Interactive Remote Authentication Dial In User Service (RADIUS) Authentication Server Model

Rohan Deshmukh
CISCO Systems Inc., India
email: rodeshmu@cisco.com

***Abstract*–Normally, RADIUS servers are passive servers, i.e., they act only on requests received from Network Access Server. In a system where there are multiple servers configured in round-robin fashion, if some of the servers go down, it takes more time to reach the actual active server after retransmissions to the non-responsive server get exhausted. Here, we present a new approach to make RADIUS Server more Interactive Server. It sends ACTIVE-Request to the Network Access Server once it becomes active and DEAD-Request once it becomes non-responsive.**

Keywords-RADIUS; NAS; ID; Attributes.

I. INTRODUCTION

RADIUS servers are being used for AAA (Authentication, Authorization and Accounting) purpose [1]. With manual intervention, RADIUS server can send CoA (Change of Authorization) and DM (Disconnect Message) to the Network Access Server (NAS) [2]. The RADIUS client function may reside in a Gateway GPRS Support Node (GGSN). When the GGSN receives a Create PDP Context request message, the RADIUS client function may send the authentication information in the request "Access-Request" to an authentication server, which is identified during the Access Point Name provisioning [3]. The NAS sends an Access-Request packet to the RADIUS Server with NAS-Identifier, NAS-Port, User-Name and User-Password. The RADIUS server then sends back either an Access-Accept or Access-Reject based on whether the response matches the required value, or it can even send another Access-Challenge. Figure 1a describes this.

If the RADIUS server does not send any response, the NAS re-transmits the request to the same server without change in its attributes like Request Authenticator, ID, and source port. If any attributes have changed, a request is generated with new Request Authenticator and ID. Use of Status-Server Packets in the RADIUS protocol is mentioned in [4]. But, it is for *clients* to query the status of a RADIUS server. While the Status-Server (12) code was defined as experimental in [1], Section 3, details of the operation and potential uses of the code are not provided.

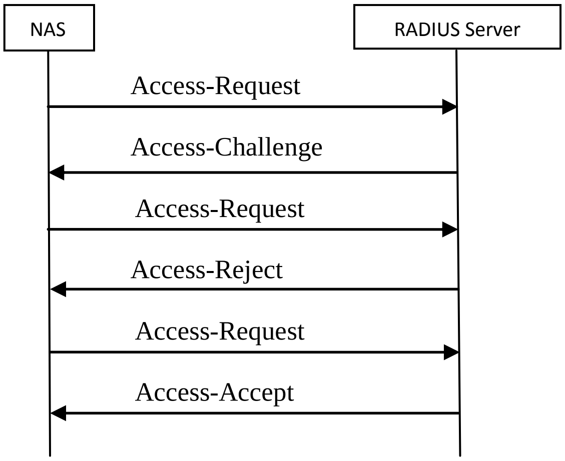

Figure 1a. RADIUS Auth Messages

The RADIUS server can send CoA and DM requests only. CoA-Request packets contain information for dynamically changing session authorizations. The NAS responds to a CoA-Request sent by a RADIUS server with a CoA-ACK if the NAS is able to successfully change the authorizations for the user session, or a CoA-NAK if the request is unsuccessful. A Disconnect-Request packet is sent by the RADIUS server in order to terminate a user session on a NAS to discard all associated session context. Figure 1b describes this.

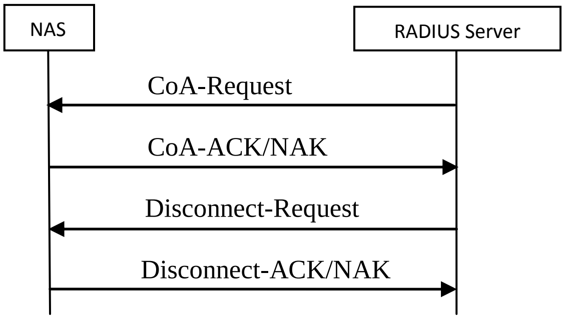

Figure 1b. RADIUS CoA/DM Messages

ISBN: 978-1-61208-203-5

Selection of RADIUS server is based on the following algorithms:

- Round-robin: designates that the context should load-balance sending data among all of the defined RADIUS servers in a cyclic manner.
- RADIUS first-server: designates that context sends data to the RADIUS server with the highest configured priority. In the event that this server becomes unreachable, data is sent to the server with the next-highest configured priority.

NAS is normally configured with:

- Max-retries: maximum number of times system will attempt to retry communications with a server before system fails over to a backup RADIUS server.
- RADIUS timeout: how long system will wait for a response from a RADIUS server before re-transmitting the request.
- Detect dead after 'x' consecutive failure: if AAA server is unreachable consecutive number of times then mark it as a dead server.

When RADIUS server does not respond to the request from NAS, NAS retransmits the same request to the same server until max-retries are exhausted. NAS marks that server as unreachable and tries to send the request to another server if configured. The new session request again goes to the same 1st server; retries until max-retries exhaust. In this process, if detect dead after 'x' consecutive failure exhaust, then NAS marks this server as dead.

Here, we assume RADIUS server as a remote server only and not as a Proxy server and also a system comprising of multiple RADIUS servers where selection of a server is based on round-robin algorithm.

Normally, in a dense area where there are more number of customers sending Create PDP Context request message to GGSN, multiple radius servers (more than 10) are used in round-robin fashion for load balancing.

II. PROPOSAL

With multiple RADIUS servers configured in the system, sending a keep-alive message is strongly discouraged, since it adds to load and harms scalability without providing any additional useful information [1]. When multiple servers are used in the network and if some of the servers go down (not responding), it is really time consuming to send request to each and every server if round-robin is used till it reaches the active server. With all the RADIUS requests going to multiple non-responsive servers, it also adds to load in the network. Figure 2 explains this.

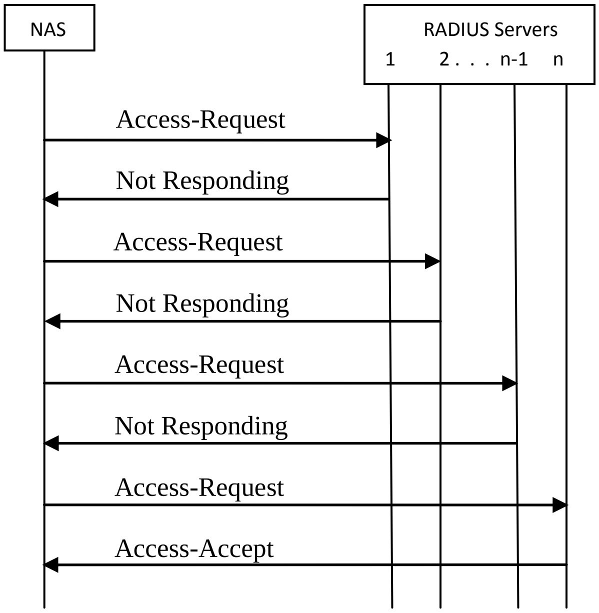

Figure 2. Non responding servers in Multi-Server System

Here, we propose to send an ACTIVE-Request from RADIUS server to NAS in case RADIUS server becomes active from dead state or becomes active for the first time. ACTIVE-Request will also add an attribute which will be its server ID. Also the new attribute values "Active Server ID" and "DEAD Server ID" for Type: 26 (Vendor-Specific) are introduced. This is based on [6].

```
12 Code = ACTIVE-Request
    1 ID = 71
    2 Length = 52
   16 Request Authenticator

   Attribute Type: 26 (Vendor-Specific)
          Length: 12
          Vendor Type: ACTIVE
          Vendor Length: 6
          Value: 00 00 01    Active Server ID
```

NAS will store the ID of the last active server and will update its priority table. So, when new request comes, NAS will look into its priority table and selects the active server in round-robin fashion. When some of the servers do not respond in a system of multiple RADIUS servers, with this new mechanism, NAS will select the appropriate active server first as it stores ID of the last active server and update the priority table. So, in meantime, if any of the non-responsive servers becomes active, it will send ACTIVE-Request to NAS. Then NAS comes to know about new active server, stores its ID and then sends the next new session request to this newly active server if its priority is less than the last active server else it will be selected in normal round-robin fashion. This way it will rather reduce the load in the network by avoiding the retransmissions to the non-responsive server.

Algorithm:

```
If last Active Server received ID < Last Active Server
   Last Active Server received ID = = Last Active server
       Then follow the round-robin
Else place last Active Server received ID in round-robin
queue matching priority and follow round-robin
```

The value of the attribute will be proprietary which will contain information about the last active server. So, NAS will select the active server for next session request to be sent based on this information and comparing the round-robin priorities.

After server sends ACTIVE-Request, NAS will send new request to this active server rather than trying to send requests to non-responsive servers. In Figure 3, server (n-1) sends ACTIVE-Request. So, NAS sends a new request to (n-1)th server directly; thus saving on requests and retransmissions of (n-2) servers considering all other servers before (n-1) are non-responsive.

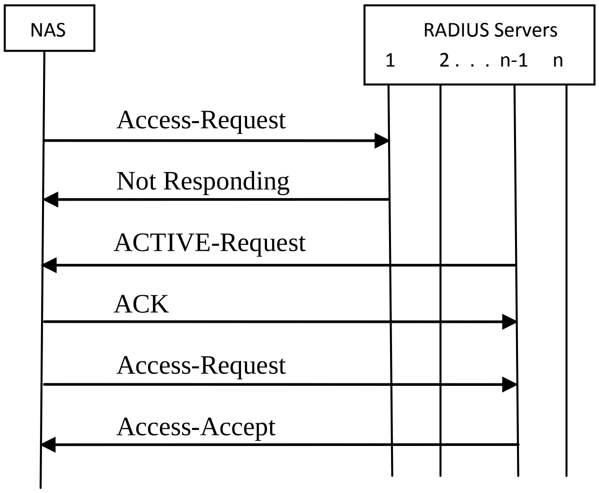

Figure 3. Server sends ACTIVE-Request

Let us assume following configuration at NAS:

Radius round-robin algorithm
Max-retries: 5
RADIUS timeout: 3 seconds
Detect dead after consecutive failure: 4
RADIUS Servers configured: 12
RADIUS dead time: 10 minutes
Keep alive not configured

In normal scenario, NAS will send request to 1st top priority server and that server will respond; 2nd request will go to 2nd server and so on.

Now, assume that all the servers are not responding except 12th server. So, as per above configuration, NAS will keep on sending request to 1st server until max-reties get exhausted. As radius timeout is 3 seconds, for each server radius request will take 18 seconds ((1+5) retries * 3 sec timeout).

For 1st request to reach 12th server, it will take (18 sec * 11 servers) = 198 seconds (3 minutes 18 seconds).
For 2nd request to reach 12th server, it will take (18 sec * 10 servers) = 180 seconds (3 minutes 0 seconds) and so on.

As detect dead after consecutive failure is 4, it will take 198 * 4 = 792 seconds = 13 min 12 seconds, before it marks 1st server as dead. After this, any request that comes from NAS will not be sent to 1st server but to 2nd server. As 2nd server also does not respond, this cycle repeats till it marks 2nd server as dead. This time it will take (198 -18) * 4 = 720 seconds = 12 minutes.

In this way, NAS will mark all servers dead except 12th one. As NAS is configured with no keep alive, NAS will never try to send any request to 1st server even if it is active or to any other server which is active until RADIUS dead time expires (which is 10 minutes in this case).

If, for example, 10th server becomes active, with new proposal, it will send ACTIVE-Request. NAS will then select this active server to send any new session request, thus saving on retransmissions. In meantime, if another server (e.g., 2nd server) becomes active, it will also send ACTIVE-Request. Then NAS will compare its priority table to select the appropriate active server for new request.

Selection of the active server can also be achieved by reducing the number of requests and retransmissions within a system of multiple RADIUS servers. But, this will not give fair amount of time to a particular server if that server becomes active before retransmissions are exhausted. Then, after retransmissions to non-responsive server are exhausted, there will be another round of retransmissions to the next selected server if that is also non-responsive. This will again add load in the system of multiple servers, even if number of requests and retransmissions are less.

We also propose to send DEAD-Request when the RADIUS server goes down. But, it has some limitation. In some scenarios like power outage or kernel panic, server does not get any chance to send any information.

13 Code = DEAD-Request
  1 ID = 71
  2 Length = 52
  16 Request Authenticator

  Attribute Type: 26 (Vendor-Specific)
    Length: 12
    Vendor Type: DEAD
    Vendor Length: 6
    Value: 00 00 01 DEAD Server ID

After receiving DEAD-Request, NAS will update its priority table and then selects the next available active RADIUS server for the new request. In this case, NAS will not retransmit the request to the dead server. If this dead server again becomes active, it again sends ACTIVE-Request and cycle repeats. Figure 4 shows this. In Figure 4, assume that server 2 is non-responsive and it sends DEAD-Request. Then NAS will send new request to next active server after 2nd, i.e., (n-1)th server, by looking into its priority table thereby saving on requests and retransmissions to 2nd server.

If 2nd server again becomes active, it will send ACTIVE-Request to NAS. As 2nd server's ID is less than the (n-1)th server, NAS will send new request to 2nd server instead of (n-1)th server.

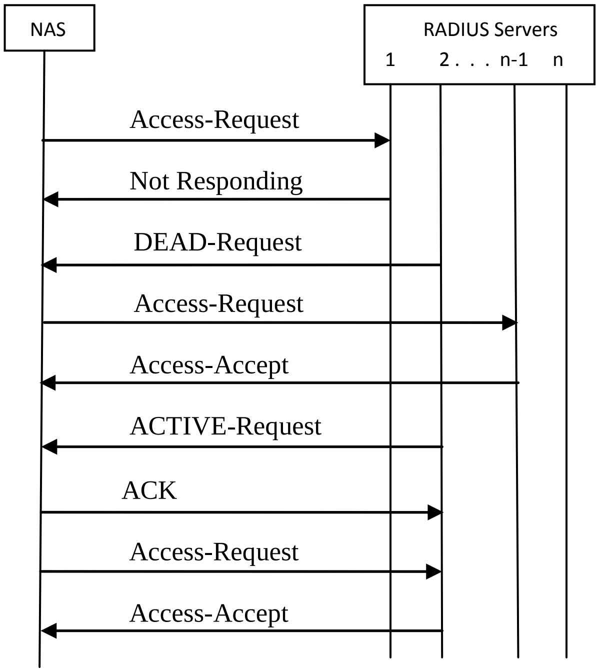

Figure 4. Server sends DEAD-Request

If we reduce the RADIUS dead time to minimum value (around 1 to 2 minutes), then NAS will send new requests to all dead servers in round-robin after this timer expiry and if it does not get any response then again marking of server dead cycle will start.

III. CONCLUSION AND FUTURE WORK

This proposal attempts to improve the communication efficiency between NAS and RADIUS server. It does so by allowing the RADIUS server to communicate its state (active/dead) to NAS. In some circumstances, the server does not get an opportunity to send anything when it goes offline like in case of power outage or kernel panic. But this proposal will help in effectively selecting RADIUS server.

Future work of this proposal includes simulation model of interaction between NAS and RADIUS server. It will also include the performance evaluation to support the concept in real system.

This proposal will help to select active server properly by avoiding retransmissions to the non-responsive servers thereby causing less CPU utilization in the network.

REFERENCES

[1] C. Rigney, S. Willens, A. Rubens, and W. Simspson, "Remote Authentication Dial In User Service (RADIUS)", IETF RFC 2865, June 2000.

[2] M. Chiba, G. Dommety, M. Eklund, D. Mitton, and B. Aboba, "Dynamic Authorization Extensions to Remote Authentication Dial In User Service (RADIUS)", IETF RFC 3576, July 2003.

[3] 3GPP TS 29.061 V9.2.0 (2010-03), page 56.

[4] A. DeKok, "Use of Status-Server Packets in the Remote Authentication Dial In User Service (RADIUS) Protocol", IETF RFC 5997, August 2010.

[5] 3GPP TS 32.295 V8.1.0 (2009-09), pp. 21-24.

[6] D. Mitton, "Network Access Servers Requirements: Extended RADIUS Practices", IETF RFC 2882, July 2000.