Network Working Group
Request for Comments: 2607
Category: Informational

B. Aboba
Microsoft Corporation
J. Vollbrecht
Merit Networks, Inc.

June 1999

Proxy Chaining and Policy Implementation in Roaming

Status of this Memo

Copyright Notice

1. Abstract

This document describes how proxy chaining and policy implementation can be supported in roaming systems. The mechanisms described in this document are in current use.

However, as noted in the security considerations section, the techniques outlined in this document are vulnerable to attack from external parties as well as susceptible to fraud perpetrated by the roaming partners themselves. As a result, such methods are not suitable for wide-scale deployment on the Internet.

2. Terminology

This document frequently uses the following terms:

Network Access Server
: The Network Access Server (NAS) is the device that clients contact in order to get access to the network.

RADIUS server
: This is a server which provides for authentication/authorization via the protocol described in [3], and for accounting as described in [4].

RADIUS proxy
In order to provide for the routing of RADIUS authentication and accounting requests, a RADIUS proxy can be employed. To the NAS, the RADIUS proxy appears to act as a RADIUS server, and to the RADIUS server, the proxy appears to act as a RADIUS client.

Network Access Identifier
In order to provide for the routing of RADIUS authentication and accounting requests, the userID field used in PPP (known as the Network Access Identifier or NAI) and in the subsequent RADIUS authentication and accounting requests, can contain structure. This structure provides a means by which the RADIUS proxy will locate the RADIUS server that is to receive the request. The NAI is defined in [6].

Roaming relationships
Roaming relationships include relationships between companies and ISPs, relationships among peer ISPs within a roaming association, and relationships between an ISP and a roaming consortia. Together, the set of relationships forming a path between a local ISP’s authentication proxy and the home authentication server is known as the roaming relationship path.

3. Requirements language

In this document, the key words "MAY", "MUST, "MUST NOT", "optional", "recommended", "SHOULD", and "SHOULD NOT", are to be interpreted as described in [5].

4. Introduction

Today, as described in [1], proxy chaining is widely deployed for the purposes of providing roaming services. In such systems, authentication/authorization and accounting packets are routed between a NAS device and a home server through a series of proxies. Consultation of the home server is required for password-based authentication, since the home server maintains the password database and thus it is necessary for the NAS to communicate with the home authentication server in order to verify the user’s identity.

4.1. Advantages of proxy chaining

Proxies serve a number of functions in roaming, including:

Scalability improvement
Authentication forwarding
Capabilities adjustment
Policy implementation
Accounting reliability improvement
Atomic operation

Scalability improvement
In large scale roaming systems, it is necessary to provide for scalable management of keys used for integrity protection and authentication.

Proxy chaining enables implementation of hierarchical forwarding within roaming systems, which improves scalability in roaming consortia based on authentication protocols without automated key management. Since RADIUS as described in [3] requires a shared secret for each client-server pair, a consortium of 100 roaming partners would require 4950 shared secrets if each partner were to contact each other directly, one for each partner pair. However, were the partners to route authentication requests through a central proxy, only 100 shared secrets would be needed, one for each partner. The reduction in the number of partner pairs also brings with it other benefits, such as a reduction in the number of bilateral agreements and accounting and auditing overhead. Thus, hierarchical routing might be desirable even if an authentiation protocol supporting automated key exchange were available.

Capabilities adjustment
As part of the authentication exchange with the home server, the NAS receives authorization parameters describing the service to be provided to the roaming user. Since RADIUS, described in [3], does not support capabilities negotiation, it is possible that the authorization parameters sent by the home server will not match those required by the NAS. For example, a static IP address could be specified that would not be routable by the NAS. As a result, capbilities adjustment is performed by proxies in order to enable communication between NASes and home servers with very different feature sets.

As part of capabilities adjustment, proxies can edit attributes within the Access-Accept in order to ensure compatibility with the NAS. Such editing may include addition, deletion, or modification of attributes. In addition, in some cases it may be desirable for a proxy to edit attributes within an Access-Request in order to clean up or even hide information destined for the home server. Note that if the proxy edits attributes within the Access-Accept, then it is possible that the service provided to the user may not be the same as that requested by the home server. This creates the possibility of disputes arising from inappropriate capabilities adjustment.

Note that were roaming to be implemented based on an authentication/authorization protocol with built-in capability negotiation, proxy-based capabilities adjustment would probably not be necessary.

Authentication forwarding

Since roaming associations frequently implement hierarchical forwarding in order to improve scalability, in order for a NAS and home server to communicate, authentication and accounting packets are forwarded by one or more proxies. The path travelled by these packets, known as the roaming relationship path, is determined from the Network Access Identifier (NAI), described in [6]. Since most NAS devices do not implement forwarding logic, a proxy is needed to enable forwarding of authentication and accounting packets. For reasons that are described in the security section, in proxy systems it is desirable for accounting and authentication packets to follow the same path.

Note: The way a proxy learns the mapping between NAI and the home server is beyond the scope of this document. This mapping can be accomplished by static configuration in the proxy, or by some currently undefined protocol that provides for dynamic mapping. For the purposes of this document, it is assumed that such a mapping capability exists in the proxy.

Policy implementation

In roaming systems it is often desirable to be able to implement policy. For example, a given partner may only be entitled to use of a given NAS during certain times of the day. In order to implement such policies, proxies may be implemented at the interface between administrative domains and programmed to modify authentication/authorization packets forwarded between the NAS and the home server. As a result, from a security point of view, a proxy implementing policy

operates as a "man in the middle."

Accounting reliability improvement
In roaming systems based on proxy chaining, it is necessary for accounting information to be forwarded between the NAS and the home server. Thus roaming is inherently an interdomain application.

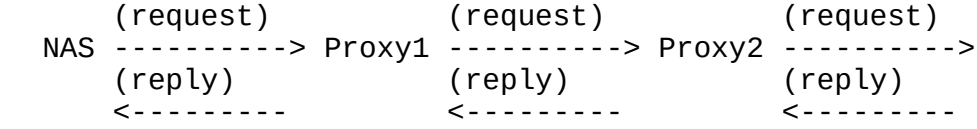
This represents a problem since the RADIUS accounting protocol, described in [4] is not designed for use on an Internet scale. Given that in roaming accounting packets travel between administrative domains, packets will often pass through network access points (NAPs) where packet loss may be substantial. This can result in unacceptable rates of accounting data loss.

For example, in a proxy chaining system involving four systems, a one percent failure rate on each hop can result in loss of 3.9 percent of all accounting transactions. Placement of an accounting proxy near the NAS may improve reliability by enabling enabling persistent storage of accounting records and long duration retry.

Atomic operation
In order to ensure consistency among all parties required to process accounting data, it can be desirable to assure that transmission of accounting data is handled as an atomic operation. This implies that all parties on the roaming relationship path will receive and acknowledge the receipt of the accounting data for the operation to complete. Proxies can be used to ensure atomic delivery of accounting data by arranging for delivery of the accounting data in a serial fashion, as discussed in section 5.2.

5. Proxy chaining

An example of a proxy chaining system is shown below.



In the above diagram, the NAS generates a request and sends it to Proxy1. Proxy1 forwards the request to Proxy2 and Proxy2 forwards the request to the Home Server. The Home Server generates a reply and sends it to Proxy2. Proxy2 receives the reply, matches it with the request it had sent, and forwards a reply to Proxy1. Proxy1

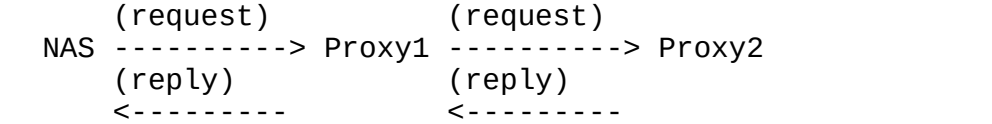
matches the reply with the request it sent earlier and forwards a reply to the NAS. This model applies to all requests, including Access Requests and Accounting Requests.

Except for the two cases described below, a proxy server such as Proxy2 in the diagram above SHOULD NOT send a Reply packet to Proxy1 without first having received a Reply packet initiated by the Home Server. The two exceptions are when the proxy is enforcing policy as described in section 5.1 and when the proxy is acting as an accounting store (as in store and forward), as described in section 5.2.

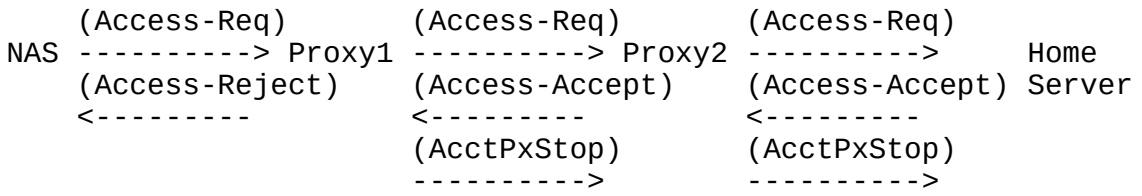
The RADIUS protocol described in [3] does not provide for end-to-end security services, including integrity or replay protection, authentication or confidentiality. As noted in the security considerations section, this omission results in several security problems within proxy chaining systems.

5.1. Policy implementation

Proxies are frequently used to implement policy in roaming situations. Proxies implementing policy MAY reply directly to Access-Requests without forwarding the request. When replying directly to an Access-Request, the proxy MUST reply either with an Access-Reject or an Access-Challenge packet. A proxy MUST NOT reply directly with an Access-Accept. An example of this would be when the proxy refuses all connections from a particular realm during prime time. In this case the home server will never receive th Access-Request. This situation is shown below:



A proxy MAY also decide to Reject a Request that has been accepted by the home server. This could be based on the set of attributes returned by the home server. In this case the Proxy SHOULD send an Access-Reject to the NAS and an Accounting-Request with Acct-Status-Type=Proxy-Stop (6) to the home server. This lets the home server know that the session it approved has been denied downstream by the proxy. However, a proxy MUST NOT send an Access-Accept after receiving an Access-Reject from a proxy or from the home server.



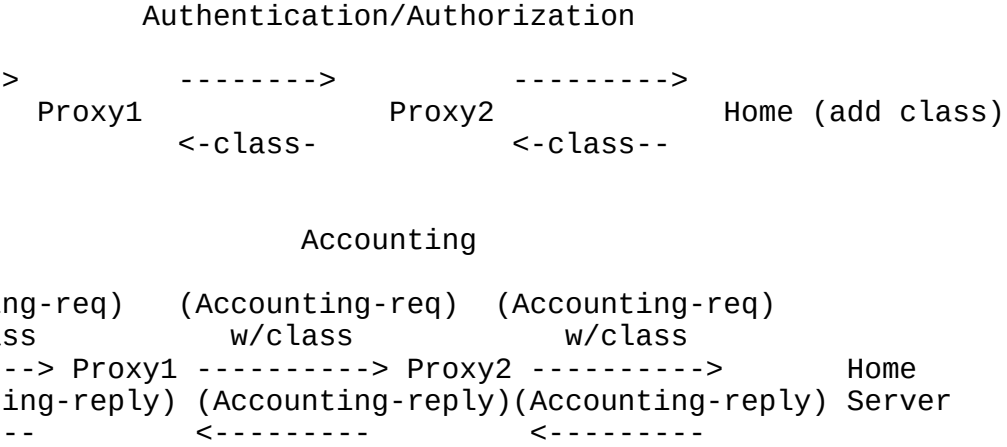
5.2. Accounting behavior

As described above, a proxy MUST NOT reply directly with an Access-Accept, and MUST NOT reply with an Access-Accept when it has received an Access-Reject from another proxy or Home Server. As a result, in all cases where an accounting record is to be generated (accepted sessions), no direct replies have occurred, and the Access-Request and Access-Accept have passed through the same set of systems.

In order to allow proxies to match incoming Accounting-Requests with previously handled Access-Requests and Access-Accepts, a proxy SHOULD route the Accounting-Request along the same realm path travelled in authentication/authorization. Note that this does not imply that accounting packets will necessarily travel the identical path, machine by machine, as did authentication/authorization packets. This is because it is conceivable that a proxy may have gone down, and as a result the Accounting-request may need to be forwarded to an alternate server. It is also conceivable that authentication/authorization and accounting may be handled by different servers within a realm.

The Class attribute can be used to match Accounting Requests with prior Access Requests. It can also be used to match session log records between the home Server, proxies, and NAS. This matching can be accomplished either in real-time (in the case that authentication and accounting packets follow the same path, machine by machine), or after the fact.

Home servers SHOULD insert a unique session identifier in the Class attribute in an Access-Accept and Access-Challenge. Proxies and NASes MUST forward the unmodified Class attribute. The NAS MUST include the Class attribute in subsequent requests, in particular for Accounting-Requests. The sequence of events is shown below:



Since there is no need to implement policy in accounting, a proxy MUST forward all Accounting Requests to the next server on the path. The proxy MUST guarantee that the Accounting Request is received by the End Server and all intermediate servers. The proxy may do this either by: 1) forwarding the Accounting Request and not sending a Reply until it receives the matching Reply from the upstream server, or 2) acting as a store point which takes responsibility for reforwarding the Accounting Request until it receives a Reply.

Note that when the proxy does not send a reply until it receives a matching reply, this ensures that Accounting Start and Stop messages are received and can be logged by all servers along the roaming relationship path. If one of the servers is not available, then the operation will fail. As a result the entire accounting transaction will either succeed or fail as a unit, and thus can be said to be atomic.

Where store and forward is implemented, it is possible that one or more servers along the roaming relationship path will not receive the accounting data while others will. The accounting operation will not succeed or fail as a unit, and is therefore not atomic. As a result, it may not be possible for the roaming partners to reconcile their audit logs, opening new opportunities for fraud. Where store and forward is implemented, forwarding of Accounting Requests SHOULD be done as they are received so the downstream servers will receive them in a timely way.

Note that there are cases where a proxy will need to forward an Accounting packet to more than one system. For example, in order to allow for proper accounting in the case of a NAS that is shutting down, the proxy can send an Accounting-Request with Acct-Status-Type=Accounting-Off (8) to all realms that it forwards to. In turn, these proxies will also flood the packet to their connected realms.

6. References

[1] Aboba, B., Lu J., Alsop J., Ding J. and W. Wang, "Review of Roaming Implementations", RFC 2194, September 1997.

[2] Aboba, B. and G. Zorn, "Criteria for Evaluating Roaming Protocols", RFC 2477, January 1999.

[3] Rigney, C., Rubens, A., Simpson, W. and S. Willens, "Remote Authentication Dial In User Service (RADIUS)", RFC 2138, April 1997.

[4] Rigney, C., "RADIUS Accounting", RFC 2139, April 1997.

[5] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[6] Aboba, B. and M. Beadles, "The Network Access Identifier", RFC 2486, January 1999.

7. Security Considerations

The RADIUS protocol described in [3] was designed for intra-domain use, where the NAS, proxy, and home server exist within a single administrative domain, and proxies may be considered a trusted component. However, in roaming the NAS, proxies, and home server will typically be managed by different administrative entities. As a result, roaming is inherently an inter-domain application, and proxies cannot necessarily be trusted. This results in a number of security threats, including:

Message editing
Attribute editing
Theft of passwords
Theft and modification of accounting data
Replay attacks
Connection hijacking
Fraudulent accounting

7.1. Message editing

Through the use of shared secrets it is possible for proxies operating in different domains to establish a trust relationship. However, if only hop-by-hop security is available then untrusted proxies are capable of perpetrating a number of man-in-the-middle attacks. These include modification of messages.

For example, an Access-Accept could be substituted for an Access-Reject, and without end-to-end integrity protection, there is no way for the NAS to detect this. On the home server, this will result in an accounting log entry for a session that was not authorized. However, if the proxy does not forward accounting packets or session records to the home server, then the home server will not be able to detect the discrepancy until a bill is received and audited.

Note that a proxy can also send an Access-Reject to the NAS after receiving an Access-Accept from the home server. This will result in an authentication log entry without a corresponding accounting log entry. Without the proxy sending an Accounting-Request with Acct-Status-Type=Proxy-Stop (6) to the home server, then there will be no way for the home server to determine whether the discrepancy is due to policy implementation or loss of accounting packets. Thus the use of Acct-Status-Type=Proxy-Stop can be of value in debugging roaming systems.

It should be noted that even if end-to-end security were to be available, a number of sticky questions would remain. While the end-points would be able to detect that the message from the home server had been modified by an intermediary, the question arises as to what action should be taken. While the modified packet could be silently discarded, this could affect the ability of the home server to . accept an Acct-Status-Type=Proxy-Stop message from an intermediate proxy. Since this message would not be signed by the NAS, it may need to be dropped by the home server.

This is similar to the problem that IPSEC-capable systems face in making use of ICMP messages from systems with whom they do not have a security association. The problem is more difficult here, since in RADIUS retransmission is driven by the NAS. Therefore the home server does not receive acknowledgement for Access-Accepts and thus would have no way of knowing that its response has not been honored.

7.2. Attribute editing

RADIUS as defined in [3] does not provide for end-to-end security or capabilities negotiation. As a result there is no way for a home server to securely negotiate a mutually acceptable configuration with the NAS or proxies. As a result, a number of attribute editing attacks are possible.

For example, EAP attributes might be removed or modified so as to cause a client to authenticate with EAP MD5 or PAP, instead of a stronger authentication method. Alternatively, tunnel attributes might be removed or modified so as to remove encryption, redirect the tunnel to a rogue tunnel server, or otherwise lessen the security

provided to the client. The mismatch between requested and received services may only be detectable after the fact by comparing the Access-Accept attributes against the attributes included in the Accounting-Request. However, without end-to-end security services, it is possible for a rogue proxy to cover its tracks.

Due to the complexity of proxy configuration, such attacks need not involve malice, but can occur due to mis-configuration or implementation deficiencies. Today several proxy implementations remove attributes that they do not understand, or can be set up to replace attribute sets sent in the Access-Accept with sets of attributes appropriate for a particular NAS.

In practice, it is not possible to define a set of guidelines for attribute editing, since the requirements are very often implementation-specific. At the same time, protection against inappropriate attribute editing is necessary to guard against attacks and provide assurance that users are provisioned as directed by the home server.

Since it is not possible to determine beforehand whether a given attribute is editable or not, a mechanism needs to be provided to allow senders to indicate which attributes are editable and which are not, and for the receivers to detect modifications of "non-editable" attributes. Through implementation of end-to-end security it may be possible to detect unauthorized addition, deletion, or modification of integrity-protected attributes. However, it would still possible for a rogue proxy to add, delete or modify attributes that are not integrity-protected. If such attributes influence subsequent charges, then the possibility of fraud would remain.

7.3. Theft of passwords

RADIUS as defined in [3] does not provide for end-to-end confidentiality. As a result, where clients authenticate using PAP, each proxy along the path between the local NAS and the home server will have access to the cleartext password. In many circumstances, this represents an unacceptable security risk.

7.4. Theft and modification of accounting data

Typically in roaming systems, accounting packets are provided to all the participants along the roaming relationship path, in order to allow them to audit subsequent invoices. RADIUS as described in [3] does not provide for end-to-end security services, including integrity protection or confidentiality. Without end-to-end integrity protection, it is possible for proxies to modify accounting packets or session records. Without end-to-end confidentiality, accounting

data will be accessible to proxies. However, if the objective is merely to prevent snooping of accounting data on the wire, then IPSEC ESP can be used.

7.5. Replay attacks

In this attack, a man in the middle or rogue proxy collects CHAP-Challenge and CHAP-Response attributes, and later replays them. If this attack is performed in collaboration with an unscrupulous ISP, it can be used to subsequently submit fraudulent accounting records for payment. The system performing the replay need not necessarily be the one that initially captured the CHAP Challenge/Response pair.

While RADIUS as described in [3] is vulnerable to replay attacks, without roaming the threat is restricted to proxies operating in the home server's domain. With roaming, such an attack can be mounted by any proxy capable of reaching the home server.

7.6. Connection hijacking

In this form of attack, the attacker attempts to inject packets into the conversation between the NAS and the home server. RADIUS as described in [3] is vulnerable to such attacks since only Access-Reply and Access-Challenge packets are authenticated.

7.7. Fraudulent accounting

In this form of attack, a local proxy transmits fraudulent accounting packets or session records in an effort to collect fees to which they are not entitled. This includes submission of packets or session records for non-existent sessions. Since in RADIUS as described in [3], there is no end-to-end security, a rogue proxy may insert or edit packets without fear of detection.

In order to detect submissions of accounting packets or session records for non-existent sessions, parties receiving accounting packets or session records would be prudent to reconcile them with the authentication logs. Such reconciliation is only typically possible when the party acts as an authentication proxy for all sessions for which an accounting record will subsequently be submitted.

In order to make reconciliation easier, home servers involved in roaming include a Class attribute in the Access-Accept. The Class attribute uniquely identifies a session, so as to allow an authentication log entry to be matched with a corresponding accounting packet or session record.

If reconciliation is put in place and all accounting log entries without a corresponding authentication are rejected, then the attacker will need to have obtained a valid user password prior to submitting accounting packets or session records on non-existent sessions. While use of end-to-end security can defeat unauthorized injection or editing of accounting or authentication packets by intermediate proxies, other attacks remain feasible. For example, unless replay protection is put in place, it is still feasible for an intermediate proxy to resubmit authentication or accounting packets or session records. In addition, end-to-end security does not provide protection against attacks by the local proxy, since this is typically where end-to-end security will be initiated. To detect such attacks, other measures need to be put in place, such as systems for detecting unusual activity of ISP or user accounts, or for determining whether a user or ISP account is within their credit limit.

Note that implementation of the store and forward approach to proxy accounting makes it possible for some systems in the roaming relationship path to receive accounting records that other systems do not get. This can result in audit discrepancies. About the best that is achievable in such cases is to verify that the accounting data is missing by checking against the authentication logs.

8. Acknowledgments

Thanks to Pat Calhoun of Sun Microsystems, Mark Beadles of CompuServe, Aydin Edguer of Morningstar, Bill Bulley of Merit, and Steven P. Crain of Shore.Net for useful discussions of this problem space.

9. Authors' Addresses

Bernard Aboba
Microsoft Corporation
One Microsoft Way
Redmond, WA 98052

Phone: 425-936-6605
EMail: bernarda@microsoft.com

John R. Vollbrecht
Merit Network, Inc.
4251 Plymouth Rd.
Ann Arbor, MI 48105-2785

Phone: 313-763-1206
EMail: jrv@merit.edu

10. Full Copyright Statement

Acknowledgement

Funding for the RFC Editor function is currently provided by the Internet Society.