Application Detection and Control Overview

This chapter provides an overview of the Application Detection and Control (ADC) in-line service, formerly known as Peer-to-Peer Detection.

The System Administration Guide provides basic system configuration information, and the product administration guides provide procedures to configure basic functionality of core network service. It is recommended that you select the configuration example that best meets your service model, and configure the required elements for that model, as described in the respective product Administration Guide, before using the procedures in this chapter.

This chapter covers the following topics:

ADC Overview

The ADC in-line service is mainly used to detect Peer-to-Peer protocols by analyzing traffic. Other popular applications that generate the bulk of Internet traffic like Social Networking and Gaming applications can be detected.

The ADC in-line service works in conjunction with the following products:

- GGSN
- PDSN
- P-GW

The in-line service now known as ADC is continued to be referred as "P2P" in the configuration.

Peer to Peer (P2P) is a term used in two slightly different contexts. At a functional level, it means protocols that interact in a peering manner, in contrast to client-server manner. There is no clear differentiation between the function of one node or another. Any node can function as a client, a server, or both — a protocol may not clearly differentiate between the two. For example, peering exchanges may simultaneously include client and server functionality, sending and receiving information. P2P is a type of transient Internet network that allows a group of computer users with the same networking program to connect with each other and directly access files from one another's hard drives. A common use case of a P2P application is file sharing.

Once the P2P Client is downloaded and installed, it will log on to a central indexing server. This central server indexes all users who are currently online connected to the server. This server does not host any files for downloading. The P2P client can search for a specific file. The utility queries the index server to find other connected users with the specific file. When a match is found, the central server directs to find the requested

file. The result is chosen from the search query and the utility will then attempt to establish a connection with the computer hosting the requested file. If a successful connection is made, it will begin downloading the file. Once the file download is complete, the connection will be broken.

The stunning growth and intensive bandwidth nature of P2P applications can have a significant impact on the underlying network. As most deployments are designed with a significant bias towards downstream traffic, P2P applications stress uplink capacity resulting in increased latency, decreased responsiveness and packet loss.

To avoid detection, P2P software undergoes frequent changes and this requires service providers to upgrade the software with the latest P2P detection logic. This upgrade is time consuming, also causing disruption in services and revenue loss. The Dynamic Software Upgrade (DSU) addresses these problems by enabling operators to upgrade their detection capabilities with no downtime. The detection logic is separated out from the main code and shipped as a plugin. Whenever there is a need for software upgrade, the new plugin will be shipped and loaded into the system. For more information, refer to the *Dynamic Software Upgrade* section.

Qualified Platforms

ADC is a StarOS in-line service application that runs on Cisco ASR 5500 platforms. For additional platform information, refer to the appropriate System Administration Guide and/or contact your Cisco account representative.

License Requirements

The ADC is a licensed Cisco feature. A separate feature license may be required. Contact your Cisco account representative for detailed information on specific licensing requirements. For information on installing and verifying licenses, refer to the *Managing License Keys* section of the *Software Management Operations* chapter in the *System Administration Guide*.

Dynamic Software Upgrade

This section describes the Dynamic Software Upgrade (DSU) process that can be used to dynamically update plugins without having to update StarOS and reload the system.

Overview

Dynamic Software Upgrade is the new approach to upgrade the P2P library version that will enable operators to upgrade their detection capabilities with no downtime. This is done by providing updates in the form of software patches which the operator can apply in a live setup with minimal interference.

In this approach, P2P detection code is now delivered as a plugin within the StarOS binary. The plugin is loaded into the system at run time. Whenever there is a change in P2P detection logic of an existing application or a new P2P protocol/application needs to be added, a new version of the plugin is provided as a plugin module. The new plugin is loaded onto the system dynamically without disrupting other services. Once the plugin has been installed and configured, the new P2P rules come into effect for detection.

Important The dynamically loaded plugins are not incremental. A plugin loads protocol detection logic for all the protocols/applications. A user can update to a higher priority plugin or rollback to a lower priority plugin.

Patching is the process used to install a plugin as an update to a StarOS release. One patch can be provided to multiple compatible, concurrent product releases. A plugin patch is distributed in the form of a compressed distribution kit through the internet or by other means (USB, CD, etc.).

A plugin is a functional software entity that provides updates to a pre-existing StarOS software component. Plugins can be dynamically loaded at runtime and do not require a system restart.

A plugin module is a specific instance of a plugin version consisting of at least one file that can be added to a running, in-service system. The module contains the information or instructions for a specific component's update. Typically this will be a single plugin file.

The Version Priority List (VPL) is a linked list of module versions associated with a specific plugin. Each plugin has one VPL. The list is sorted in ascending order by the priority number that is assigned by the administrator. When updating, the lowest priority number is loaded first and if that version is not successful, the version in the VPL with the next sequentially greater priority number is loaded. This list is iterated until a successful version is found. The VPL also supports manual rollback to a previous version (higher priority number).

The basic sequence for the dynamic software upgrade process is as follows:

- Downloading the Patch Kit
- Unpacking the Patch Kit
- Configuring the Plugin
- Loading the Plugin
- Rolling Back to a Previous Plugin Version

For the detailed procedure on performing dynamic software upgrade, refer to the *Configuring Dynamic Software Upgrade* section of the *Application Detection and Control Configuration* chapter.

Note For information on the applications and protocols currently supported by the Application Detection and Control in-line service, contact your Cisco Account representative.

License Requirements

From Release 21.6 onwards, DSU is a licensed-controlled feature. A separate feature license may be required. Contact your Cisco account representative for detailed information on specific licensing requirements. For information on installing and verifying licenses, refer to the *Managing License Keys* section of the *Software Management Operations* chapter in the *System Administration Guide*.

Limitations for DSU

This section lists the limitations of Dynamic Software Upgrade.

- Support for session recovery is limited and there is no support for ICSR in this release.
- The system will allow loading two plugins at the most at any point of time. If there is a need to upgrade the system again, the oldest plugin will be unloaded.
- Detection state for a few subscribers may be lost if a plugin is unloaded from the memory.
- The newly upgraded plugin will be used for all new calls. The existing calls will continue to use the previous plugin.

ADC Protocol Group Detection

Application Detection and Control (ADC) performs traffic analysis and classifies flows into applications and its traffic type. To provide a high-level classification, the protocol grouping feature is implemented to support various application/protocol groups like gaming, file-sharing, e-mail, communicator, etc. Protocol Grouping is done based on the functionality provided by the application. For example, applications like Skype and Yahoo are used for VoIP, so these applications are grouped as Communicator group. The feature is implemented based on the Dynamic Software Upgrade plugin philosophy.

For configuration-related information, refer to the *Configuring P2P Protocol Groups* section of the *Application Detection and Control Configuration* chapter.

Behavioral Traffic

Behavioral Traffic Analysis is a method to analyze network traffic such that all the traffic is analyzed by the generic behavior of each flow. ADC supports behavioral traffic analysis for P2P (Peer-to-Peer), Video, VoIP (Voice over IP), Upload and Download. If the generic behavior of protocols is detected and traffic classified correctly using behavioral analysis, lesser amount of unknown traffic flows can be seen. These behavioral detections must not be used for charging purposes. This feature is based on the Dynamic Software Upgrade plugin philosophy.

Behavioral P2P and behavioral VoIP are meant for zero day detection of P2P/file sharing protocols and VoIP traffic respectively. Behavioral Video is meant for support of video detection. Behavioral Upload/Download must detect flows of non-standard ports that cannot be detected by ECS. This is similar to client-server upload/download using HTTP/FTP/SFTP encrypted download.

Important This feature is disabled by default and meant only for statistical purposes (not for charging purposes).

For configuration-related information, refer to the *Configuring Behavioral P2P and VoIP* section of the *Application Detection and Control Configuration* chapter.

SNI Detection

Server Name Indication (SNI) is an extension of the Transport Layer Security (TLS) protocol that provides a mechanism for the client to tell the server which hostname it is trying to connect to.

ADC detects encrypted traffic using the SNI field (signatures) of TLS/SSL (Secure Sockets Layer) traffic. Due to increased number of applications moving towards TLS/SSL, an option is provided to configure the SNI in ruledef and classify traffic based on the configured SNI with this release.

For detailed overview and configuration information, see the *Support for SNI Detection* chapter in this guide.

SSL Renegotiation Tracking

SSL Renegotiation flows can be detected by tracking the SSL Session ID and its associated protocol. This feature is disabled by default. CLI support is added to enable or disable this feature. The maximum entries of SSL Session ID tracked per Session Manager and the reduce factor can be configured.

Certain applications like Facebook, Gmail, Yahoo, Skype, Twitter, iCloud, etc. widely use the SSL Session Renegotiation feature in their mobile applications to reduce the computational intensive operations involved in a complete SSL negotiation.

Limitations: In certain cases, the SSL Renegotiation detection logic does not work if the SSL sessions involved in the negotiation is spread across more than one subscriber session.

For configuration-related information, refer to the *Configuring SSL Renegotiation* section of the *Application Detection and Control Configuration* chapter.

Analyzer Interworking

Analyzer interworking feature is implemented to analyze the various analyzers simultaneously if the flow is detected as P2P and based on ruledef priority, appropriate charging action will be taken. Currently supported analyzers are FTP, HTTP, RTSP and SIP. CLI support is added to enable or disable this feature. This feature is enabled by default if P2P detection/protocol is enabled.

For configuration-related information, refer to the *Configuring Analyzers* section of the *Application Detection and Control Configuration* chapter.

Traffic Sub-classification

ADC has the capability to detect network traffic for sub-classification of audio, file transfer, instant messaging, video, voipout or unclassified traffic. The duration of the call is a direct indication to the revenue impact of the network operator. The ADC in-line service is well poised to process the network traffic online to detect and control the presence of different network traffic, and generate records that can be used to calculate the traffic call duration.

ADC Support for TRM/FP

P2P flows now support the Transactional Rule Matching (TRM) feature. The TRM/FP feature enables the Enhanced Charging Service (ECS) to bypass per-packet rule matching on a transaction once the transaction is fully classified. This enables ECS to better utilize CPU resources and accommodate additional throughput for the system, thus improving the overall performance.

A transaction for TRM can be defined as the entire UDP flow, the ACK of the 3-way handshake to the FIN/RST of a TCP flow, or the HTTP request to the next HTTP request, or HTTP request to the FIN/RST for the final request of the flow. The TRM feature can also perform rule matching on IP L4 rules (UDP, TCP), HTTP, and HTTPS.

For more information on the TRM/FP feature, refer to the *ECS Administration Guide*.

ADC Support for FAPA

The Flow Aware Packet Acceleration (FAPA) feature improves the throughput in terms of PPS, by caching rule matching results of a flow for selected flows so as not to incur the lookup penalty for a large number of packets in that flow. This new accelerated path is capable of performing a full range of basic functions including handling charging, modification of packet headers, and incrementing various counters. The accelerated path dynamically evaluates the current flow state and reverts back to the slow path when the flow cannot be handled on the fast path.

TRM/FP support has been extended beyond rule-matching. The FAPA function identifies packets that need only a small amount of processing, and performs necessary tasks on these packets. Only those packets that do not require DPI are allowed to enter the Accelerated path. VoLTE, encrypted, HTTP, HTTPS, RTP and plain TCP/UDP traffic where L7 analysis is not enabled, and so on are all the flows that will get accelerated.

Important A Flow Aware Packet Acceleration license is required on ASR5500 and VPC platforms.

P2P flows will be optimized and accelerated using FAPA. ADC when enabled with FAPA improves P2P performance considerably.

Important FAPA accelerates P2P flows for most protocols except for some protocols/applications explicitly listed below.

FAPA accelerates some P2P flows for the following protocols and not all:

- Ares
- Bittorrent
- Didi
- DirectConnect
- Edonkey
- Iskoot
- PPlive
- Scydo
- Soulseek
- Thunder
- ThunderHS
- Tunnelvoice
- Viber
- Whatsapp
- Winny
- Zattoo

The following protocols do not support FAPA:

- ActionVoip
- BBM
- Blackdialer
- Facetime
- Gtalk
- Jabber
- Jumblo
- Kakaotalk
- Magicjack
- MyPeople
- Nateontalk
- Oscar
- Paltalk
- Plingm
- Skype
- Smartvoip
- Tango

- Voipdiscount
- Vopium
- Vtok

For more information on the FAPA feature, refer to the *ECS Administration Guide*.

ADC Support over Gx

The ADC Rule feature will support detection of application level flows as described in Release-11 of 3GPP standard. ADC Rules are certain extensions to dynamic and predefined PCC Rules in order to support specification, detection and reporting of an application flow. These rules are installed (modified/removed) by PCRF via CCA-I/CCA-U/RAR events. ADC rules can be either dynamic PCC or predefined PCC rules, and the existing attributes of dynamic and predefined rules will be applicable.

Important ADC Rule support is a licensed-controlled feature. Contact your Cisco account representative for detailed information on specific licensing requirements.

When the license is not enabled, P2P continues to function as per its original behavior, that is, it monitors the traffic at the entire rulebase level. When the license is enabled, the P2P behavior changes such as to monitor traffic at per subscriber level.

In 19.3 and later releases, this feature is extended to support non-ADC based rules (ECS protocols) in addition to existing P2P protocols, and also detection of application flows for Group of Ruledefs. The following enhancements are supported:

- ADC rules support combination of P2P and non-P2P rule lines in the same ruledef.
- Detection of application flows based on group of ruledefs.
- Application START/STOP event reporting at instance level, that is, per flow basis. This was supported per Application ID basis in previous releases.
- Support of dynamic routes to analyzers for installed ADC rules. Dynamic routes will be supported only for these protocols - HTTP, HTTPS, FTP, RTP, RTCP and SIP.
- Support multi-line AND logic for rulelines when configuring ADC ruledefs.
- Removal of all PCC rules will result in termination of Application Detection for that application. In previous releases, if more than one PCC rule with same Application ID is installed, then removal of any of the PCC rules will terminate Application Detection for that application.

Dynamic PCC rule contains either traffic flow filters or Application ID. When Application ID is present, the rule is treated as ADC Rule. Application ID is the name of the ruledef which is pre-defined in the ASR 5500. This ruledef contains application filters that define the application supported by P2P protocols and non-P2P protocols.

In releases prior to release 19.3: PCEF will process and install ADC rules that are received from PCRF interface, and will detect the specified application(s) and report detection of application traffic to the PCRF. Reporting of application traffic are controlled by PCRF and generates Application Start/Stop events along with the Application ID. Application mute status can be enabled or disabled on both dynamic and predefined ADC rules. When mute is disabled, Application Start/Stop event trigger will be generated by PCEF for that specific Application ID. Mute status can be enabled or disabled by PCRF for dynamic rules, and configured on ASR 5500 for pre-defined rules.

In 19.3 and later releases: When a subscriber attaches to the network, PCRF will install ADC rule/Group of Ruledefs towards PCEF to detect Application flow. The Install ADC rules will additionally enable default routes to HTTP, HTTPS, FTP, RTSP, RTCP or SIP analyzer based on the rule-definition. The default routes use the standard ports associated with the respective protocol. When a new flow comes, the route matching happens for dynamic routes first, then static routes and finally default routes. When a flow matches that ADC rule, an APP-START notification is sent to PCRF with Application ID, Instance ID and flow information. Instance ID is a unique identifier for a particular ADC flow. PCRF then takes necessary action for the detected application. When ADC flow terminates, an APP-STOP notification is sent to PCRF with Application ID and Instance ID.

The following types of ADC ruledefs can be configured.

- **ruledef adc_rule**
  p2p protocol = <name>
  p2p protocol-group = <name>
  p2p behavioral = <name>
  multiline-or all-lines
  end

  multiline-or all-lines is optional if rule contains only one line.

- **ruledef adc_rule_type2**
  p2p protocol = <name>
  p2p traffic-type = <sub_type_name>
  end

- **ruledef adc_rule_type3**
  p2p anymatch = TRUE
  end

  When **p2p any-match** = **TRUE** is configured, only one rule containing this rule line can be installed. This rule line must not be used with any other P2P rule line.

ADC Mute Customization

Earlier, 3GPP ADC over Gx did not support application MUTE status change. Once the application was muted, it was not possible to unmute it. From release 21.1, this feature introduces custom MUTE/UNMUTE functionality. ASR 5500 PCEF now supports customization to control reporting of the Application Detection Information CCRUs. For this, an AVP has been introduced with two possible values - custom MUTE and custom UNMUTE.

- A Gx message might contain both Standards based MUTE and the custom MUTE.
- Standards based MUTE is given preference over the custom MUTE/UNMUTE.
- A dynamic ADC rule can be installed and modified with a custom MUTE.
- Custom-Mute-Notification AVP can be sent by the PCRF in CCA-I and RAR.
- A dynamic ADC rule can be modified with a custom UNMUTE.
- On a custom MUTE for a given dynamic ADC rule, PCEF sends a single APPLICATION_START/ APPLICATION_STOP response for the entire application traffic rather the per flow APPLICATION_START /APPLICATION_STOP response.
- On a custom MUTE for a given dynamic ADC rule, if no APPLICATION_START has been sent prior to the custom MUTE then a single APPLICATION_START is sent on the next flow packet that hits the dynamic rule.

- On a custom MUTE for a given dynamic rule, the APPLICATION_START response is sent with the flow's 5-tuple information.
- On a custom MUTE for a given dynamic rule, the APPLICATION_START response is sent with TDF-Application-Instance-Identifier = 0.
- On a custom MUTE for a given dynamic rule, a single APPLICATION_STOP is sent when the last flow associated with the given dynamic rule is terminated. Such an APPLICATION_STOP will not contain 5-tuple information of the last flow and is sent with TDF-Application-Instance-Identifier = 0.
- On a custom UNMUTE for a given dynamic rule, APPLICATION STARTs response is matched with the given dynamic rule and then sent to all the forthcoming flows.
- There is no change in behavior for a custom UNMUTE, which has not been custom MUTED or standard MUTED before UNMUTING. APPLICATION_STARTs and APPLICATION_STOPs is continued to be sent per flow as before.
- On a custom UNMUTE, PCEF sends an APPLICATION_STOP each for all flows that terminate then onwards.
- A given dynamic rule is recovered in both SR and ICSR including the Custom MUTE/UNMUTE status. The APPLICATION_START status for a given dynamic rule is check-pointed and recovered. This ensures that an extra APPLICATION_START is not sent to the PCRF post recoveries.

TOS/DSCP Support

In 19.3 and later releases, the ADC functionality is extended to identify applications and distinguish bearer traffic based on TOS/DSCP. DSCP/TOS based ADC dynamic rules over Gx will be supported for default and dedicated bearers. Bearer mapping and rule matching will be done based on DSCP/TOS value. Filters can be created for PCC rules based on TOS-Traffic-Class AVP under flow information.

When a subscriber attaches to the network, PCRF will install PCC rule with TOS/DSCP filter towards PCEF. PCEF will create a dedicated bearer and send the packet filters to UE as well. When a new flow comes with first packet as Uplink, UE does bearer matching based on the TOS/DSCP value, and sends flow on the correct dedicated bearer. For downlink packet, ECS does bearer lookup and assigns correct bearer to the flow based on the TOS/DSCP value.

ADC Event reporting will contain flow template with outer IP 3 tuples (Source IP, Destination IP, Port). L4-L7 rule match will also work for PMIP service.

Limitations

The limitations for the ADC over Gx feature are:

- Registration of the duplicate application IDs are not supported.
- Readdress/Redirection for P2P flows will not be supported.
- Redirection happens only on transactions of GET/Response.
- Port based, IP Protocol based, and URL based applications are not supported.
- Pre-configured options (precedence, redirect-server-ip) for dynamic ADC Rules are not supported.
- Simultaneous instances of an application for the same subscriber are not distinguished.
- Flow recovery is not supported for application flows.

Dynamic Advertisement Server Correlation

ADC supports many streaming applications that are ad-supported and the flows corresponding to these third-party advertisements are generic. These advertisement flows could not be differentiated from specific

application flows based on the deterministic pattern. As part of this feature, a configurable option is provided to dynamically correlate advertisement flows and associate the respective applications.

Any advertisement service is associated with the corresponding application protocol. The type of ad-flow will be configured per application. Refer to the *Configuring P2P Advertisement server* section in the *Application Detection and Control Configuration* chapter for more information on configuring the P2P Advertisement Server correlation group.

Limitations

Some limitations of this feature are listed below:

- Maximum number of ads groups that can be configured is 100.
- Maximum number of ad-source lines per ads-group that can be configured is 32.
- Configuration will take effect only for new flows.
- Applications added using TLS/SNI ruledefs (custom defined protocols) will not be supported.

Bulk Statistics Support

The system can be configured to collect bulk statistics (performance data) and send them to a collection server (called a receiver). Bulk statistics are statistics that are collected in a group. The individual statistics are grouped by schema. ADC uses P2P schema for bulk statistics support.

Important The bulk statistics format previously supported by the older implementation for individual ADC protocols in ECS schema is deprecated, and the new bulk statistics format is supported in 14.0 and later releases in the new P2P schema.

The P2P schema is designed in such a way that all variables that end with numeral value "name" are used to extract all data with numeral values "value" for all the protocols supported by the currently loaded patch. With the Dynamic Software Upgrade, the operator need not change the P2P schema by adding or removing variables related to a particular protocol manually for each new patch.

The following is a sample configuration of bulk statistics in the P2P schema:

p2p schema p2p *format*

"%p2p-protocol%\n%p2p-protocol-group%\n%p2p-uplnk-bytes-name%:%p2puplnk-bytes-value

%\n%p2p-dwlnk-bytes-name%:%p2p-dwlnk-bytes-value%\n%p2p-uplnk-pktsname%:%p2p-uplnk-pkts-value%

p2p-uplnk-pkts-value%\n%p2p-dwlnk-pkts-name%:%p2p-dwlnk-pkts-value%\n%p2pduration-name%:

%p2p-duration-value%\n----------\n"

Important If detection of a specific P2P protocol is enabled, bulk statistics for that protocol will be automatically generated based on the plugin installed on the chassis. In the case of protocols that support sub-classification (audio/file transfer/instant messaging/video/voipout/unclassified), the bulk statistics will be dynamically generated for each of the supported sub-classifications per protocol and also the corresponding total count which is the sum of values of the sub-classified data.

For more information, see the *P2P Schema* chapter of the *Statistics and Counters Reference*.

How ADC Works

As part of traffic analysis, packets will be first passed through the ADC analyzers when "p2p dynamic-flow-detection" is enabled. If it is not detected as P2P by any of the ADC analyzers, then it will follow the rule matching to find an application analyzer.

ADC analyzers examine uplink and downlink traffic and use rules that define what packet content to take action on and what action to take when the rule is true. The analyzers also generate usage records for the billing system. The rules are configured/defined in the same way as ECS in-line service ruledefs and rulebases.

For a few specific protocols, packets will be sent to non-ADC analyzers even after marking the flow as P2P. If the flow is marked as P2P and also analyzed by other analyzers, the statistics for display and debug purposes reflect in both analyzers. The EDR also displays the ADC application/protocol names if configured.

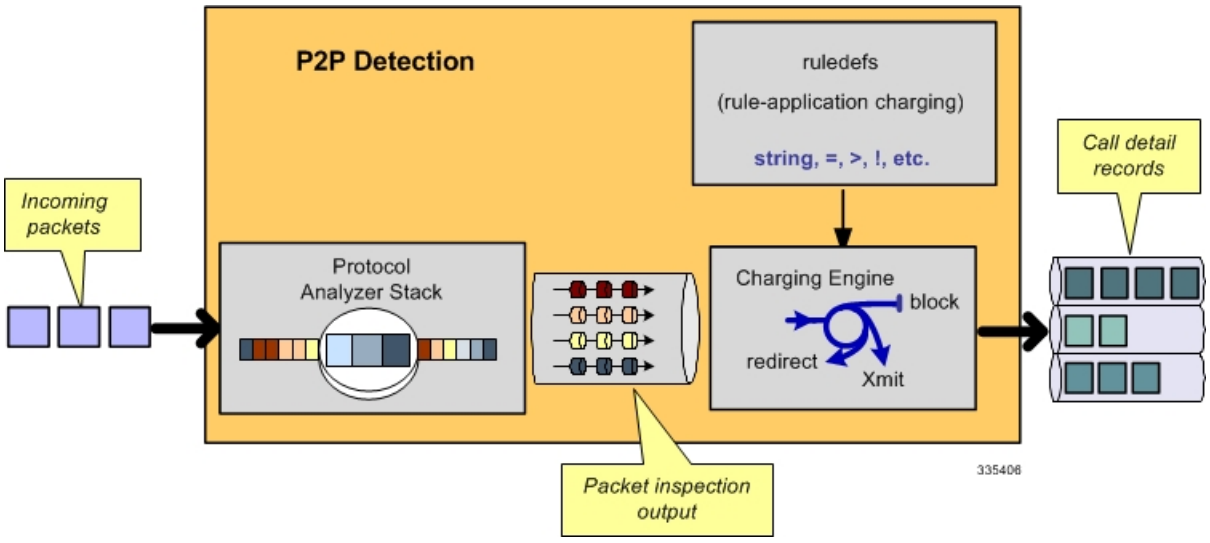
ADC also interfaces to a PCRF Diameter Gx interface to accept policy ACLs and rulebases from a PDF. ADC supports real-time dynamic policy updates during a subscriber session. This includes modifying the subscriber's policy rules during an active session by means of ACL name and Rulebase name.

In Gx interface, a Charging Rulebase will be treated as a group of ruledefs. A group of ruledefs enables grouping rules into categories, so that charging systems can base the charging policy on the category. When a request contains names of several Charging Rulebases, groups of ruledefs of the corresponding names are activated. For ADC rules to work in the group of ruledefs, P2P detection has to be enabled in the rulebase statically.

Static policy is supported initially. A default subscriber profile is assumed and can be overwritten on the gateway. Per-subscriber static policy is pulled by the gateway from the AAA service at subscriber authentication.

The following figure illustrates how packets travel through the system using ADC. The packets are investigated and then handled appropriately using ruledefs for charging.

Figure 1: Overview of Packet Processing in ECSv2



Limitations

This section lists the limitations for the ADC protocols that support audio and video sub-classification.

If Audio and Video contents are in the same flow (TCP/UDP), video is considered as the predominant component and the flow is marked as "video". In this scenario, throttling video will block both audio and video. Throttling only audio or video is not possible.