Authentication Authorisation Accounting (AAA) Protocols: A Look at RADIUS and DIAMETER

Introduction

Authentication, **Authorisation** and **Accounting** (AAA) processes are used when the user of a system that is trying to connect to the internet or other network. Such processes provided the network with, as the name suggests: **Authentication--**verifying the identity of the user; **Authorisation--**determining whether the requesting entity is allowed access to a resource; and **Accounting--**collection of information on resource usage for the purposes of capacity planning, auditing, billing or cost allocation.

This concept has been standardised by the IETF in various RFC's such as **Generic AAA Architectures** RFC2903, **AAA Authorisation Application Examples** RFC2905 and **AAA Authorisation Framework** RFC2904.

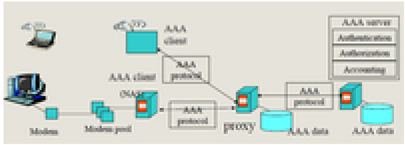

Figure 26. Example AAA Scenario

In general an AAA protocol should support AAA communication between AAA clients and servers. Figure fig:aaa:example contains an example scenario. Three well known AAA protocols
are: **TACACS** RFC1492, **RADIUS** RFC2865 and **DIAMETER**RFC3588. This chapter will provide an overview of both RADIUS and DIAMETER.

4.2. RADIUS

RADIUS is an AAA protocol used to carry AAA information between a **Network Access Server** (NAS) an AAA Client and a shared AAA Server. It operates in a Client/Server model in that the NAS generates an AAA request and forwards this onto the RADIUS Server. A RADIUS server can also operate as a forwarding proxy. Thus RADIUS can be deployed/used in two ways:

Non-Proxy this is when a client will communicate directly with the required server.

1. **Proxy** the AAA server cannot handle the request and forwards it on to the required server that can handle the request.

4.2.1. Key Features

Some key features of RADIUS include:

Network Security

Any and all transactions are authenticated using a shared secret key and are **manually** distributed between the client and servers. RADIUS will obscure user passwords using MD5 (RFC1321) and other techniques. Given this End-to-End Security can not be guaranteed when RADIUS is in proxy mode, however there are some assurances when used in non-proxy mode. This is further explored in Section aaa:radius:e2e.

Authentication

The authentication mechanisms are flexible, PPP CHAP Challenge Handshake Protocol, UNIX and EAP Extensible Authentication Protocol, can all be used.

Protocol

Furthermore the protocol itself is fairly extensible, transactions are comprised of variable length Attribute-Length-Value triples. New attribute values can be added without disturbing existing implementations.

Transport

Interestingly, RADIUS has been built on top of UDP. This is explored more in Section aaa:radius:udp

4.2.2. Document Architecture

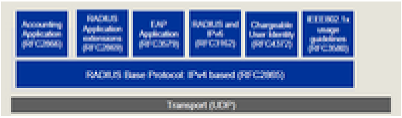
Accounting Application (RFC2866)
RADIUS Application extensions (RFC2869)
EAP Application (RFC3579)
RADIUS and IPv6 (RFC3162)
Chargeable User Identity (RFC4372)
IEEE802.1x usage guidelines (RFC3580)
RADIUS Base Protocol IPv4 based (RFC2865)
Transport (UDP)

Figure 27. RADIUS Overview

The default RADIUS specification, built upon IPv4, defines the base protocol for authentication and authorisation, Accounting is added through use of a separate application. Various extensions that support various AAA applications extensions are also defined separately. RADIUS also has an EAP application module for the support of various authentication methods. Moreover specifications for chargeable user identity attributes for use in roaming. Finally there are guidelines for RADIUS usage in wireless LANs.

4.2.3. Message Types

Some of the codes and message types include:

Table 1. CSV data, 15% each column

1	Access-Request
2	Access-Accept
3	Access-Reject
4	Accounting-Request
5	Accounting-Response
11	Access Challenge
12	Status-Server //experimental
13	Status-Client //experimental
255	Reserved

4.2.4. Session Establishment and Termination

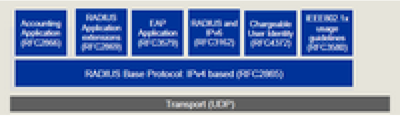
Accounting Application (RFC2866)
RADIUS Application extensions (RFC2869)
EAP Application (RFC3579)
RADIUS and IPv6 (RFC3162)
Chargeable User Identity (RFC4372)
IEEE802.1x usage guidelines (RFC3580)
RADIUS Base Protocol IPv4 based (RFC2865)
Transport (UDP)

Figure 28. RADIUS Session Overview

RADIUS utilises the same session for both the authentication/authorisation and accounting messages. Figure fig:aaa:radius:session summarises the session establishment and termination.

4.2.5. On use of UDP

There are various advantages and disadvantages of using UDP over TCP for transport:

Advantages

> If the request to the primary authentication server fails, a secondary server must be queried. Hence a copy of the request must be kept above the transport layer and retransmission timers are still required, again above the transport layers. Given the stateless nature of RADIUS within a communication network, UDP simplifies the operation. Transport connection between client/server remains if network failures are occurring.

Disadvantages

> Primarily the use of UDP implies that transport is not reliable, the layer above has to take care of packet loss i.e. more work. Moreover TCP, by design, can adapt to network congestion, UDP cannot.

4.3. DIAMETER

DIAMETER is a successor to RADIUS and also provides an AAA framework and solution for applications that need network access or IP mobility. It utilises the same modes as RADIUS i.e. proxy and non-proxy. DIAMETER is a peer-to-peer based protocol in that any node can initiate a request and can be either one of the aforementioned entities. Moreover a node can act as an agent for certain requests, while acting as a server for others. The core base protocol, provides the minimum requirements needed and facilitates for activities such as user session handling or accounting. For instance, it may be used on its own for accounting or in conjunction with another DIAMETER application i.e. MobileIPv4. This base protocol can be extended for use in other applications.

4.3.1. Nodes and Agents

There are three entities within DIAMETER:

Client: a node at the edge of the network performing access control, it generates DIAMETER messages for AAA of the user.

Server: a node that performs the actual AAA of the user.

Agent: a node that does not authenticate and/or authorise messages locally, there are several types of agents:

Relay

forwards a message to an appropriate destination. It can aggregate requests from different realms/regions to a specific one. This eliminates the overhead of network access for every server change.

Proxy

similar to a relay, in that it forwards messages, but a proxy has the ability to modify the content i.e. enforce local rules, admin tasks etc.

Redirect

a centralised configuration repository for other DIAMETER nodes. It redirects requests based upon the routing information it stores. This is useful as other nodes can utilise a redirect node to find other nodes and thus not need to store information locally.

Translation

a special agent, its purpose is to translate AAA messages from one format i.e. RADIUS, to another i.e. DIAMETER. This is useful for the integration of user databases of different application domains, whilst keeping original AAA protocols.

4.3.2. Key Features

Some key features of DIAMETER include:

Capabilities Negotiation

The first messages exchanged after connection establishment are for capability exchange. Capabilities Exchange message carries a peer’s identity and its capabilities (protocol version number, supported Diameter applications, etc.). A Diameter node only transmits commands to peers that have advertised support for the Diameter application associated with the given command.

AVP

AVP are used by the base protocol to support the transportation of user authentication and service specific authorisation information. The core protocol must also exchange resource usage information, as used for accounting purposes. Finally, relaying, proxy and redirecting of messages is supported through a server hierarchy.

Network Security

DIAMETER by default uses IPSec for operation, thus Hop-by-Hop security is guaranteed. DIAMETER also has the option of using TLS thus ensuring end-to-end security. Each hop is authenticated via a shared secret key, distributed using an automatic method i.e. Internet Key Exchange. For this either HMAC-MD5-96 RFC2403 or HMAC-SHA1-96 RFC2404 can be used. Each hop is encrypted using symmetric cryptography i.e. DES-CBC.

Transport

DIAMETER is works on top of TCP and SCTP. Clients can support either protocol while agents and servers must support both. Application-level heartbeat messages called the Device-Watchdog-Request and Device-Watchdog-Answer messages are used to proactively detect transport failures. These messages are sent periodically when a peer connection is idle and when a timely response has not been received for an outstanding request.

Failure Procedures

If a transport failure is detected with a peer, a Diameter node attempts to failover to an alternate peer, which means that all pending request messages sent to the failed peer will be forwarded to the alternate peer. A Diameter node periodically attempts to re-establish the transport connection with a failed peer. Should a connection be re-established, a node can failback to this peer (i.e., messages can once again be forwarded to this peer). A failover to an alternate proxy agent may result in the reception of duplicate request messages by the home server.

Session Management

The authentication/authorisation session management can be independent of the accounting session management. Implying that accounting information can be routed to different servers than the authentication/authorisation messages.

Misc

delivery of avps, error notification, extensibility.

4.3.3. Document Architecture

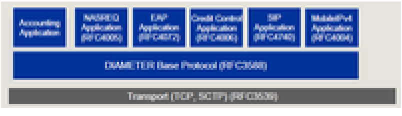
Accounting Application
NASREQ Application (RFC4005)
EAP Application (RFC4072)
Credit Control Application (RFC4006)
SIP Application (RFC4740)
MobileIPv4 Application (RFC4004)
DIAMETER Base Protocol (RFC3588)
Transport (TCP, SCTP) (RFC3539)

Figure 29. DIAMETER Document Overview

The document architecture for DIAMETER is similar to that for RADIUS. However it is built upon, TCP and SCTP instead of UDP. There is support for **NASREQ** for dialin and terminal server applications. EAP is used for authentication support. There is a Credit Control application for Real-time credit control. SIP is supported as well as MobileIPV4. Moreover there is a transport profile document that includes fail over mechanisms and a state machine. Figure fig:aaa:diameter summarises this.

4.3.4. Command Names

Some of the commands used include:

Table 2. CSV data, 15% each column

(Authent/Authz-Request)	
(Authent/Authz-Answer)	
Abort-Session-Request	ASR 274
Abort-Session-Answer	ASA 274
Accounting-Request	ACR 271
Accounting-Answer	ACA 271
Capabilities-Exchange-Request	CER 257
Capabilities-Exchange-Answer	CEA 257
Device-Watchdog-Request	DWR 280
Device-Watchdog-Answer	DWA 280
Disconnect-Peer-Request	DPR 282
Disconnect-Peer-Answer	DPA 282
Re-Auth-Request	RAR 258
Re-Auth-Answer	RAA 258
Session-Termination-Request	STR 275
Session-Termination-Answer	STA 275

4.3.5. Sessions Management

Unlike RADIUS, DIAMETER maintains separate authorisation and accounting sessions and supports re-authorisation. The diagrams in Figure fig:aaa:diameter:sessions provides an overview of session Establishment, renewal and termination.

DIAMETER Session Management Diagrams

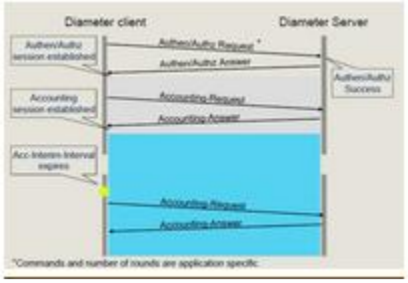

Figure 30. Diameter Session Establishment

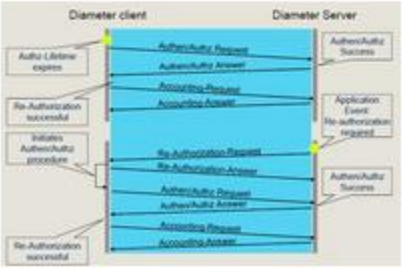

Figure 31. Diameter Re-Authorisation

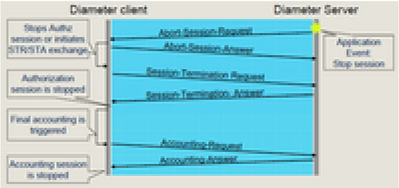
Diameter client
Diameter Server
Stops Authz session or initiates STR/STA exchange
Abort-Session-Request
Abort-Session-Answer
Application Event Stop session
Session-Termination-Request
Session-Termination-Answer
Authorization session is stopped
Final accounting is triggered
Accounting-Request
Accounting-Answer
Accounting session is stopped

Figure 32. Diameter Termination

4.4. Differences: RADIUS and DIAMETER

The differences between RADIUS and DIAMETER are in terms of the advantages that DIAMETER offers are enumerated below:

4.4.1. Transport

- Diameter runs over a reliable transport, TCP or SCTP.
- Lost packets are retransmitted at each hop.

- A persistent connection with an application-level heartbeat message (called a Watchdog message) supports timely failover.
- TCP and SCTP adapt to network congestion.

4.4.2. Proxying

- Hop-by-hop transport failure detection allows failover to occur at the appropriate place—proxies can locally failover to an alternate next-hop peer.
- The proxy automatically does retransmission of pending request messages following a failover.
- An AV pair that identifies the ultimate destination allows multiple transactions for a given session to be routed to the same home server.

4.4.3. Session Control

- Session management is independent of accounting. Accounting information can be routed to a different server than authentication/authorization messages. Session termination is conveyed by a specific Session-Termination message rather than an Accounting Stop message.
- The server may initiate a message to request session termination.
- The server may initiate a message to request re-authentication and/or reauthorization of a user.

4.4.4. Security

- Hop-by-hop security is provided using IPsec or TLS.
- End-to-end security protects the integrity and/or confidentiality of sensitive A-V pairs through intermediate proxies.

4.5. On Translating between RADIUS and DIAMETER

There are at least two types of **Command Codes** that cannot be translated in to an equivalent RADIUS code. They are session related and server-initiated hence the resulting codes are (in their pairs):

4.5.1. Server-Initiated

For example: - RAR-258 — **Re-Auth-Request** - RAR-258 — **Re-Auth-Answer**

Due to the **Peer-to-Peer** (P2P) design of DIAMETER their is client-server in that the client sends a message and the server responds. Hence in RADIUS there is no support for server originated messages that for example aborts the service or demands re-authentication/re-authorisation.

4.5.2. Session Related

For example: - STR-275 — **Session-Termination-Request** - STA-275 — **Session-Termination-Answer**

A major difference between RADIUS and DIAMETER is that DIAMETER is built upon TCP, a stateful, connection-oriented protocol, while RADIUS has been built upon UDP. A result of this is that DIAMETER operates around the notion of sessions, that can be terminated and established. Thus any session related command codes will not have a RADIUS equivalent.

4.6. End-to-End Security

End-to-End security (E2Esec) with respect to Network Security describes the case that the application provides the means to protect the sensitive data from source to destination. This is normally achieved through the use of encryption. Both RADIUS andDIAMETER support, to varying degrees E2Esec.

4.6.1. RADIUS

RADIUS operates via a Client Server model, in which the client, typically a **Network Access Server** will generate an AAA request and send it to a RADIUS server for processing. RADIUS secures the sensitive information (password) through the use of a message digest algorithm i.e. MD5, and transactions are authenticated via a shared secret key that is manually distributed via out-of-bounds channels. All other information such as accounting is sent in the clear. A RADIUS server can also operate as a proxy to a secondary RADIUS server. This usually occurs when the original server is not familiar with the client and/or the secondary server is used as a back-up server. There can be several proxies in between the end server and the client.

E2Esec **exists**, since it is assumed that only the client and server will known their shared secret thus authenticity of messages can be attained. However this is only between a particular client and a server, it can be the case that the same client will authenticate with the same server but via multiple or a singular proxy server. In this case it is quite possible for any of those proxy servers to not only collect sensitive information but to also modify the messages. Thus it is only in non-proxy mode that E2Esec can be provided, yet as it can be seen E2Esec is not necessarily guaranteed.

4.6.2. DIAMETER

Unlike RADIUS, DIAMETER does support full E2Esec in both a traditional Client Server mode and also in proxy mode. This is primarily due to DIAMETER being built around the notion of P2P in which any node can act as both a client and a server. Any DIAMETER node can also act as a proxy (agent) that relays requests between nodes. Because of this P2P design each connection between nodes has to be secured using IPsec and/or TLS, this is also called **Hop-by-Hop Security**.

The result of using IPsec is that each connection is authenticated using a (shared) secret key that is distributed securely using IKE. Moreover as a result of using IPsec the communication between the original client and end server can be protected through the use of encryption. Ergo E2Esec is not only provided but also guaranteed.

4.6.3. Summary

In this chapter, the AAA concept has been introduced and the main characteristics and differences between RADIUS and DIAMETER have been addressed.

4.7. Recommended Reading

The information in this chapter has been based upon the following material:

- **DIAMETER- AAA Protocol**, a presentation by **Hannes Tschofenig** of Nokia-Siemens
- RFC2865 RADIUS, RFC2866 RADIUS Accounting,
  and RFC3858 DIAMETER
- **Introduction to DIAMETER -** HP Tech Report T1428-90011—
  http://docs.hp.com/en/T1428-90011/
- **Introduction to DIAMETER -** IBM developerWorks Article —
  http://www.ibm.com/developerworks/library/wi-diameter/index.html

Source: http://jfdm.host.cs.st-andrews.ac.uk/notes/netsec/#_security_attacks