Содержание

4 FastPCRF settings for Radius servers

Radius servers within the radius_server list are unequal: the first one is considered to be the main Radius server, whereas the rest are considered to be as backup servers. If FastPCRF detects that the main Radius server is not responding for a long time, the corresponding will be reset and FastPCRF will connect to the next radius server from the list. However, the FastPCRF periodically attempts to connect to the main radius server until it becomes available.

Parameter	Format	Default value	Description
default_reject_policing	string	no	The default policing profile name for unauthorized users
default_reject_whitelist	string	no	Service 5 (White list) profile name used by default for unauthorized users.
radius_revive_period	seconds	120	Periodicity of the reconnection to the main Radius server.
radius_max_pending_requests	number	1000000	The maximum number of pending requests from the FastDPI servers. When this threshold is exceeded, the incoming requests from the FastDPI servers will not be handled.
coa_max_pending_requests	number	100000	The maximum number of pending CoA requests from the Radius servers. This value should not be higher than the value of the `async_queue_size` parameter; the recommended value should not be more than `async_queue_size` / 2.

Parameter	Format	Default value	Description
radius_server	secret@ip%dev:port{;param=value}*	no	Specifies a single Radius server and its configuration parameters: `secret` - the secret of the Radius server; IP - the Radius server IP address; dev (optional) - the name of the interface used to establish connection; if it is not specified, then the interface will be selected by the operating system; `port` is the port; `param=value` is the configuration parameters (via semicolon) for a given Radius server. See the Radius_server parameter description

Radius_server parameter description

Each Radius server is described by a separate `radius_server` parameter in the configuration file. At least two radius servers are typically specified: the primary Radius server and the backup one, it implies that there must be at least two lines with the `radius_server` parameter, i.e. one line for the primary Radius server and second one - for the backup servers. The maximum number of Radius servers is 16. The Radius server specified in the configuration file in the first place is considered as the main server, whereas the rest ones (specified within the configuration file later on) are considered the backup ones. Backup servers are used when the main one is unreachable in the order in which they are specified within the conf file. At any given moment, only one Radius server is active.

The configuration parameters of the radius server can be specified in three ways:

1. Values that are the same for all the Radius servers - they are specified as normal parameters in the fastpcrf.conf file. The main idea is that they must be specified before the `radius_server` parameters, only in this case they will be applied to all the Radius servers.
2. You can create your own configuration file for each radius-server, whose name is specified by the `conf` parameter within the `radius_server` line, for example:

   ```
   radius_server=secret@10.10.3.5:1812;conf=radius-main.conf
   ```

   In this example, the values from radius-main.conf take precedence over the values of common Radius servers parameters.

3. Parameters unique to a given radius server can be specified explicitly in the `radius_server` line, for example:

```
radius_server=secret@10.10.3.5:1812;conf=radius-
main.conf;msg_auth_attr=1
```

In this example, the `msg_auth_attr` parameter is specified for a given 10.10.3.5 server and overrides the value of the corresponding parameter in the radius-main.conf configuration file. Note that the order in which the parameters appear within the `radius_server` line is very important: the parameters will by applied following the exact order they are specified in the `radius_server` line. If you invert `conf` and `msg_auth_param` in the example above and if `msg_auth_param=0` within the radius-main.conf configuration file, then `msg_auth_param=` 0 will be used from radius-main.conf instead.

Individual parameters of Radius servers

Parameter in fastpcrf.conf	Parameter in `radius_server`	Format	Default value	Description
radius_dead_timeout	dead_timeout	seconds	60	If within this period there is no responses from the Radius server, whereas requests are being sent, then the server is considered to be inactive, so the FastPCRF switches to the next Radius server from the list. If the main radius-server does not respond, then the `radius_revive_period` timer will be started and after its expiration a reconnection attempt to the main Radius server will be made.
radius_max_connect_count	max_connect_count	number	16	The maximum number of connections to one Radius server. According to RFC 2865, an identifier allowing you to compare request and response occupies 1 byte field, which suggests that one connection can simultaneously handle up to 256 requests. To overcome this limitation, the specification suggests creating several connections to a single Radius server. Actually, this parameter specifies the number of simultaneous requests to one Radius server:`radius_max_connect_count * 256`.
radius_response_timeout	response_timeout	seconds	30	Timeout for a response to an Access-Request request to the Radius server. If the response to the request hasn't been received within this period of time, the request is considered to be dropped by the Radius server (for example, due to “too much simultaneous requests”) and fastpcrf tries to send the request again.
radius_resend_count	resend_count	number	0	The maximum number of attempts to resend the requests. If the number of attempts to resend the requests is exhausted and the response from the Radius server is not received, fastpcrf does not report anything to the fastdpi server. If there is no response to authorization request within a certain timeout (the `auth_resend_timeout` parameter in the fastdpi.conf file) the Fastdpi will send a renewed authorization request.

Parameter in fastpcrf.conf	Parameter in `radius_server`	Format	Default value	Description
radius_status_server	status_server	boolean type	1	The parameter specifying whether the Radius server supports the Status-Server request defined in the RFC 5997. This request type is used by fastpcrf to ping the Radius server, especially when the main radius server is temporarily unavailable. Without Status-Server support, it is very difficult to understand that the main Radius server has recovered.
radius_user_password	user_password	string	VasExperts.FastDPI	User-Password attribute value in the Access-Request request.
radius_user_name_auth	user_name_auth	string	login,ip,qinq	Starting with the VAS Experts DPI version 7.4, the `radius_user_name_auth` parameter specifies the value of the User-Name attribute in the order of preference: login - to use the subscriber login ; ip - to use the subscriber IP address; qinq - to use the QinQ tag using the following format: «outerVLAN.innerVLAN»; for example, «101.205»
radius_unknown_user	unknown_user	string	VasExperts.FastDPI.unknownUser	User login in case its real login is unknown to the FastDPI. This is the value of the User-Name attribute of the Access-Request request in case `radius_user_name_ip = 0` and the user login is unknown. It is assumed that the Radius server in the Access-Accept response will specify the real user login identified by its IP address extracted from the Framed-IP-Address attribute. Note that this parameter is closely related to the radius_user_name_auth parameter and is applied only if there are no other ways to specify the User-Name attribute.
radius_unknown_user_psw	unknown_user_pws	string	VasExperts.FastDPI	The value of the User-Password attribute for unknown user login. It is applied only if the `radius_user_name_ip = 0`.
radius_msg_auth_attr	msg_auth_attr	boolean type	1	The parameter specifying whether the Radius server supports the `Message-Authenticator` attribute described in the RFC 2869. If the attribute is supported, the FastPCRF will compute and include a Message-Authenticator in each `Access-Request` and each `Status-Server` request. The FastPCRF also analyzes this attribute in the responses; if the check of Message-Authenticator attribute contained in the response fails, then such a response will be dropped.
radius_attr_nas_port_type	attr_nas_port_type	number	5 (Virtual)	Value of the `NAS-Port-Type` attribute (RFC 2865) within the Access-Request request.
radius_attr_nas_ip_address	attr_nas_ip_address	IPv4 address	no	The value of the `NAS-IP-Address` attribute specified in the RFC 2865 in the Access-Request. If it is not specified, the `NAS-IP-Address` attribute will not be included in the request.

Parameter in fastpcrf.conf	Parameter in `radius_server`	Format	Default value	Description
radius_attr_nas_id	attr_nas_id	string	no	The value of the `NAS-Identifier` attribute in the Access-Request request. According to RFC2865, either a `NAS-IP-Address` or a `NAS-Identifier` should be specified in the Access-Request.
radius_attr_service_type	attr_service_type	number	2 (Framed)	The value of the `Service-Type` attribute from RFC 2865 Access-Request.
radius_attr_cui	attr_cui	boolean type	1	The parameter specifying whether the Radius server supports the `Chargeable-User-Identity` (CUI) attribute from RFC 4372. If this attribute is supported, then the FastPCRF places the user login in this attribute within the Access-Request request; if the login is unknown, then the attribute will contain a zero byte, which means, according to RFC 4372, a login request from the radius server. The FastPCRF expects the real user login will be sent within the Access-Accept response attribute. The real user login can be determined by the Radius server by its IP address (the `Framed-IP-Address` attribute contained in the request).
radius_coa_port	coa_port	UDP port	3799	UDP port used to receive the Disconnect-Request Change-of-Authorization (CoA) notifications and CoA-Request specified by RFC 5176. If the Radius server does not support CoA, you should set this parameter to 0.
radius_coa_resend_timeout	coa_resend_timeout	seconds	1	Timeout for CoA responses resending (Disconnect-ACK, Disconnect-NAK, CoA-ACK, CoA-NAK) in case of problems with the socket (usually caused by the socket queue overflow).The number of retries is specified by the `radius_resend_count` parameter.
coa_reauth_ack	coa_reauth_ack	boolean type	0	How to respond to a CoA-Request with `Service-Type=8` (Authenticate-Only): 0 (default value) - according to RFC5176 p.3.2: it should by replied with CoA-NAK contained `Error-Cause=507` (Request Initiated); 1 means non-standard behavior, so it should by replied with CoA-ACK