RayV Grid rules

Feature explanation

Table of Contents

Scope

The purpose of this document is to provide a detailed explanation of the 'Grid Rules' feature that enables controlling various aspects of the RayV service. In addition to the list of properties that can have rules assigned to them, enclosed a section with the parameters that are configured and affected by the rules.

Overview

Grid Business Rules allows controlling multiple aspects of Grid Service behavior by means of textual DSL-like language.

Many Grid Service features use business rules. Business rules are a means of controlling their behavior at runtime, rather than hard-coding it in compile time.

By using a common infrastructure for rules, whenever the Business Rules framework is extended to support new properties, multiple Grid Service features benefit from them at once.

Example Features

Some of the features that use Business Rules are: *P2P Policies (who will contribute to whom)*, *Dynamic allocation of Edge-Nodes, Adaptive bitrate per viewer*.

Rule Conditions

The *condition* of the rule is encoded in the language describe below, while the *behavior* of the rule is the response of some particular feature to the evaluation of the condition at runtime.

Kinds of Feature Rules

There are several kinds of business rules:

Kind	Function	Comments

Session-Channel rules	Control behavior with respect to an online client in the context of a channel	The most common rule kind
Session rules	Control behavior on the level of the login session, regardless of channels	Does not have access to *Channel.** properties
Login rules	Affect only the scope of authentication	Has access only to the session's nickname, client type, and version information

Each rule belongs to one of these kinds. For example *Client Policy* rules are of *session-channel* kind.

Reference

Session Properties

Property	Description	Example
Session.ApplicationVersion	The *appVersion,* as sent by grid clients.	Session.AppVersion >= "3.0"
Session.ClientType	The client type, as provisioned in *Named Users* screen. Valid values: • *"Player"* • *"Broadcaster"* • *"Probe"* • *"Gateway"* • *"EdgeNode"*	Session.ClientType = "Player"
Session.City	The user's approximate city.	Session.City = "Jerusalem" Retrieved from MaxMind
Session.Cpu	The CPU usage	Session.Cpu > 70
Session.Distributor	The player's distributor.	Session.Distributor = "HBO"
Session.ExternalIp	The external IP of the session.	Session.ExternalIp = "21.32.4.200"
Session.ExternalIpCountry	The short name of the session's country, determined based on the external IP.	Session.ExternalIpCountry = "IL" Retrieved from MaxMind
Session.ISP	The full registered name of an ISP.	Session.Isp = "Bezeq International" Retrieved from MaxMind
Session.JoiningUrl	The container URL of a channel being joined, before it is joined. Used only by *Security Blocking Rule.*	Session.JoiningUrl.Contains(".HBO.com/")
Session.Memory	Memory (in MegaBytes) reported by application, otherwise zero.	Session.Memory > 1000

Session.NatType	The NAT type, as a 1..5 number. Valid values: • *1 - Open* • *2 - IP-Filtered* • *3 - IP&Port-Filtered* • *4 - Destination-Specific Ports* • *5 – Blocked*	Session.NetType > 3
Session.Nickname	Peer ID. Note that for named users, the peer ID is the nickname.	Session.Nickname = "bc_ab1"
Session.PostalCode	Postal code, where available.	Session.PostalCode = "abc123"
Session.Product	The product ID sent by clients.	Session.Product = "IPhone"
Session.Region	The country's region, state or province.	Session.Region = "04"
Session.TransportIp	Transport-determined IP address.	Session.TransportIP = "202.12.44.1"
Session.Version	The version must appear as a non-empty string, such as *"1.2", "1.2.5","1.2.5.0".*	Session.Version > "1.2"
Session.Zone	The session's zone key.	Session.Zone = "uk"

Channel Properties

Below is a list of channel property. Note that all properties are **empty** if the session is not joined to any channel.

Property	Description	Example
Channel.BandwidthCategory	The bandwidth category key	Channel.BandwidthCategory = "normal"
Channel.EnableRecording	*True* if *Engine Stats* in *Channels* screen is set, false otherwise.	Channel.EnableRecording = true
Channel.Key	The joined channel's key.	Channel.Key.StartsWith("HBO_")
Channel.Profile	The channel's profile key.	Channel.Profile = "HD"
Channel.Publisher	The channel's publisher key.	Channel.Publisher = "HBO"

Parameters

Some Important engine parameters that control the peers' behaviors

Name	Description	Example values
General:		
• *PCT1secs*	how often should nodes monitor peering status	3 sec
• *FreeVectorSize*	the maximal size of net data in data packet	1150Bytes
• *MaxNumBitsEqs*	the maximal size of segment in free vectors–Segment size	50bytes
• *MaxSegmentDurationMs*	maximal media duration of segment	750ms
• *BufferingTimeoutMs*	Buffer size	10000ms
• *MinSegmentTimeMs*	minimal durations between segments	300ms
• *MaxSatellites*	Number of edge nodes to connect to	2 if viewer, 1 if amp
• *MaxHigherSats*	Max number of sats from higher layers	1 if viewer, 0 if amp
When viewer:		
• *SchWakeupEveryMs*	how often should viewers monitor segments status	200ms
• *DynamicDonorsMaxConnections*	maximal number of client donor for viewer	100
• *DynamicDonorsStepUp*	peer connect agressiveness, size of jump	3
• *MaxHigherSats*		
When amplifier:		
• *MaxQuota*	number of peers a node will allow as acceptors	20
• *IncreaseRatio*	multiple connection congestion control agressiveness (upload) jump	5
• *AmplifyMinNumAcceptorsToOrder*		2