NETWORK WORKING GROUP
Internet-Draft
Expires: May 22, 2008

N. Williams
Sun

November 19, 2007

GSS-API Internationalization and Domain-Based Service Names and Name Type

draft-ietf-kitten-gssapi-domain-based-names-04.txt

Status of this Memo

Copyright Notice

Abstract

This document describes domainname-based service principal names and the corresponding name type for the Generic Security Service Application Programming Interface (GSS-API). Internationalization of the GSS-API is also covered.

Domain-based service names are similar to host-based service names, but using a domain name (not necessarily an Internet domain name) in addition to a hostname. The primary purpose of domain-based names is to provide a measure of protection to applications that utilize insecure service discovery protocols. This is achieved by providing a way to name clustered services after the "domain" which they service, thereby allowing their clients to authorize the service's servers based on authentication of their service names.

Table of Contents

1. Conventions used in this document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

2. Introduction

Some applications need to discover the names of servers for a specific resource. Some common methods for server discovery are insecure, e.g., queries for DNS [RFC1035] SRV resource records [RFC2782] without using DNSSEC [RFC4033] and subject to attacks whereby a client can be re-directed to incorrect and possibly malicious servers. A client may even be re-directed to a server that has credentials for itself and may thus authenticate itself to the client, and yet it could be incorrect or malicious (because it has been compromised, say).

Domain-based names allow for GSS-API [RFC2743] initiator applications (clients) to authorize acceptor principals (servers) to serve the resource for which the client used insecure server discovery without either securing the server discovery method nor requiring an additional protocol for server authorization -- either a discovered server has credentials for authenticating the domain-based service names that it is intended to respond to, or it does not. Availability of valid credentials for authenticating domain-based names embodies the authorization of a given server to a domain-wide service.

A domain-based name consists of three required elements:

- a service name
- a domain name
- a hostname

The domain name and the hostname should be Domain Name System (DNS) names, though domain-based names could be used in non-DNS environments. Because of the use of DNS names we must also provide for internationalization of the GSS-API.

Note that domain-based naming isn't new. According to a report to the KITTEN WG mailing list there exists at least one implementation of LDAP which uses domain-based service naming, and the DIGEST-MD5 HTTP/SASL mechanism [RFC2831] describes a similar notion (see section 2.1.2, description of the "serv-name" field of the digest-response).

3. Name Type OID and Symbolic Name

The new name type has an OID of

[NOTE: OID assignment to be made with IANA.]

{iso(1) org(3) dod(6) internet(1) security(5) nametypes(6) gss-domain-based(5)}

The recommended symbolic name for this GSS-API name type is "GSS_C_NT_DOMAINBASED_SERVICE".

4. Query and Display Syntaxes

There is a single name syntax for domain-based names.

The syntax is:

```
domain-based-name :=

    <service> '@' <domain> '@' <hostname>
```

Note that for Internet domain names the trailing '.' MUST NOT be included in the hostname part of the display form GSS-API domain-based MNs; hostnames MUST NOT contain '@'.

4.1. Examples of domain-based names

These examples are not normative:

- ldap@example.tld@ds1.example.tld
- nfs@example.tld@nfsroot1.example.tld

5. Internationalization (I18N) considerations

We introduce new versions of GSS_Import_name() and GSS_Display_name() to better support Unicode. Additionally we provide for the use of ACE-encoded DNS in the non-internationalized interfaces [RFC3490].

5.1. Importing internationalized names

When the input_name_type parameter is the GSS_C_NT_DOMAINBASED_SERVICE OID then GSS_Import_name() implementations and GSS-API mechanisms MUST accept ACE-encoded internationalized domain names in the hostname and domain name slots of the given domain-based name string.

Support for non-ASCII internationalized domain names SHOULD be provided through a new function, GSS_Import_name_utf8(), that operates exactly like GSS_Import_name(), except that it MUST accept internationalized domain names both, as UTF-8 strings and as ACE-encoded strings via its input_name_string argument.

5.2. Displaying internationalized names

Implementations of GSS_Display_name() MUST only output US-ASCII or ACE-encoded internationalized domain names in the hostname and domain name slots of domain-based names (or mechanism names (MN) that conform to the mechanism's form for domain-based names).

Support for non-ASCII internationalized domain names SHOULD be provided through a new function, GSS_Display_name_utf8(), that operates exactly like GSS_Display_name(), except that it outputs UTF-8 strings via its name_string output argument. GSS_Display_name_utf8() MUST NOT output ACE-encoded internationalized domain names.

6. Application protocol examples

The following examples are not normative. They describe how the author envisions two applications' use of domain-based names.

6.1. NFSv4 domain-wide namespace root server discovery

Work is ongoing to provide a method for constructing domain-wide NFSv4 [RFC3530] filesystem namespaces where there is a single "root" with one or more servers (replicas) and multiple filesystems glued into the namespace through use of "referrals." Clients could then construct a "global" namespace through use of the DNS domain hierarchy.

Here clients would always know, from context, when they need to find the root servers for a given DNS domain. Root server discovery would be performed using DNS SRV RR lookups, without using DNSSEC where DNSSEC has not been deployed.

When using RPCSEC_GSS [RFC2203] for security NFSv4 clients would then use domain-based names to ensure that that the servers named in the SRV RRs are in fact authorized to be the NFSv4 root servers for the target domain.

6.2. LDAP server discovery

LDAP clients using the GSS-API through SASL too would benefit from use of domain-based names to protect server discovery through insecure DNS SRV RR lookups, much as described above.

Unlike NFSv4 clients, not all LDAP clients may always know from context when they should use domain-based names. That's because existing clients may use host-based naming to authenticate servers discovered through SRV RR lookups. Changing such clients to use domain-based naming when domain-based acceptor credentials have not been deployed to LDAP servers, or when LDAP servers have not been modified to allow use of domain-based naming, would break interoperability. That is, there is a legacy server interoperability issue here. Therefore LDAP clients may require additional configuration at deployment time to enable (or disable) use of domain-based naming.

Note: whether SASL [RFC4422] or its GSS-API bridges [I-D.ietf-sasl-gssapi] [I-D.josefsson-sasl-gs2] require updates in order allow use of domain-based names is not relevant to the theory of how domain-based naming would protect LDAP clients' server discovery.

7. Security Considerations

Use of GSS-API domain-based names may not be negotiable by some GSS-API mechanisms, and some acceptors may not support GSS-API domain-based names. In such cases initiators are left to fallback on the use of hostbased names, in which case the initiators MUST also verify that the acceptor's hostbased name is authorized to provide the given service for the domain that the initiator had wanted.

The above security consideration also applies to all GSS-API initiators who lack support for domain-based service names.

8. References

8.1. Normative References

[RFC1035] Mockapetris, P., "Domain names - implementation and specification", STD 13, RFC 1035, November 1987.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC2743] Linn, J., "Generic Security Service Application Program Interface Version 2, Update 1", RFC 2743, January 2000.

[RFC2782] Gulbrandsen, A., Vixie, P., and L. Esibov, "A DNS RR for specifying the location of services (DNS SRV)", RFC 2782, February 2000.

[RFC2831] Leach, P. and C. Newman, "Using Digest Authentication as a SASL Mechanism", RFC 2831, May 2000.

[RFC3490] Faltstrom, P., Hoffman, P., and A. Costello, "Internationalizing Domain Names in Applications (IDNA)", RFC 3490, March 2003.

[RFC4033] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "DNS Security Introduction and Requirements", RFC 4033, March 2005.

8.2. Informative References

[I-D.ietf-sasl-gssapi] Melnikov, A., "The Kerberos V5 ("GSSAPI") SASL mechanism", draft-ietf-sasl-gssapi-08 (work in progress), September 2006.

[I-D.josefsson-sasl-gs2] Josefsson, S., "Using GSS-API Mechanisms in SASL: The GS2 Mechanism Family", draft-josefsson-sasl-gs2-00 (work in progress), November 2005.

[RFC2203] Eisler, M., Chiu, A., and L. Ling, "RPCSEC_GSS Protocol Specification", RFC 2203, September 1997.

[RFC3530] Shepler, S., Callaghan, B., Robinson, D., Thurlow, R., Beame, C., Eisler, M., and D. Noveck, "Network File System (NFS) version 4 Protocol", RFC 3530, April 2003.

[RFC4422] Melnikov, A. and K. Zeilenga, "Simple Authentication and

Security Layer (SASL)", RFC 4422, June 2006.

Author's Address

Nicolas Williams
Sun Microsystems
5300 Riata Trace Ct
Austin, TX 78727
US

Email: Nicolas.Williams@sun.com

Full Copyright Statement

Intellectual Property

Acknowledgment

Funding for the RFC Editor function is provided by the IETF Administrative Support Activity (IASA).