

INTERNET DRAFT
Category: Informational
Title: [draft-ietf-aaa-issues-05.txt](#)
Date: January 2002

Pat R. Calhoun (editor)
Sun Microsystems, Inc.
Bernard Aboba
Microsoft
Erik Guttman
Sun Microsystems, Inc.
Dave Mitton
Nortel Networks
Dave Nelson
Enterasys Networks, Inc.
Juergen Schoenwaelder
Technical University Braunschweig
Barney Wolff
Databus Inc.
Lixia Zhang
UCLA

AAA Problem Statements

Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of [Section 10 of RFC2026](#). Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at:

<http://www.ietf.org/ietf/1id-abstracts.txt>

The list of Internet-Draft Shadow Directories can be accessed at:

<http://www.ietf.org/shadow.html>.

This document is an individual contribution for consideration by the AAA Working Group of the Internet Engineering Task Force. Comments should be submitted to the diameter@diameter.org and aaa-wg@merit.edu mailing lists.

INTERNET DRAFT

January 2002

Distribution of this memo is unlimited.

Copyright (C) The Internet Society 2000. All Rights Reserved.

Abstract

The AAA Evaluation Team's recommendation document raised some issues with the DIAMETER protocol. The AAA WG has created a design team to address these issues. This document is an attempt to describe the problems, which the WG can then concentrate on solving.

Table of Contents

- 1.0 Introduction
 - 1.1 Requirements language
- 2.0 Error Codes and Messages
 - 2.1 Error Messages Action Item List
- 3.0 Accounting
 - 3.1 The Universal Approach
 - 3.2 Batch Accounting Issues
 - 3.3 Proxy Accounting Issues
 - 3.4 Semantic Issues
 - 3.5 Accounting Message "Bloat" Issues
 - 3.6 Accounting Message Format Issues
 - 3.7 Accounting Action Item List
- 4.0 IPv6 Support
- 5.0 Transports
 - 5.1 Failover & Recovery Sending
 - 5.1.1 Failover Action Item List
- 6.0 Proxies
 - 6.1 Proxy Behavior
 - 6.2 State/Path Retention
 - 6.3 Proxy Action Item List
- 7.0 RADIUS Compatibility
- 8.0 End-to-End Security
 - 8.1 End-to-End Security Action Item List
- 9.0 Data Model
 - 9.1 Separability of DIAMETER Header and Message
 - 9.2 Data Types supported in DIAMETER
 - 9.3 Formal notation for application specific data types
 - 9.3.1 Goals for formal notation

- 9.3.2 How to evaluate proposals for formal data
type languages
- 9.4 Ordering of Data
- 9.5 Mandatory AVPs
- 9.6 BNF Notation

- 9.7 Data Model Action Item List
- 10.0 SNMP Support (DIAMETER MIB)
 - 10.1 Configuration of Sensitive Parameters
 - 10.2 Modular MIB(s)
 - 10.3 Traps and Informs
 - 10.4 SNMP Action Item List
- 11.0 Re-Authentication & Authorization
- 12.0 Server/Resource Management State
- 13.0 Access Rules and Filters
 - 13.1 Access Rules and Filters Action Item List
- 14.0 AAA Server Discovery
 - 14.1 Server Discovery Action Item List
- 15.0 Loop Detection
 - 15.1 Loop Detection Action Item List
- 16.0 Issues identified in the Mailing List
 - 16.1 DIAMETER Identifier Field
 - 16.2 DIAMETER State Machine
- 17.0 IANA Considerations
- 18.0 Security Considerations
- 19.0 Acknowledgments
- 20.0 References
- 21.0 Authors' Addresses
- 22.0 Full Copyright Statement

INTERNET DRAFT

January 2002

[1.0](#) Introduction

The AAA Evaluation Team's recommendation document raised some issues with the DIAMETER protocol. The AAA WG has created a design team to address these issues. This document is an attempt to describe the problems, which the WG can then concentrate on solving.

[2.0](#) Error Codes and Messages

Given the extensibility nature of the DIAMETER protocol, future extensions will most likely need to allocate error codes to handle error conditions specific to the extension. This extensibility leads to an interoperability problem with implementations that do not support new extensions, since it becomes difficult to recognize how the error should be treated.

The HTTP protocol has solved this problem by creating classes of errors. Implementations only need to understand the error class in order to properly process the error code.

One of the major issues with RADIUS [[11](#)] was the lack of error messages. In RADIUS there was no way for the Accounting server to indicate conditions such as "too busy" or "out of disk space", so the only choice for an Accounting server experiencing difficulties is to not respond to an Accounting-Request, thereby causing re-transmissions that could exacerbate the problem. In addition, since there is no status message equivalent of "OK" or "Processing" in RADIUS there is no way for an accounting server to explicitly

indicate that an accounting record has been written to stable storage. These issues preclude interpretation of a RADIUS Accounting-Response as an application-layer ACK.

In addition to addressing these problems, DIAMETER also needs to define the interaction of error messages and security. For example, allowing error messages from an intermediate node to affect end-to-end communications may enable denial of service attacks.

[2.1](#) Error Messages Action Item List

The action items identified in this section are summarized as follows:

- 1.) Define how the DIAMETER protocol could represent classes of errors.
- 2.) Investigate how intermediate proxies inserting error codes in DIAMETER messages breaks end-to-end security, and find a

Calhoun et al.

expires May 2002

[Page 4]

INTERNET DRAFT

January 2002

solution.

- 3.) Determine whether error messages should be used, or whether error codes are sufficient.
- 4.) Ensure that the error codes defined have the necessary richness required for servers to return specific error conditions (e.g. out of disk space), which should cause the NAS (or downstream proxy) to try another accounting server.

[3.0](#) Accounting

[3.1](#) The Universal Approach

One problem with the current DIAMETER Accounting draft [6] is that accounting is handled in a single protocol extension document. The features of accounting include support for distinct environments, such as roaming, that may not be needed in other applications. It may be useful to consider accounting subsets for each of the supported application areas.

[3.2](#) Batch Accounting Issues

[Section 1.2](#) of the DIAMETER Accounting draft [6] says "Each Accounting-Delivery-Max-Batch / Accounting-Delivery-Max-Delay AVP pair with different values forms one pool of accounting data in the DIAMETER node acting as a client." It seems that some more explicit and restricted form of pool creation may be desirable given the potentially large number of possible different "value pairs" for these AVPs.

It has been suggested by some that FTP is a more appropriate protocol for batch accounting, depending of course, on what the ultimate batch size might be.

[3.3](#) Proxy Accounting Issues

Sections [1.3](#), [1.6](#) and [2.2](#) of the DIAMETER Accounting draft [6] deal with how accounting messages are handled for routing or broker proxy applications. The producer of a co-mingled-destination Accounting-Request record (e.g. a NAS) is required to handle a series of partial Accounting-Answer records from various home Accounting Servers. It would appear to be desirable for proxies to handle this extra work and complexity on behalf of the DIAMETER client. Complexity arises out of partially acknowledged accounting data, potentially requiring the DIAMETER client to distinguish and communicate with individual home Accounting Servers that it would otherwise have no knowledge of.

The issues of multi-party trust, counter-signatures and distributed non-repudiation capabilities are not as clearly explained in the draft as they might be. This is a complex area, and perhaps one in which accounting extensions specific to broker proxy operations might be warranted.

[3.4](#) Semantic Issues

The semantics of a successful Accounting-Answer message need to be further defined. Does this mean that the corresponding accounting data has been reliably written to stable storage (to disk, for example) at the ultimate destination (the home Accounting Server, for example)? Should there be error codes to indicate certain classes of failure, such as "out of disk space"? It might well be that the retry

policy for a "no space" error would be different from that of a transient network outage.

The use of the Accounting-Delivery-Max-Batch AVP should be clarified in the case of proxy chains. Will this AVP be modified by each proxy in the chain? How else would the aggregation of accounting data for multiple destinations be handled? The draft implies that the ultimate destination is uniquely and unambiguously defined by the NAI. Is this always the case?

The use of the Accounting-State AVP and Accounting-Status-Ind AVP should be clarified in the case of proxy chains. Do proxies send these messages both upstream and downstream? Do all DIAMETER peers send them (e.g. Servers to NASes)? Are both AVPS really required?

[3.5](#) Accounting Message "Bloat" Issues

Accounting-Answer messages include the same ADIF-Record AVPs as were contained in the Accounting-Request messages, often with additional signatures. This is clearly to obtain distributed non-repudiation in proxy applications, especially brokering environments. This does lead to SNMP-style response "bloat" however, and it is not needed in all environments.

[3.6](#) Accounting Message Format Issues

The DIAMETER Accounting draft indicates that ADIF [\[15\]](#) is the only supported data format for accounting. There has been some indication that an applicability statement may be required for how ADIF is used within DIAMETER.

[3.7](#) Accounting Action Item List

The action items identified in this section are summarized as follows:

- 1.) Investigate feature-specific subsets of the DIAMETER Accounting protocol, to mirror the DIAMETER Extension drafts [\[2,4,5,6,8\]](#).

- 2.) Evaluate recommendation for "pools" created by distinct Accounting-Delivery-Max-Batch / Accounting-Delivery-Max-Delay AVP pairs.
- 3.) Consider definition of batch size, and time intervals for batch accounting, and the use of other protocols, such as FTP.
- 4.) Reconsider partial packet ACKs of accounting responses via proxy chains.
- 5.) Elaborate on issues of multi-party trust, counter-signatures and distributed non- repudiation capabilities.
- 6.) The semantics of a successful Accounting-Answer message need to be further defined.
- 7.) The use of the Accounting-Delivery-Max-Batch AVP should be clarified in the case of proxy chains.
- 8.) The use of the Accounting-State AVP and Accounting-Status-Ind AVP should be clarified in the case of proxy chains. Are both AVPs really needed?
- 9.) Address the accounting response message "bloat" issue.
- 10.) An applicability statement may be required for how ADIF is used within DIAMETER.

4.0 IPv6 Support

Compatibility with IPv6 requires both the ability for DIAMETER to be transported over IPv6, as well as the ability to carry attributes required to configure IPv6 network access. Since hosts requiring network access may be dual stack, and the AAA server may not know this a-priori, a AAA server may need to return both IPv4 and IPv6 attributes and allow the NAS and host to decide which ones to use.

A first pass at determining the IPv6 attributes required for dialup access is available at: <http://www.ietf.org/internet-drafts/draft-aboba-radius-ipv6-02.txt> In order to come up with a list of IPv6 attributes for DIAMETER, it is expected that a similar review will be needed for other types of network access.

5.0 Transports

AAA protocols typically are application driven, which means that the

time between requests is typically larger than the round-trip time

(RTT). This results in interactions between AAA protocols and reliable transport mechanisms including Nagle algorithm, RTT estimation, delayed ACKs, congestion control, fast re-transmit and fast recovery.

In addition, AAA systems frequently include proxies. As a result, the implementation details of such proxies significantly impact end-to-end behavior. For example, a proxy not implementing "back pressure" can prevent end-to-end self-clocking even if the AAA protocol is running over a transport (such as TCP or SCTP) which supports congestion control.

As a result of these issues, it is necessary to develop a detailed understanding of how DIAMETER will behave when run over transports such as UDP, TCP and SCTP in the presence of proxies.

[5.1](#) Failover & Recovery Sending

SCTP gives an indication of peer failure, but nothing in any DIAMETER or SCTP document the evaluator was able to find even mentions how or when to switch back to a primary server to which communication was lost. After a failure, the state machines end in a CLOSED state and nothing seems to trigger exit from that state. It was not clear whether a server, on rebooting, would initiate an SCTP connection to all its configured clients. If not, and in any case when the communication failure was in the network rather than in the server, the client must itself, after some interval, attempt to re-establish communication. But no such guidance is given.

[5.1.1](#) failover Action Item List

The action items identified in this section are summarized as follows:

- 1.) Add a clear statement of the need for application-level failure detection and failover.
- 2.) Add a clear statement of the need for restart of the transport protocol by one side or the other, to detect recovery of the partner.
- 3.) Evaluate whether the algorithms for 1 & 2 need to be commonly agreed, or can be left to implementors. For example, is a fixed rate of restart attempts (say once per minute) acceptable? Or must there be exponential backoff of this too? Is success of transport protocol establishment sufficient to indicate application availability, or must an application-

- level probe be sent?
- 4.) If the algorithm needs to be universal, design & document it.

[6.0](#) Proxies

[6.1](#) Proxy Behavior

There two known types of servers, redirect and proxies. The DIAMETER specification describes these types of servers as brokers and proxies. The use of the term broker is problematic, since this term is mostly business-driven, and not necessarily technical. The specification should make use of the terminology redirect servers and proxies, and defined the functionality of each type of server.

[6.2](#) State/Path Retention

The DIAMETER specification makes use of two different types of mechanism to determine the path of the messages; Proxy-State and Destination-NAI. The former mechanism is based on the RADIUS protocol, while the latter was added to allow the reverse path the be different from the forwarding path. However, very little interest has been shown for the latter feature.

In the RADIUS protocol, each RADIUS proxy MUST remove any existing Proxy-State attribute, and replace it with a new version. The Proxy-State attribute can contain state information that is generally useful to the sending server. This requires that each proxy maintain the source of the message, in order to forward the response back to the same peer.

The SIP RFC supports this functionality through its Via header field, which allows intermediate nodes to add their information to the message. This effectively builds a "source-route" of the message, and ensures that the reverse path is the same as the forwarding path. A Via-like AVP could also be used to encode state information, which would be used in the reverse (response) message.

There are instances where a server would not wish to disclose the original server (or set of servers) of a message. This server should be allowed to remove the necessary portion of the Via-like AVP, and replace it with its own. This would ensure that it would receive the corresponding response, and add the list of Via-like AVPs that had been truncated.

INTERNET DRAFT

January 2002

The action items identified in this section are summarized as follows:

- 1.) Properly define the terms proxy, broker, redirect server, transparent and non-transparent proxy in the DIAMETER specifications, and how each type of device should function.
- 2.) Investigate whether the current Proxy-State and Destination-NAI AVPs provide the functionality required. Determine whether a SIP-like approach would be more advantageous. The SIP-like approach should allow for state information to also be encoded within the AVP.
- 3.) If a SIP-like approach is used, investigate how end-to-end security is affected by entities adding their local information to messages.

[7.0](#) RADIUS Compatibility

RADIUS compatibility is a requirement for a new AAA protocol, as to be able to transition existing RADIUS environments to the new infrastructure over time. It is clear that there will be features of the new protocol that will not be completely expressible in RADIUS. Those features should not be used in such a configuration, but all attempts should be made to design the protocol so that typical and common RADIUS operations of today can be easily supported by future AAA servers.

Since AAA servers are typically implemented on host systems as software, they are more easily upgraded than NAS systems which often have embedded software constraints. The likely situation in the future is that new AAA server systems will be deployed to provide new enhanced services, but also support legacy NAS servers which will either be eventually upgraded or replaced.

The most desirable compatibility functions would be:

- a) An AAA server that can speak both RADIUS and DIAMETER, using a common system. (not just two separate servers)
- b) An AAA proxy server that can support a number of RADIUS

speaking NASes, but communicates with a DIAMETER server or infrastructure.

There should be a core correspondence between RADIUS messages and DIAMETER operations which is straight forward to implement and covers all typical RADIUS operations. Returned attributes/AVPs that cannot be translated can be discarded if not mandatory. Incompatible mandatory attributes MUST cause the operation to fail with an appropriate error return. Solving this type of problem would have to

be done by reconfiguring the gateway or the user return profile. It may be possible (but not mandatory) that the gateway implements configurable translation rules for vendor or site specific features. It may also be possible that the server gets hints from the gateway about RADIUS translation, and knows not to return problem attributes.

It is NOT a goal to provide the ability to gateway all new DIAMETER functions into RADIUS messages. Particularly features such as new Vendor specific attributes, or security encodings cannot be expected to be supported. Peer-to-peer features have no standard equivalents.

[8.0](#) End-to-End Security

The existing DIAMETER protocol provides security between the NAS (or Mobility Agent) and the Home DIAMETER server through the use of the Cryptographic Message Syntax (CMS) protocol. Typically, this security is used to detect fraudulent DIAMETER proxies, which could attempt to modify critical data within messages (e.g. session lifetime). There has been concern that the use of CMS may be too processor intensive, given that it makes exclusive use of public cryptography.

Although the DIAMETER protocol does mention that in many cases, CMS would not be invoked by the NAS itself, but rather by the first hop DIAMETER, it is possible that this point needs to be strengthened. Furthermore, there is still concern that public key crypto on the servers may also be too intensive, and not always necessary.

One possible solution is to look at the new work underway in the S/MIME group, which is defining support for symmetric key transforms within CMS [\[22\]](#). This would require an additional round trip to exchange the keys, but this would only be done when keys are not

available. Another possible solution is to look at another cryptographic protocol.

[8.1](#) End-to-End Security Action Item List

The action items identified in this section are summarized as follows:

- 1.) Discuss and justify the threat model under which NAS (or remote ISP) to home-server security is any improvement over hop-by-hop.
- 2.) Stop the misuse of "end-to-end" to mean "NAS-to-home-server".
- 3.) Clarify the separate concerns of disclosure of user information and cheating on billable accounting attributes. Which is the real concern? Does one solution work for both?

Calhoun et al.

expires May 2002

[Page 11]

INTERNET DRAFT

January 2002

- 4.) Coordinate work on establishing a shared symmetric key with the server-discovery work.

[9.0](#) Data Model

[9.1](#) Separability of DIAMETER Header and Message

The current DIAMETER protocol specification does not distinguish between the base protocol (with its set of base data types) and the application specific commands and the data structures communicated by the protocol. Experience with other protocols shows that it is valuable to logically separate the application specific data definitions carried by a protocol from the core services provided by a protocol. It is often useful to be able to interpret a message using high-level functional parameters (for instance, in the header), even if the complete message can not be parsed (for example, if a particular AVP code is not supported by an AAA proxy).

To this end, it may be useful to specify in the DIAMETER header whether a particular message is an Request, an Answer, a Query, a Response, or an Indication.

[9.2](#) Data Types supported in DIAMETER

The base data types in the current DIAMETER specification include Data, String, Address, Integer32, Integer64, Time and Complex. It is in general desirable to reduce the base types shipped by a protocol to a small orthogonal set which is sufficient to support the application specific data carried by the protocol, especially since it is difficult to add new base data types in the future.

The Integer32 and Integer64 types are used for signed and sometimes for unsigned AVPs. It may make sense to introduce Unsigned32 and Unsigned64 as base types so avoid any ambiguities.

The Time type is an Unsigned32 number reporting the number of seconds since January 1st 1900. There are several issues with this. First, the introduction of an Unsigned32 time implies that Time with its additional semantics as a base type is not needed anymore since the data definition can just define the special Time semantics on top of Unsigned32. Second, the 32 bit unsigned number will wrap in 2038 and it is desirable to prevent a "year 2038 problem" in DIAMETER, e.g. by starting the epoch at January 1st 2000 or by using the Unsigned64 base type with the January 1st 1900 epoch.

The Data, String and Address types are closely related as they are

all application specific interpretations of strings of octets. So rather than having the specific semantics for addresses or printable strings in the base data types, it seems desirable to collapse these three types into a single OctetString or Data type and to let the (extension specific) AVPs define the special semantics.

There is a RADIUS AVP which uses 32 bit IEEE floating point values. It should be considered whether there is a need to provide support for 32, 64 and 128 bit IEEE floating point numbers. (Note again that the addition of new base types later on is extremely costly.)

The Complex data type should be avoided as it usually requires code changes when it is being used. It is suggested that the 'Grouped' type should replace 'Complex' and that Grouped be composed of a well known sequence of base data types.

It has also been suggested that an additional data type could be added: List. Lists would include items of identical type, whose

length is only known at the time the list is generated or interpreted.

[9.3](#) Formal notation for application specific data types

The use of a formal notation for the definition of (potentially complex) application specific data types has proven to be valuable, especially if a protocol is designed to be extensible. A formal notation enables tools that can (a) verify the correctness of data definitions, (b) automate some of the implementation process, (c) help in debugging scenarios, and (d) enable implementations that can be easily adapted to vendor specific extensions. Furthermore, the separation of the data definitions from the core protocol specification allows extension writers to re-use existing data definitions (e.g. for addressing types) and it thus promotes consistency across a variety of management protocols.

The formal notation will serve the extensibility of AAA implementations in terms of their data storage, interpretation, validation, display, etc. Support for the formal notation is in this sense an implementation option to enhance support ease the integration of extensions, but not required for compliance.

The formal notation is not intended to define the transfer encoding (the on-the-wire byte formatting). The DIAMETER specifications of the base types include these definitions and rules.

[9.3.1](#) Goals for formal notation

The goals for developing a formal notation would be to facilitate the implementation of various functions, such as a data dictionary, a packet filter or an extensible administrative console. Such facilities have been shown to be useful in RADIUS implementations.

The facilities based on formal notation for supported AVP would:

- make it possible to add new AVP storage, type checking, etc. support without changing code.
- make it easier to extend functionality to sniffers, debuggers management tools and DIAMETER implementations (potentially

without adding code).

Non-goals are to define a formal notation which can express any possible (i.e., non-DIAMETER) message. We only need to express DIAMETER messages which have been currently defined and those messages which are likely to be defined, given experience with RADIUS. It is thought that support for extensible functions would be a useful but not mandatory feature for DIAMETER implementations.

[9.3.2](#) How to evaluate proposals for formal data type languages

The 'metrics' by which proposals for a formal notation and additional data types for DIAMETER data will be assessed are:

- Is it possible to specify the existing DIAMETER messages?
- Is the formal specification language tied to standards which are expected to remain stable (that is, not expected to change in the near to medium term)?
- It is clear that using groups or lists of primitive data types will be less efficient than a complex data type (which could include byte by byte data fields, for example, and would not require AVP headers for each element in the data group/ data table. Still, it is reasonable to sacrifice some efficiency for the sake of simplifying protocol implementation and facilitating formalization of data types as described above. If messages are, for example, 20% larger that would be more acceptable than if they became 50% larger. If messages became 100% larger, it would be difficult to accept the alternative to the current "Complex" data type in DIAMETER.

[9.4](#) Ordering of Data

The current DIAMETER specification says that AVPs can be added arbitrarily as long as the required AVPs are included and AVPs which are explicitly excluded are not included. Some specific AVPs however

introduce special requirements and even dependencies between AVPs. (The Session-Id AVPs SHOULD appear first, Integrity-Check-Value AVPs must follow the data to be protected, the Nonce AVP must appear before an Integrity-Check-Value AVP, the last Nonce AVP before an

Encrypted-Payload AVP is significant for MD5 payload hiding and so on.)

It seems useful to formally define the ordering constraints, potentially restricting the "AVPs can be added arbitrarily" rule to simplify the validation of messages. Note that it is necessary to provide for optional AVPs within a message so that the optional AVPs can be protected by an Integrity-Check-Value AVP.

[9.5](#) Mandatory AVPs

The current DIAMETER protocol specification defines a mandatory bit ('M') in the AVP header. It basically requires that receivers of messages discard any DIAMETER messages which contain AVPs with the text is silent how a sender of a DIAMETER message decides which AVPs are tagged as mandatory.

There seem to be different opinions about the purpose and the precise usage of the 'M' bit. It is necessary to figure out what the purpose of the 'M' is and whether it is a static or dynamic property of an AVP. If it is static for e.g. each AVP of a certain type in a given command, then the question arises whether the 'M' must be carried in the protocol at all.

[9.6](#) BNF Notation

The DIAMETER documents frequently use a BNF style notation to describe message formats. It needs to be clarified whether the BNF has any semantic relevance. If yes, the document needs to say somewhere precisely what the notation means or to replace it with some other mechanism. It should be noted that it is possible to make the BNF notation more powerful by indicating multiplicity constraints in the BNF itself, rather than having it plain text.

[9.7](#) Data Model Action Item List

The action items identified in this section are summarized as follows:

- 1.) How could the DIAMETER message header be made to easily distinguish between Request, Answer, Query, Response and

Indication messages?

- 2.) What basic data types specifically should be included? We would need to look at all RADIUS extensions to see if we've missed any types.
- 3.) How would List and Grouped data types be transmitted over the wire?
- 4.) What formal notation could be used for DIAMETER messages? Assess these according to the goals and metrics described in [Section 9](#).

[10.0](#) SNMP Support (DIAMETER MIB)

[10.1](#) Configuration of Sensitive Parameters

A decision needs to be reached as to whether the DIAMETER MIB(s) should include "sensitive" parameters. Sensitive parameters include shared secrets, certificates, keys, security policies, and any other information that would facilitate an attack on any DIAMETER peer. The first step would be to create a list of all DIAMETER management parameters. The DIAMETER draft [\[1\]](#) already includes much of this information. These parameters should then be classified by level of sensitivity.

It may be possible to take advantage of the security features of SNMPv3 [\[ref\]](#) to allow for complete remote configuration of DIAMETER peers (clients, servers proxies). The level of security available within SNMPv3 should be evaluated against the level of sensitivity of the DIAMETER management parameters.

It has been pointed out that security may be compromised when key material from one protocol is carried in a second protocol, especially when the second protocol has weaker cryptographic protections or is not a key distribution protocol that has been thoroughly analyzed in the published cryptographic literature so that there is high confidence that it is sound. Many in the security community consider it strongly undesirable to have cascading vulnerabilities, whereby a compromise or implementation problem in one protocol necessarily leads to a compromise of a second protocol.

The RADIUS MIB [\[17,18,19,20\]](#) was written when SNMPv1 and SNMPv2C were all we had, and thus avoided any sensitive configuration parameters that would have posed a security risk for remote access. There are interesting problems to be solved to obtain remote plug-'n-play configuration of DIAMETER peers using SNMP and/or some other suitable protocol, such as IKE [\[16\]](#) or Kerberos [\[21\]](#), but this is a desirable design goal.

INTERNET DRAFT

January 2002

[10.2](#) Modular MIB(s)

DIAMETER should probably have a series of MIBs covering the base protocol in its various roles: client, server and proxy, as well as each of its functional extensions. Counters, such as provided in the RADIUS MIB, should be included, as well as metrics and policy information that might aid in remote problem resolution of specific areas, such as proxy operations.

The list of DIAMETER management parameters should also be categorized by the peer role(s) to which each parameter applies. The goal is to permit modular MIB deployment description to match the modular feature deployment. As an alternative, a single MIB could be written, together with Module Compliance Statements, indicating which subset of objects must be implemented in a given peer, by optional function and by optional role.

[10.3](#) Traps and Informs

A list of significant events occurring within DIAMETER peers needs to be compiled. From this list of events, recommendations for Traps and/or Informs for use within the DIAMETER MIB(s) should be presented.

[10.4](#) SNMP Action Item List

The action items identified in this section are summarized as follows:

- 1.) List all DIAMETER peer managed objects.
- 2.) Categorize the list of managed objects by optional role and optional function.
- 3.) Categorize the list of managed objects by security sensitivity.
- 4.) Analyze SNMPv3, IKE, Kerberos and any other suitable protocols as to applicability, from a security perspective, for carrying the managed objects that are deemed sensitive.
- 5.) Analyze the proposed MIB for remote plug-'n-play configuration

- capabilities.
- 6.) Investigate both a single MIB with appropriate (conditional) Module Compliance Statements, as well as multiple MIBs, for modularity mirroring the DIAMETER Extension drafts.
 - 7.) Compile a list of significant DIAMETER peer events.
 - 8.) Recommend Traps and Informs for inclusion in the MIB(s).

[11.0](#) Re-Authentication & Authorization

The AAA protocol must be able to support dynamic re-Authentication [[1,3](#)] initiated by either the NAS or the Server.

Typical uses are periodic CHAP re-authentication, or server re-authentication probes.

There should be a clear description of the messages and commands used to initiate a re-authentication from the NAS or server end.

What is there:

- Mentioned in Base in the [section 1.0](#) wrt to unsolicited messages.
- Mentioned in NASreq in the context of EAP auth.

What is missing:

- How such an unsolicited message would be recognized as a re-auth by the server vs a new request
- What message for a server to use to request re-auth and how a NAS correlates that with an existing session
- What information is allowable in each message (AVP list)

The AAA protocol must be able to support dynamic re-Authorization [[2,4](#)] initiated by either the NAS or the Server.

Typical uses are the changing of service parameters (e.g. Packet filters or Session expiration timers).

There should be a clear description of the messages and commands used to initiate a re-authentication from the NAS or server end.

What is there: anything??

What is missing:

- How would a NAS request a re-authorization?
- How would a Server indicate a re-authorization? Specify messages, commands, AVP datum
- How are these messages distinguished from new requests, and correlated with existing sessions?

Note that a related but different topic, termination of session service is already covered by the Base document, [section 3.4](#), and three messages are described there:

- Session-Termination-Request (STR) NAS->Server,
- Session-Termination-Indication (STI) Server->NAS, and
- Session-Termination-Answer(STA) acknowledgement

[12.0](#) Server/Resource Management State

Some implementations of RADIUS supported explicit resource management between the NAS and a server for authorized and active sessions [[12](#)]. They implemented a simple resource request/response protocol, where the NAS requested and freed resources managed by the server, with distinct messages. Typical resources managed this way would be IP addresses, concurrent login counts, tunnel session limits, or resource usage based authorization limits.

That this happens after authentication, allows the NAS to only request the resources actually needed to provide the service. Whereas authorization parameters in an Access-Accept may anticipate resources not actually needed (like additional addresses for a multi-link sessions, where the multi-link status could not be determined until after authentication). Likewise, some network requirements cannot be determined until PPP NCP negotiation.

The DIAMETER protocol should be able to provide similar functions [[13](#),[14](#)]. An extensible set of resources should be able to be allocated and deallocated by the NAS, upon request by the server. The server then serves as a management point of the resource for all clients of the server. This service would be optional to the implementation.

Additional capability should be provided, where the server or client

may interrogate identified resource's status, and the server can revoke or modify the resource status [14]. This is similar to the ability to re-authorize NAS services from the Server.

This type of service MAY be robust in nature, in that the server may use a permanent store, or a state distribution protocol to synchronize backup servers between outages. The reliability of such a service will be implementation and configuration dependent. The design of high availability state protocol is beyond the scope of what we wish to attempt at this time. Only an interface would be specified.

Such a service must scale reasonably in the presence of proxy systems. Broadcast or wildcard queries will not be allowed. It is highly likely that resources may be managed in different locations. For example, IP addresses may be managed locally, while user login concurrency checking could be done remotely.

Maintenance of a consistent image of resources in use is a genuinely hard problem in the face of server, NAS and network failures and PDU reordering. The protocol support should give clear guidance to implementors of the known implementation techniques, such as upstream

querying by a recovering server, periodic reconfirm (aka interim accting) by the NAS, & inter-server replication. And on the danger of self-inflicted DoS if the model is not somehow self-correcting.

13.0 Access Rules and Filters

The AAA protocol must support powerful semantics for setting constraints on what an authenticated user may do. That is, it should lean toward being able to invoke any capability that any NAS can provide in this area, rather than restrict itself to being able to express things that every NAS can provide - because the latter is completely inadequate to protect the Internet against malicious users.

Examples of specific expressive power that must be provided:

- source address assurance
- prevention of IP options
- prevention of specific IP options, such as source routing

- prevention of malicious fragmentation, such as offset=1
- restriction of SMTP connections to the home server only and so on.

This implies of course that the action of the NAS when it cannot support a particular filter capability must be specified. (Here 'NAS' might really be a proxy that translates the AAA syntax to vendor-specific syntax for the particular NAS.)

[13.1](#) Access Rules and Filters Action Item List

The action items identified in this section are summarized as follows:

- 1.) Design a syntax for access rules and filters that has sufficient expressive power use in current and clearly foreseeable networks. Extensibility and ability to incorporate vendor-specific capabilities are highly desirable.
- 2.) Specify NAS behavior when it is unable to implement a particular rule.

[14.0](#) AAA Server Discovery

In some cases, AAA servers may be dynamically discovered. This allows for more robust deployments of network access services, for example. If a primary AAA server fails, for which an AAA client is configured, the AAA client can discover another suitable AAA server.

In order to promote interoperable implementations, a definite 'search order' should be recommended, if dynamic service discovery is employed. For example, the search order could be:

1. Use static (manual) configuration if it exists.
2. Use DNS SRV RR for DIAMETER services, if it exists.
3. Use SLP to discover administratively DIAMETER services.

Dynamic discovery of AAA servers will only be secure if previous configuration provided AAA clients with the security parameters which can be used to authenticate the discovered AAA server. This is possible, for instance, using DNS security, SLPv2 authentication, etc.

In some respects configuring hosts to enable them to do dynamic configuration seems at cross purposes. Here the goal is not zero administration but rather robust deployment (so clients can automatically find servers, for instance, in the case that a server fails or is swapped out).

[14.1](#) Server Discovery Action Item List

The action items identified in this section are summarized as follows:

- 1.) Specific worked out solutions for how to discover AAA servers both locally (in the same administrative domain) or across the internet.
- 2.) For each of the above solutions, describe how the mechanism could be made secure, so that AAA clients could verify that the AAA server they discover is 'trustworthy' according to some policy. What preconfiguration is needed? What policy (with respect to trust) is implied by possession of a key?

[15.0](#) Loop Detection

In current RADIUS networks, it is possible for an improperly configured domain routing table to cause messages to be stuck in a loop. The DIAMETER protocol does not provide any loop detection, and therefore is subjected to the same problem.

The Mobile IP extension does have a requirement that some messages destined for the home DIAMETER server be sent back to the visited DIAMETER server for Home Agent allocation. Therefore, the loop detection must be designed to in such a way that the Mobile IP extension can still be used as specified.

[15.1](#) Loop Detection Action Item List

The action items identified in this section are summarized as follows:

- 1.) Determine how the protocol can be enhanced to detect loops, without breaking end-to-end security.
- 2.) Ensure that the mechanism does allow for certain messages to loop back to the requestor, when it is intended to do so.

[16.0](#) Issues identified in the Mailing List

The following issues have been raised in the AAA Mailing list, and are included in this document for completeness.

[16.1](#) DIAMETER Identifier Field

The DIAMETER base protocol does not properly describe the function of the identifier field in the protocol header. The original intent of the field was to facilitate the originator of a request's task of matching up the corresponding response. This field was NOT intended to be mutable (meaning it cannot be changed by intermediate proxies).

The issue at hand is whether this field actually provides useful functionality in the protocol, or whether it should be removed.

[16.2](#) DIAMETER State Machine

A bug in the DIAMETER state machine was raised on the mailing list. The state machine states that once a transport level connection is established, and a DRI is received, the state should move to the open state. This should be changed to reflect a new state called the Wait-DRI state. The connection stays in this state until the peer's DRI is received.

[17.0](#) IANA Considerations

DIAMETER makes extensive use of IDs (command codes, extensions, result codes, AVP attributes, Integrity-Check-Value AVP Transform code). These are collected in the base protocol specification, but defined in the DIAMETER extension docs.

[18.0](#) Security Considerations

DIAMETER [1] is a framework providing authentication and authorization services for network access. [Section 11](#) and 13 concern how these features could be refined or improved in subsequent work.

DIAMETER itself contains a number of security features. [Section 8](#) discusses how these could be redesigned for less reliance on public key cryptography.

[19.0](#) Acknowledgments

The authors would like to thanks Ran Atkinson and William Bulley for their valuable comments.

[20.0](#) References

- [1] P. Calhoun, A. Rubens, H. Akhtar, E. Guttman. "DIAMETER Base Protocol", [draft-calhoun-diameter-17.txt](#), IETF work in progress, September 2000.
- [2] P. Calhoun, W. Bulley, A. Rubens, J. Haag, "DIAMETER NASREQ Extension", [draft-calhoun-diameter-nasreq-05.txt](#), IETF work in progress, September 2000.
- [3] Calhoun, Zorn, Pan, Akhtar, "DIAMETER Framework", [draft-calhoun-diameter-framework-08.txt](#), IETF work in progress, June 2000.
- [4] P. Calhoun, C. Perkins, "DIAMETER Mobile IP Extensions", [draft-calhoun-diameter-mobileip-11.txt](#), IETF work in progress, September 2000.
- [5] P. Calhoun, W. Bulley, S. Farrell, "DIAMETER Strong Security Extension", [draft-calhoun-diameter-strong-crypto-05.txt](#) (work in progress), September 2000.
- [6] Arkko, Calhoun, Patel, Zorn, "DIAMETER Accounting Extension", [draft-calhoun-diameter-accounting-08.txt](#), IETF work in progress, September 2000.
- [7] P. Calhoun, A. Rubens, H. Akhtar, E. Guttman, W. Bulley, J. Haag, "DIAMETER Implementation Guidelines", [draft-calhoun-diameter-impl-guide-03.txt](#), IETF work in progress, June 2000.
- [8] P. Calhoun, N. Greene, "DIAMETER Resource Management", [draft-calhoun-diameter-res-mgmt-05.txt](#), IETF Work in Progress,

INTERNET DRAFT

January 2002

September 2000.

- [9] Aboba et al, "Network Access AAA Evaluation Criteria", IETF work in progress, [draft-ietf-aaa-na-reqts-07.txt](#), August 2000.
- [10] Mitton et al, "Authentication, Authorization, and Accounting: Protocol Evaluation", IETF work in progress, [draft-ietf-aaa-proto-eval-00.txt](#), July 2000.
- [11] Rigney, et alia, "RADIUS", [RFC-2138](#), April 1997
- [12] [RFC 2882](#) "NASReq Extended RADIUS Practices", Sect 6, page 9-10
- [13] [draft-ietf-nasreq-criteria-05.txt](#), Sect 8.3.1.2 page 10
- [14] [draft-ietf-aaa-na-reqts-07.txt](#), Sect 4.4.3.3 Authorization Requirements, State Reconciliation, Note f
- [15] B. Aboba, D. Lidyard, "The Accounting Data Interchange Format (ADIF)", [draft-ietf-roamops-actng-07.txt](#), IETF work in progress, April 2000.
- [16] D. Harkins, D. Carrell, "The Internet Key Exchange (IKE)", [RFC 1409](#), November 1998.
- [17] B. Aboba, G. Zorn, "RADIUS Authentication Client MIB", [RFC 2618](#), June 1999.
- [18] G. Zorn, B. Aboba, "RADIUS Authentication Server MIB", [RFC 2619](#), June 1999.
- [19] B. Aboba, G. Zorn, "RADIUS Accounting Client MIB", [RFC 2620](#), June 1999.
- [20] G. Zorn, B. Aboba, "RADIUS Accounting Server MIB", [RFC 2621](#), June 1999.
- [21] J. Kohl, C. Neuman, "The Kerberos Network Authentication Service (V5)", [RFC 1510](#), September 1993.
- [22] S. Farrell, S. Turner, "Reuse of CMS Content Encryption Keys", [draft-ietf-smime-rcek-00.txt](#), IETF work in progress, September

2000.

[21.0](#) Authors' Addresses

Questions about this memo can be directed to:

Calhoun et al.

expires May 2002

[Page 24]

INTERNET DRAFT

January 2002

Pat R. Calhoun
Network and Security Research Center, Sun Labs
Sun Microsystems, Inc.
15 Network Circle
Menlo Park, California, 94025
USA

Phone: +1 650-786-7733
Fax: +1 650-786-6445
E-mail: pcalhoun@eng.sun.com

Bernard Aboba
Microsoft Corporation
One Microsoft Way
Redmond, WA 98052
USA

Phone: +1 425 936-6605
Fax: +1 425 936-7329
E-mail: bernarda@microsoft.com

Erik Guttman
Network and Security Research Center, Sun Laboratories
Sun Microsystems, Inc.
Eichhoelzelstr. 7
74915 Waibstadt
Germany

Phone: +49-7263-911-701
E-mail: erik.guttman@germany.sun.com

David Mitton

Nortel Networks
880 Technology Park Drive
Billerica, MA 01821
USA

Phone: +1 978 288 4570
E-mail: dmitton@nortelnetworks.com

David B. Nelson
Enterasys Networks, Inc. (a Cabletron Systems company)
50 Minuteman Road
Andover, MA 01810-1008

Calhoun et al.

expires May 2002

[Page 25]

INTERNET DRAFT

January 2002

USA

Phone: +1 978 684 1330
E-Mail: dnelson@enterasys.com

Juergen Schoenwaelder
Technical University Braunschweig
Dept. Operating Systems & Computer Networks
Bueltenweg 74/75, 38106 Braunschweig,
Germany

Phone: +49 531 391 3289
Fax: +49 531 391 5936
E-Mail: schoenw@ibr.cs.tu-bs.de

Barney Wolff, Pres.
Databus Inc.
15 Victor Drive
Irvington, NY 10533-1919 USA
USA

Phone: +1 914 591 5677
E-mail: barney@databus.com

Lixia Zhang

UCLA Computer Science Department
4531G Boelter Hall
Los Angeles, CA 90095-1596
USA

Phone: +1 310 825 2695
E-Mail: lixia@cs.ucla.edu

[22.0](#) Full Copyright Statement

Copyright (C) The Internet Society (2000). All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the

Calhoun et al.

expires May 2002

[Page 26]

INTERNET DRAFT

January 2002

copyright notice or references to the Internet Society or other Internet organizations, except as needed for the purpose of developing Internet standards in which case the procedures for copyrights defined in the Internet Standards process must be followed, or as required to translate it into languages other than English. The limited permissions granted above are perpetual and will not be revoked by the Internet Society or its successors or assigns. This document and the information contained herein is provided on an "AS IS" basis and THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

