrypting traffic.

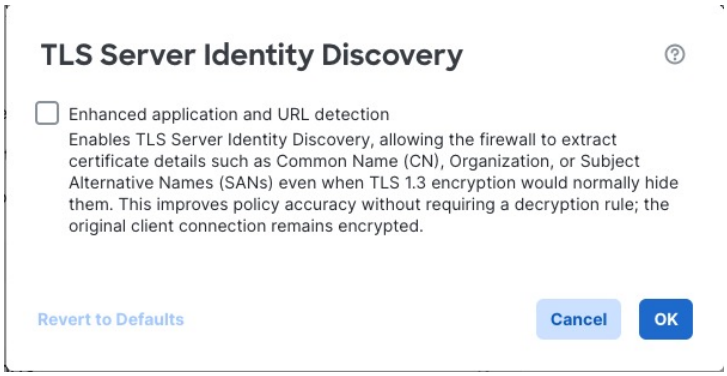
If you enable this option, we recommend you also enable the decryption policy's advanced TLS adaptive server identity probe option as well. Together, these options enable more efficient decryption of TLS 1.3 traffic. For more information, see TLS 1.3 decryption best practices.

When a new connection starts that will be affected by TLS server identity discovery, the Firewall Threat Defense holds the original ClientHello packet to determine the identity of the server to which it connects before continuing. The Firewall Threat Defense device sends a specialized connection from the Firewall Threat Defense to the server. The server's response includes the server certificate, the specialized connection is terminated, and the original connection is evaluated as required by the access control policy.

TLS server identity discovery prioritizes the certificate's Common Name (CN) over the Server Name Indication (SNI).

To enable TLS server identity discovery, click the **Advanced** tab, click **Edit** (✎) for the setting, and select **Early application detection and URL categorization**.





We strongly recommend enabling it for any traffic you want to match on application or URL criteria, especially if you want to perform deep inspection of that traffic. An decryption policy is not required because *traffic is not decrypted* in the process of extracting the server certificate.

Note

- TLS server identity discovery *cannot* be used with any of the following:
  - STARTTLS traffic
  - The HTTP CONNECT method
  - Traffic that is already being decrypted by another device on the network
- Because the certificate is decrypted, TLS server identity discovery can reduce performance depending on the hardware platform.
- TLS server identity discovery is not supported in inline tap mode or passive mode deployments.
- Enabling TLS server identity discovery is supported on any Secure Firewall Threat Defense Virtual version deployed in AWS. TLS server identity is incompatible with certain deployment scenarios that involve Gateway Load Balancer (GWLB). Whether the device initiates a server certificate probe depends on its deployment method with GWLB and the traffic direction.
  - Geneve single-arm flow: The probe *is not initiated* for either east–west or north–south traffic. A connection event **PROBE_FLOW_DROP_BYPASS_PROXY** increments every time the device attempts to extract the server certificate.
  - Geneve dual-arm flow: The probe *is initiated only* for north–south traffic.
- TLS Server Identity Discovery also operates on TLS 1.2 sessions.

Intelligent application bypass

Intelligent application bypass (IAB) is a security feature that

- identifies applications that you trust to traverse your network without further inspection if performance and flow thresholds are exceeded
- implements on traffic allowed by access control rules or the access control policy's default action, before the traffic is subject to deep inspection, and

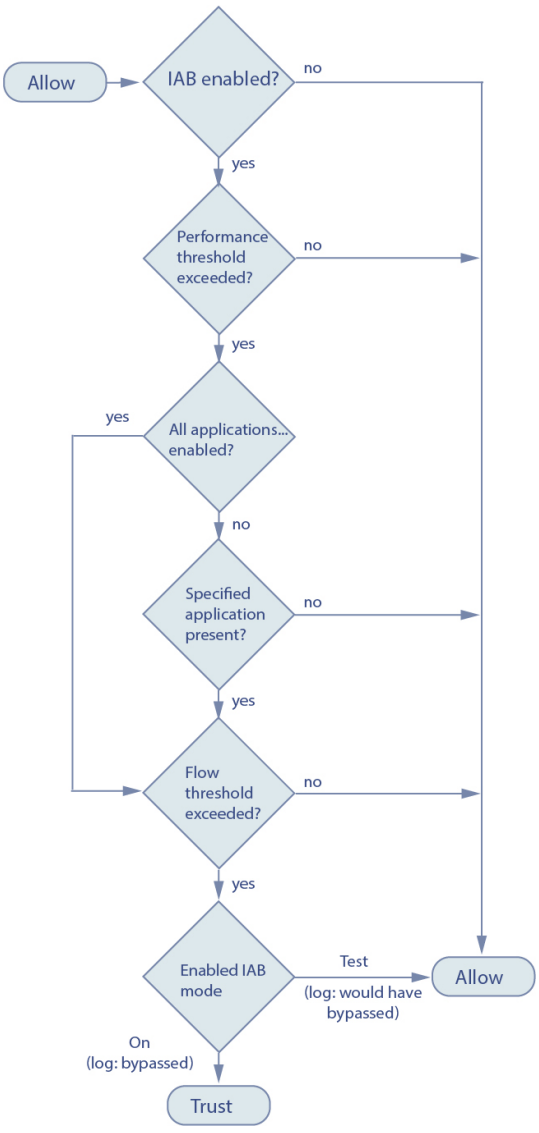
- can be configured so that when an inspection performance threshold is exceeded, IAB trusts all traffic that exceeds any flow bypass threshold, regardless of the application type.

Additional information

For example, if a nightly backup significantly impacts system performance, you can configure thresholds that, if exceeded, trust traffic generated by your backup application.

A test mode allows you to determine whether thresholds are exceeded and, if so, to identify the application flows that would have been bypassed if you had actually enabled IAB (called *bypass mode*).

This graphic illustrates the IAB decision-making process:



Configure intelligent application bypass

This task enables intelligent application bypass (IAB) to improve system performance by bypassing inspection of specified traffic when performance thresholds are exceeded.

Not all deployments require IAB, and those that do might use it in a limited fashion. Do not enable IAB unless you have expert knowledge of your network traffic, especially application traffic, and system performance, including the causes of predictable performance issues. Before you run IAB in bypass mode, make sure that trusting the specified traffic does not expose you to risk.

Before you begin

IAB settings are applicable for Snort2 devices or pre 7.2.0 Snort3 devices. For Snort 3 devices, use elephant flow detection instead.

Follow these steps to configure intelligent application bypass:

Procedure

Step 1 In the access control policy editor, click **Advanced Settings** from the **More** drop-down arrow at the end of the packet flow line. Then, click **Edit** (✎) next to **Intelligent Application Bypass Settings**.

Step 2 Set the enabled **State** for IAB.

Turn IAB **Off** or **On**, or enable IAB in **Test** mode.

In test mode, connection events and dashboards tell you what the system would have done if IAB had been on, but traffic is not impacted. Use test mode to check your configuration.

Step 3 Set the **Performance Sample Interval**.

The **Performance Sample Interval** specifies the time in seconds between IAB performance sampling scans, during which the system collects system performance metrics for comparison to IAB performance thresholds. The default is 5 seconds. The range is 1 to 1000 seconds.

Step 4 Select the **Bypassable Applications and Filters**.

Choose from:

- ***X* Applications/Filters**—Click the link and select the applications or application filters whose traffic you want to bypass. You can select by general attributes, specific applications, or both. For example, you could limit bypassable traffic to very low risk applications only.
- **All applications including unidentified applications**—Do not restrict bypass. When an inspection performance threshold is exceeded, IAB trusts all traffic that exceeds any flow bypass threshold, regardless of the application type. This is the default.

Step 5 Configure the performance and flow thresholds.

You must configure at least one **Inspection Performance Thresholds** and one **Flow Bypass Thresholds**. However, all have defaults and you do not need to change the settings if the defaults are appropriate for your network.

When a performance threshold is exceeded, the system examines flow thresholds and, if one threshold is exceeded, trusts the specified traffic. If you enable more than one of either, only one of each must be exceeded.

a) Click **Configure** under **Inspection Performance Thresholds** and configure the options.

Inspection performance thresholds provide intrusion inspection performance limits that, if exceeded, trigger the inspection of flow thresholds. Inspection performance thresholds set to 0 are ignored.

You can configure one or more of these thresholds:

- **Drop Percentage**—The average packets dropped as a percentage of total packets, when packets are dropped because of performance overloads caused by expensive intrusion rules, file policies, decompression, and so on. This does not refer to packets dropped by normal configurations such as intrusion rules.

  Note that specifying an integer greater than 1 activates IAB when the specified percentage of packets is dropped. When you specify 1, any percentage from 0 through 1 activates IAB. This allows a small number of packets to activate IAB.

  The default is 5%. The range is 0 to 100.

- **Processor Utilization Percentage**—Average percentage of processor resources used. The default is 95. The range is 0 to 100.

- **Packet Latency**—The average packet latency in microseconds. The default is 1000. The range is 0 to 1000000.

- **Flow Rate**—The rate at which the system processes flows, measured as the number of flows per second. Note that this option configures IAB to measure flow *rate*, not flow *count*. The default is 0. The range is 0 to 1000000.

b) Click **Configure** under **Flow Bypass Thresholds** and configure the options.

Flow bypass thresholds provide flow limits that, if exceeded, trigger IAB to trust bypassable application traffic in bypass mode or allow application traffic subject to further inspection in test mode. Flow bypass thresholds set to 0 are ignored.

You can configure one or more of these flow bypass thresholds:

- **Bytes per Flow**—The maximum number of kilobytes a flow can include. The default is 500000. The range is 0 to 2147483647.

- **Packets per Flow**—The maximum number of packets a flow can include. The default is 0. The range is 0 to 2147483647.

- **Flow Duration**—The maximum number of seconds a flow can remain open. The default is 0. The range is 0 to 2147483647.

- **Flow Velocity**—The maximum transfer rate in kilobytes per second. The default is 250000. The range is 0 to 2147483647.

Step 6 Click **OK** to save IAB settings.

Step 7 Click **Save** to save the policy.

Intelligent application bypass is configured for the access control policy with the specified performance and flow thresholds.

What to do next

- Deploy configuration changes.

IAB logging and analysis

IAB logging and analysis is a monitoring system that

- forces an end-of-connection event. This event logs both bypassed flows and flows that would have been bypassed, regardless of whether you have enabled connection logging
- provides connection events. These events indicate when flows are bypassed in bypass mode or would have been bypassed in test mode, and
- enables custom dashboard widgets and reports based on connection events. These widgets and reports display long-term statistics for bypassed and would-have-bypassed flows.

IAB connection events

IAB connection events provide specific information about bypassed and would-have-bypassed flows through these key fields:

- Action: When **Reason** includes `Intelligent App Bypass`:
  - **Allow**—indicates that the applied IAB configuration was in test mode and traffic for the application specified by **Application Protocol** remains available for inspection.
  - **Trust**—indicates that the applied IAB configuration was in bypass mode and traffic for the application specified by **Application Protocol** has been trusted to traverse the network without further inspection.
- Reason: `Intelligent App Bypass` indicates that IAB triggered the event in bypass or test mode.
- Application Protocol: This field displays the application protocol that triggered the event.

You can create custom analysis dashboard widgets to display long-term IAB statistics based on connection events. Specify these settings when creating the widget:

- **Preset**: None
- **Table**: Application Statistics
- **Field**: any
- **Aggregate**: either of:
  - IAB Bypassed Connections
  - IAB Would Bypass Connections
- **Filter**: any

You can create custom reports to display long-term IAB statistics based on connection events. Specify these settings when creating the report:

- **Table**: Application Statistics
- **Preset**: None
- **Filter**: any
- **X-Axis**: any

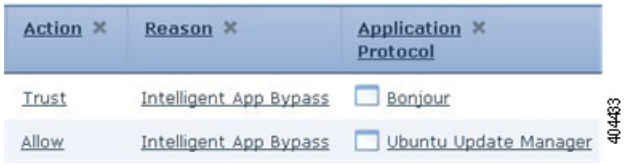
- **Y-Axis**: either of:
  - IAB Bypassed Connections
  - IAB Would Bypass Connections

Connection event examples

In the truncated graphic, some fields are omitted. The graphic shows the **Action**, **Reason**, and **Application Protocol** fields for two connection events resulting from different IAB settings in two separate access control policies.

For the first event, the `Trust` action indicates that IAB was enabled in bypass mode and Bonjour protocol traffic was trusted to pass without further inspection.

For the second event, the `Allow` action indicates that IAB was enabled in test mode, so Ubuntu Update Manager traffic was subject to further inspection but would have been bypassed if IAB had been in bypass mode.

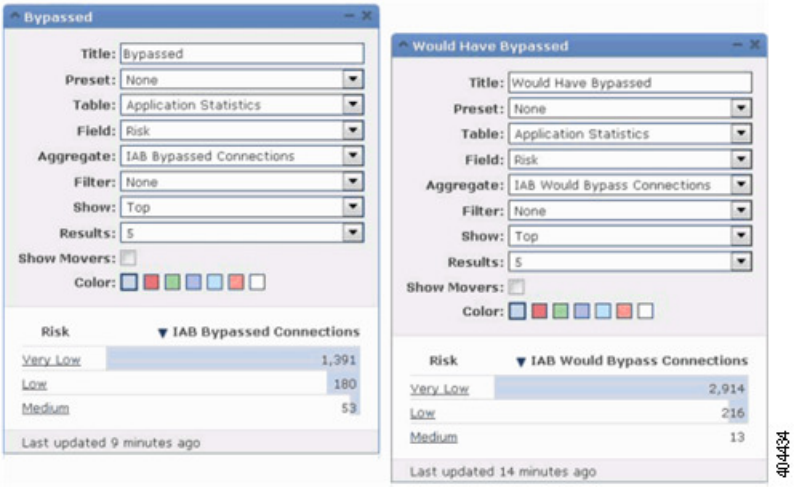


In the truncated graphic, some fields are omitted. The flow in the second event was both bypassed (**Action**: `Trust`; **Reason**: `Intelligent App Bypass`) and inspected by an intrusion rule (**Reason**: `Intrusion Monitor`). The `Intrusion Monitor` reason indicates that an intrusion rule set to **Generate Events** detected but did not block an exploit during the connection. In the example, this happened before the application was detected. After the application was detected, IAB recognized the application as bypassable and trusted the flow.



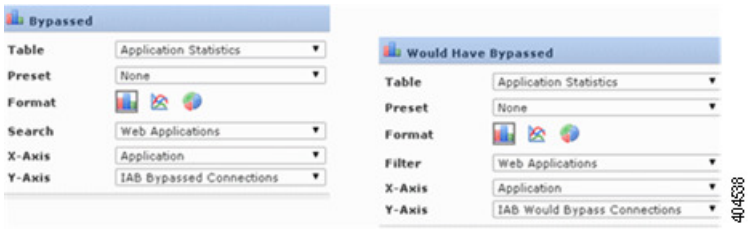
In the custom analysis dashboard widget examples:

- The *Bypassed* example shows statistics for application traffic bypassed because the applications were specified as bypassable and IAB was enabled in bypass mode in the deployed access control policy.
- The *Would Have Bypassed* example shows statistics for application traffic that would have been bypassed because the applications were specified as bypassable and IAB was enabled in test mode in the deployed access control policy.



The graphic shows two abbreviated report examples:

- The *Bypassed* example shows statistics for application traffic bypassed because the applications were specified as bypassable and IAB was enabled in bypass mode in the deployed access control policy.
- The *Would Have Bypassed* example shows statistics for application traffic that would have been bypassed because the applications were specified as bypassable and IAB was enabled in test mode in the deployed access control policy.



Transport/network layer preprocessor settings

Transport/network layer preprocessor settings are access control policy configurations that

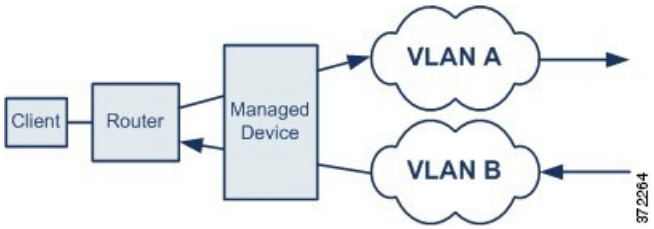
- apply globally to all networks, zones, and VLANs where you deploy your access control policy
- control how the system handles VLAN headers, active responses, and connection tracking, and
- affect traffic reassembly and rule processing for transport and network layer communications.

Configuration options

These settings control various aspects of transport and network layer processing:

- **Ignore the VLAN header when tracking connections**—Whether to ignore or include VLAN headers when identifying traffic. Different VLAN tags in traffic traveling in different directions for the same connection can affect traffic reassembly and rule processing. For example, traffic for the same connection

could be transmitted over VLAN A and be received over VLAN B. Select this option if the device might see different VLANs for the same connection. This option is off by default.



- **Maximum Active Responses**—For a TCP connection that triggers a preprocessor/intrusion drop rule that is configured to provide an active response, the maximum number of active responses per TCP connection. When additional traffic occurs on a connection where an active response has been initiated, and the traffic occurs more than **Minimum Response Seconds** after a previous active response, the system sends another active response unless the specified maximum has been reached. A setting of 0 disables additional active responses triggered active response rules. The default is no limit. The range is 0 to 25.

Note You have to specifically configure drop rules to provide active responses. For TCP connections, the active response is a RESET packet. For UDP connections, the system sends an ICMP unreachable packet to the source of the connection.

- **Minimum Response Seconds**—Until **Maximum Active Responses** occur, specifies the number of seconds to wait before any additional traffic on a connection where the system has initiated an active response results in a subsequent active response. The default is no limit. The range is 1 to 300.
- **Session Termination Logging Threshold**—Do not modify this option unless instructed to do so by Cisco Technical Support.

  Support might ask you during a troubleshooting call to configure your system to log a message when an individual connection exceeds the specified threshold. Changing the setting for this option will affect performance and should be done only with Support guidance. This option specifies the number of bytes that result in a logged message when the session terminates and the specified number was exceeded. The upper limit is 1GB.

Detection enhancement settings

Detection enhancement settings are adaptive profile configurations that

- determine whether adaptive profiles are used for application detection and intrusion rules in the access control policy
- enable the system to adapt processing behavior using host information either detected by network discovery or imported from a third party, and
- provide control over profile update frequency and network scope for performance optimization.

Detection enhancement setting options

Typically, the system uses the static settings in your network analysis policy to preprocess and analyze traffic. With adaptive profile updates enabled, the system adapts processing behavior using host information detected by network discovery or imported from a third party.

Note To enable adaptive profiles in Snort 3, you must select both the **Enable** and **Enable Profile Updates** options.

The detection enhancement settings include these options:

- **Enable**—You must enable adaptive profiling (its default state) for access control rules to perform application and file control, including malware protection (AMP), and for intrusion rules to use service metadata.
- **Enable Profile Updates**—Profile updates, like the target-based profiles you can configure manually in a network analysis policy, help to defragment IP packets and reassemble streams in the same way as the operating system on the target host. The intrusion rules engine then analyzes the data in the same format as that used by the destination host. Profile updates also compare metadata in an intrusion rule to host information to determine whether a rule should apply for a particular host. For more information, see:
  - Adaptive profile updates, on page 15
  - Adaptive profile updates and recommended rules, on page 16
- **Adaptive Profiles – Attribute Update Interval**—When profile updates are enabled, you can control how frequently in minutes network map data is synced from the management center to its managed devices. The system uses the data to determine what profiles should be used when processing traffic. Increasing the value for this option can improve performance in a large network.
- **Adaptive Profiles – Networks**—Optionally, when profile updates are enabled, you can improve performance by constraining profile updates to a comma-separated list of IP addresses, address blocks, and network variables. If you use a network variable, the system uses the variable's value in the variable set linked to the default intrusion policy for your access control policy. For example, you could enter 192.168.1.101, 192.168.4.0/24, or $HOME_NET. IPv4 and IPv6 are supported.

  The default value (0.0.0.0/0) applies adaptive profile updates to all networks.

Adaptive profile updates

An adaptive profile update is a network analysis feature that

- adapts processing behavior using host information either detected by network discovery or imported from a third party
- switches to the appropriate operating system profile based on the operating system in the host profile for the target host, and
- helps to defragment IP packets and reassemble streams in the same way as the operating system on the target host.

Adaptive profile operation

Typically, the system uses the static settings in your network analysis policy to preprocess and analyze traffic. With adaptive profile updates, the system can adapt processing behavior using host information either detected by network discovery or imported from a third party.

Profile updates, like the target-based profiles you can configure manually in a network analysis policy, help to defragment IP packets and reassemble streams in the same way as the operating system on the target host. The intrusion rules engine then analyzes the data in the same format as that used by the destination host.

Manually configured target-based profiles apply either the default operating system profile you select, or profiles you bind to specific hosts. Profile updates, however, switch to the appropriate operating system profile based on the operating system in the host profile for the target host.

Adaptive profile update scenario

Consider a scenario where you configure profile updates for the 10.6.0.0/16 subnet and set the default IP Defragmentation target-based policy to Linux. The Cloud-Delivered Firewall Management Center where you configure the settings has a network map that includes the 10.6.0.0/16 subnet.

- When the system detects traffic from Host A, which is not in the 10.6.0.0/16 subnet, it uses the Linux target-based policy to reassemble IP fragments.
- When the system detects traffic from Host B, which is in the 10.6.0.0/16 subnet, it retrieves Host B's operating system data from the network map. The system uses a profile based on that operating system to defragment the traffic destined for Host B.

Adaptive profile updates and recommended rules

Adaptive profile updates and recommended rules are security features that

- compare metadata in a rule to host information to determine whether a rule should apply for a particular host
- provide different approaches to rule optimization, with adaptive profiles applying rules dynamically on a packet-by-packet basis while recommended rules provide suggestions for rule state changes, and
- can be used in combination to ensure appropriate rule enablement and efficient rule application for specific traffic.

Feature characteristics and differences

The adaptive profile updates feature is an advanced setting in an access control policy that applies globally to all intrusion policies invoked by that access control policy. The Cisco recommended rules feature applies to the individual intrusion policy where you configure it.

Like recommended rules, profile updates compare metadata in a rule to host information to determine whether a rule should apply for a particular host. However, while recommended rules provide recommendations for enabling or disabling rules using that information, profile updates use the information to apply specific rules to specific traffic.

This table compares the key differences between adaptive profile updates and recommended rules:

Recommended rules require your interaction to implement suggested changes to rule states. Profile updates, on the other hand, do not modify intrusion policies. Treatment of rules based on profile updates happens on a packet-by-packet basis.

Additionally, recommended rules can result in enabling disabled rules. Profile updates, in contrast, only affect the application of rules that are already enabled in intrusion policies. Profile updates never change the rule state.

You can use profile updates and recommended rules in combination. Profile updates use the rule state for a rule when your intrusion policy is deployed to determine whether to include it as a candidate for applying, and your choices to accept or decline recommendations are reflected in that rule state. You can use both features to ensure that you have enabled or disabled the most appropriate rules for each network you monitor, and then to apply enabled rules most efficiently for specific traffic.

Performance settings

Performance settings are access control policy advanced settings that

- tune the performance of your system as it analyzes traffic for attempted intrusions
- apply to Snort and are relevant only if you apply intrusion policies in rules or as the default action, and
- should be changed only if you are a Snort intrusion rule expert, or at the direction of Cisco Technical Support.

Performance setting categories

These settings are organized into separate tabs for each group of settings:

- **Pattern Matching Limits**: The maximum number of events to queue and TCP payload detection options
- **Performance Statistics**: Time range and packet count settings for performance sampling
- **Regular Expression Limits**: PCRE matching and recursion limits
- **Intrusion Event Logging Limits**: Event queue and logging configuration

Pattern Matching Limits include these options:

- **Maximum Pattern States to Analyze Per Packet**: The maximum number of events to queue. The default is 5.
- **Disable Content Checks on Traffic Subject to Future Reassembly**: Whether to detect TCP payload before reassembly. It includes inspection of packets before and after stream reassembly. This process requires more processing overhead and may decrease performance. If the option is not selected, the TCP payload is detected after reassembly. The default is off.

Performance Statistics settings control when the system updates performance statistics. When the number of seconds specified elapses between performance statistics updates, the system verifies it has analyzed the specified number of packets. If it has, the system updates performance statistics. Otherwise, the system waits until it analyzes the specified number of packets.

- **Sample time (seconds)**: The time range for taking a performance sample. The default is 300 seconds.

Configuring a very low value (for example, 1 second) for the sample time can cause a huge impact on the device; the performance statistics logged on the device can cause disk space issues and affect the operation of the device. Hence, we recommend you do not configure a very low value.

- **Minimum number of packets**: How many packets to consider the minimum for a valid performance statistic. The default is 0.
- **Troubleshooting Options**:
  - **Log Session/Protocol Distribution**: Support might ask you during a troubleshooting call to enable this option to configure the system to calculate the performance statistics only when the Snort process is shut down or restarted.
  - **Summary**: Enable this option only if instructed by Cisco Technical Support.

Regular Expression Limits ensure a minimum level of performance. The default Perl compatible regular expression (PCRE) limits ensure a minimum level of performance. Overriding these limits could increase security but could also significantly impact performance by permitting packet evaluation against inefficient regular expressions.

- **Match Limit State**: The limit for matching regular expressions. You can select **Default Value**, which is 3500, **Unlimited**, or **Custom**. If you select custom, specify the number of times to attempt to match a pattern defined in a PCRE regular expression in **Match Limit**. Specify 0 to completely disable PCRE match evaluations.
- **Match Recursion Limit State**: The limit for matching regular expression recursions. You can select **Default Value**, which is 3500, **Unlimited**, or **Custom**. If you select custom, specify the number of recursions when evaluating a PCRE regular expression against the packet payload in **Match Recursion Limit**. Specify 0 to completely disable PCRE recursions.



Note For a custom match recursion limit to be meaningful, it must be smaller than the match limit.

Intrusion Event Logging Limits control event queuing and reporting. When the intrusion rules engine evaluates traffic against rules, it places the events generated for a given packet or packet stream in an event queue, then reports the top events in the queue to the user interface. When configuring the intrusion event logging limits, you can specify how many events can be placed in the queue and how many are logged, and select the criteria for determining event order within the queue.

- **Maximum Events Stored Per Packet**: The maximum number of events that can be stored for a given packet or packet stream. The default is 8.
- **Maximum Events Logged Per Packet**: The number of events logged for a given packet or packet stream. This cannot exceed the **Maximum Events Stored Per Packet** value. The default is 5.
- **Prioritize Events Logging By**: The value used to determine event ordering within the event queue. The highest ordered event is reported through the user interface. You can select:
  - **Content Length** (the default), which orders events by the longest identified content match. When events are ordered by content length, rule events always take precedence over decoder and preprocessor events.
  - **Priority**, which orders events in the queue by the event priority.

Latency-based performance settings

Latency-based performance settings are policy configurations that

- use thresholding to manage packet and rule processing performance,
- apply to Snort and are relevant only if you apply intrusion policies in rules or as the default action, and
- are automatically populated by the latest deployed intrusion rule update by default.

Performance setting behavior

By default, latency-based performance settings for both packet and rule handling are automatically populated by the latest deployed intrusion rule update, and we recommend that you do not change the default. Change these settings only if you are a Snort intrusion rule expert, or at the direction of Cisco Technical Support.

The latency settings that are applied depend on the security level of the network analysis policy (NAP) associated with the access control policy. Generally, this is the default NAP policy. However, if custom network analysis rules are configured, and if any of these specify a NAP policy that is more secure than the default NAP policy, then latency settings are based on the most secure NAP policy among the custom rules. If the default NAP policy or any custom rules invoke a custom NAP policy, then the security level used in the evaluation is the system-provided base policy on which each custom NAP policy is based.

The above is true regardless of whether the effective threshold and/or network analysis configurations are inherited or configured directly in the policy.

Use these settings to tune the latency-based performance of your system:

- **Apply Settings From**: Whether to apply latency-based performance settings from an **Installed Rule Update**, the default, or from your **Custom** settings.
- **Packet Handling**: Packet latency thresholding measures the total elapsed time taken to process a packet by applicable decoders, preprocessors, and rules, and ceases inspection of the packet if the processing time exceeds a configurable threshold. By default, the latency-based performance setting for packet handling is disabled. You may choose to enable it. However, Cisco recommends that you do not change the default value for the threshold setting. Select **Enabled** to turn it on. If you selected **Custom**, also enter the **Threshold** time in microseconds for when inspection of a packet should cease. The default is 256.
- **Rule Handling**: Rule latency thresholding measures the elapsed time each rule takes to process an individual packet, suspends the violating rule along with a group of related rules for a specified time if the processing time exceeds the rule latency threshold a configurable consecutive number of times, and restores the rules when the suspension expires. You can configure these options only if you select **Custom**.
  - **Enabled**: This option is selected automatically if you select custom. If you do not want use this feature, deselect the option. All other settings on this tab require the feature to be enabled.
  - **Threshold (microseconds)**: Specifies the time in microseconds that rules should not exceed when examining a packet. The default is 512.
  - **Consecutive Threshold Violations Before Suspending Rule**: Specifies the consecutive number of times rules can take longer than the time set for threshold to inspect packets before rules are suspended. The default is 3.
  - **Suspension Time**: How long a suspended rule should be suspended. The default is 10.