Internet Engineering Task Force	P. M. Hallam-Baker
Internet-Draft	B. Smith
Intended status: Informational	Comodo Inc.
Expires: September 10, 2011	March 09, 2011

DNS Extended Service Discovery (ESRV) Record.
draft-hallambaker-esrv-01

Abstract

General Service Description (GSRV) and Extended Service Description records are DNS Resource Records that provide information to applications attempting to establish a network connection. When authenticated using an appropriate means GSRV and ESRV records may be used to prevent a downgrade attack in cases where use of security enhancements with an application protocol are optional.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet- Drafts is at http://datatracker.ietf.org/drafts/current/.

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."

This Internet-Draft will expire on September 10, 2011.

Copyright Notice

Table of Contents

1. Definitions

The following definitions are used in this document:

Abstract Syntax Notation One (ASN.1) A notation for describing abstract types and values, as specified in X.680 *[X.680]*.

Authorization Entry An authorization assertion that grants or denies a specific set of permissions to a specific group of entities.

Canonical Domain Name A Domain Name that is not an alias.

Canonical Domain Name Value The value of a Canonical Domain Name. The value resulting from applying alias transformations to a Domain Name that is not canonical.

Certificate An X.509 Certificate, as specified in RFC 5280 *[RFC5280]*.

Certification Policy (CP) Specifies the criteria that a Certification Authority undertakes to meet in its issue of certificates.

Certification Practices Statement (CPS) Specifies the means by which the criteria of the Certification Policy are met. In most cases this will be the document against which the operations of the Certification Authority are audited.

Certification Authority (CA) An entity that issues Certificates in accordance with a specified Certification Policy.

Distinguished Encoding Rules (DER) A set of rules for encoding ASN.1 objects, as specified in X.690 *[X.690]*.

Domain The set of resources associated with a DNS Domain Name.

Domain Name A DNS Domain name as specified in RFC 1035 *[RFC1035]* and revisions.

Domain Name System (DNS) The Internet naming system specified in RFC 1035 *[RFC1035]* and revisions.

DNS Security (DNSSEC) Extensions to the DNS that provide authentication services as specified in RFC 4033 *[RFC4033]* and revisions.

Public Delegation Point A Domain Name that is obtained from a public DNS registry as defined by a Certification Policy.

Public Key Infrastructure X.509 (PKIX) Standards and specifications issued by the IETF that apply the X.509 *[X.509]* certificate standards specified by the ITU to Internet applications as specified in RFC 5280 *[RFC5280]* and related documents.

Resource Record (RR) A set of attributes bound to a Domain Name.

1.1. Requirements Language

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 RFC 2119 *[RFC2119]*.

2. Extended Service Discription

General Service Description (GSRV) and Extended Service Discrpition (ESRV) DNS Resource Records provide a mechanism for specifying properties relating to Internet services associated with a DNS name. GSRV and ESRV records are intended to serve the same function at different levels of generality. GSRV records specify properties that apply to all Internet services provided in the Domain. ESRV records are used to specify properties that apply to finer levels of detail. Extended Service Description allows properties to be expressed at three levels of granularity:

Domain
Properties that apply to all services offered at the corresponding domain name. Domain level properties do not take prefixes and are published using the GSRV Resource Record.

For example, a site might declare that all services offered at a domain name support use of SRV service discovery.

Service Properties that apply to all instances of a service offered at the corresponding domain name. Service level properties always take a service specific prefix and are published using the ESRV Resource Record.

For example, a site might specify that the SMTP service always supports use of the TLS security protocol (via the STARTTLS mechanism) while use of the TLS security protocol is required for IMAP, POP and HTTP connections.

Instance Properties that apply to a specific instance of a service on a specific host listening on a specific port. Instance level properties always take a service specific prefix and a port specific prefix and are published using the ESRV Resource Record.

Declaration of Instance specific properties is only possible when a DNS service discovery protocol such as MX, SRV, NAPTR or DDDS is in use.

For example a site with two servers offering SMTP email service might advertise different TLS certificates for each service instance.

2.1. Use of DNS Prefixing

GRSV/ESRV records make use of the service prefixing mechanism introduced in SRV and employed in later advanced service discovery mechanisms such as NAPTR and URI.
The need for separate DNS Resource Record types to express properties at different levels of granularity arises from the need to support use of wildcards and DNS aliases such as CNAME and DNAME records when specifying properties that apply to a whole domain and the need to use prefix labels to specify properties at finer granularity.

2.1.1. Interaction with Extended Service Discovery

The GSRV/ESRV discovery mechanism is designed for use by itself or in combination with service discovery mechanisms such as SRV, NAPTR and URI.
One of the main limitations of service discovery schemes such as SRV, NAPTR and URI is that they can only be used if a client knows to look for them.
Without provision for meta-service discovery, the service discovery mechanisms supported by a protocol are limited to those which exist at the time the protocol is developed and that the protocol designer decides to support. In most cases however, it is the site administrator

rather than the protocol designer who is best placed to know which form of extended discovery is most applicable to their service.

GSRV records may be used to inform clients that a service discovery mechanism is supported for specific protocols or for all protocols. In the following example clients are advised to attempt service discovery using the SRV mechanism for the HTTP protocol and to attempt URI service discovery for all others:

```
    example.com   GSRV 0   srv  "_http._tcp"
    example.com   GSRV 0   uri "*"
```

2.1.2. Abstract Services

Publication of ESRV properties for abstract services allows a site to enable clients to perform protocol negotiation by specifying the range of services offered that support a specific purpose.

For example, the SMTP, POP3 and IMAP4 protocols are all used for exchange of mail. An abstract service for the mail protocol with the prefix '_mail._as' allows clients to discover the full range of mail related protocols in a single query.

```
example.com             GSRV 0  service "_mail._as"
_mail._as.example.com   ESRV 0  prot "_smtp._tcp"
_mail._as.example.com   ESRV 0  prot "_pop._tcp"
_mail._as.example.com   ESRV 0  prot "_imap._tcp"
```

2.2. Resolution Mechanism

Extended Service description records MAY be resolved in an Incremental mode or an Optimized mode. Incremental mode allows records to be advertised and resolved by existing DNS servers, optimized mode allows for improved performance when

Implementations MUST support the Incremental mode and MAY support the Optimized mode.

2.2.1. Incremental Mode

The incremental mode allows extended service discovery to be used in conjunction with DNS resolvers that do not support the Optimized mode.

2.2.1.1. Domain Resolution

In the incremental mode the DNS client attempting service discovery begins by querying for Domain level properties.

For example, a client attempting to perform extended service discovery for the HTTP protocol server at www.example.com would begin by querying for the GSRV record at www.example.com.

Domain properties MAY include property entries for the service property type. The service property entry indicates that additional property entries are specified at the service level for either a specific serviceby means of the protocol prefix or for all services by means of the wildcard entry "*".

2.2.1.2. Service Resolution

If the domain resolution indicates that property entries are declared for the service prefix being resolved or for the wildcard prefix type, service level resolution is performed.

There are three possible outcomes in which the query is successful and returns a GSRV record:

- The queried domain name is canonical and a GSRV record is returned for that domain name.
- The queried domain name is not canonical and an alias is returned (e.g. CNAME) together with the GSRV record for the canonical name.
- The queried domain name does not exist but is in the scope of a wildcard record.

In the last case we require that the GSRV record set returned contain a canonical property entry specifying a canonical name. We are thus assured that

prefix the canonical name to perform the service lookup

In the following example a query for the _prefix protocol at domain names canonical.example.com, cname.example.com or wildcard.example.com will all result in the resolution process continuing with a query for an ESRV record for _prefix.canonical.example.com.

```
$ORIGIN example.com
canonical          GSRV 0  service "*"
cname.example.com  CNAME canonical.example.com
*.example.com      GSRV 0  canonical "canonical.example.com"
*.example.com      GSRV 0  service "*"
```

If use of a service discovery mechanism is indicated a site MAY choose to advertise properties specific to a particular service instance through use of the instance property type.

2.2.1.2.1. Prefetching of Protocol Records

Client implementations MAY attempt to optimize incremental mode discovery by initiating service resolution in parallel with domain resolution by applying the protocol prefix to the query domain.

If the response to the domain resolution query indicates that the query domain is not-canonical, any answer returned to the pre-fetched query is ignored and a new query made if inidcated by the answer returned to the domain resolution query.

2.2.1.3. Instance Resolution

If the service level resolution indicates that instance level property entries MAY exist, these are resolved by querying for the chosen host name prefixed by the service prefix and a second prefix formed from the decimal port identifier.
For example a the www.example.com HTTP server is supported by two separate service instances (host1, host2). TLS security is always offered on host1 instance but not on host2:

```
$ORIGIN example.com
www                     GSRV 0 service "*"
www                     GSRV 0 srv "_http._tcp"
_http._tcp.www          ESRV 0 instance ""
_http._tcp.www          SRV 1 1 80 host1.example.com
_http._tcp.www          SRV 1 1 80 host2.example.com
_http._tcp._80.host1    ESRV 0 tls "required"
```

2.2.2. Optimized Mode

DNS servers MAY advertise support for the optimized mode query by means of the EDNS0 meta-query extension.
When optimized mode queries are supported, the client MAY present a meta-query consisting of the protocol prefix concatenated to an iteration count concatenated to the query domain. The server receiving the query then performs the GSRV/ESRV discovery process on the client's behalf and returns the whole result chain in a single response.
For example, a query for the HTTP service at example.com would be made as:

```
METASRV ? _http._tcp._0.www.example.com
```

Since the two hosts have the same weighting, there is a 50% probability that the response to this query would be:

```
_http._tcp._0.www.example.com METASRV 2
www.example.com               GSRV 0 service "*"
www.example.com               GSRV 0 srv "_http._tcp"
_http._tcp.www.example.com    ESRV 0  instance ""
_http._tcp.www.example.com    SRV 1 1 80 host2.example.com
_http._tcp._80.host1          ESRV 0  tls "required"
```

Should the attempt to connect to host1.example.com fail, the client MAY make a second attempt:

```
METASRV ? _http._tcp._1.www.example.com
```

Since there is now only one service instance remaining, the response would be:

```
_http._tcp._1.www.example.com METASRV 2
www.example.com               GSRV 0  service "*"
www.example.com               GSRV 0  srv "_http._tcp"
_http._tcp.www.example.com    ESRV 0  instance ""
_http._tcp.www.example.com    SRV 1 1 80 host1.example.com
_http._tcp._80.host1          ESRV 0  tls "required"
```

[TBS: work out how to allow the server to calculate the random number deterministically on the query, may need to add a state parameter here.]

2.2.2.1. EDNS0 Meta-Query Extension

The EDNS0 Meta-Query extension has code TBS and is used to advertise support for one or more meta-queries.
The parameter data for the Meta-Query Extension consists of a list of DNS query numbers for the supported meta queries.

2.2.3. Interactive HTTP Properties

Interactive HTTP Attributes are properties specific to the HTTP protocol that are declared as Domain Properties rather than service properties. Specifying the properties for HTTP as interactive HTTP properties allows clients using resolvers that do not support the optimized resolution mode to resolve HTTP properties in a single round trip rather than two.
For example, the following configuration specifies that tls is always offered for the application protocols HTTP and POP.

```
$ORIGIN example.com
.                      GSRV 0  service "*"
.                      GSRV 0  http_tls "required"
_pop._tcp              ESRV 0  tls "offered"
```

The above example is functionally equivalent to specifying the HTTP properties as instance properties:

```
$ORIGIN example.com
.                       GSRV 0  service "*"
_http._tcp              ESRV 0  tls "required"
_pop._tcp               ESRV 0  tls "offered"
```

Special provision is justified in this instance by the widespread use of HTTP and the effect that service discovery latency has on the Web Browser user experience.

2.3. Syntax

The GSRV and ESRV have the same record syntax which is the same as the syntax of the CAA record. A GSRV or ESRV RR contains a single property entry consisting of a tag value pair. Each tag represents a property of the CAA record. The value of a property entry is that specified in the corresponding value field.
A domain name MAY have multiple GSRV or ESRV RRs associated with it and a given property MAY be specified more than once. Where multiple properties are specified they are additive. That is if SRV and URI records are advertised for a service then both mechanisms for advanced discovery are offered.
The GSRV/ESRV data field consists of a sequence of at least one property entry. Each property entry consists of a sequence of:

Flags One octet containing the following field:

> **Bit 0: Critical Flag** If the value is set (1), the critical flag is asserted and the property MUST be understood if the record is to be correctly processed.
>
> Note that according to the conventions set out in RFC 1035 *[RFC1035]* Bit 0 is the Most Significant Bit and Bit 7 is the Least Significant. Thus a flags value of 0x51 indicates a tag length of 5 octets and that the property entry is not critical and is not to be used for relying party processing.

Tag Length A single octet containing an unsigned integer specifying the tag length in octets. The tag length MUST be at least 1 and SHOULD be no more than 15.

Tag The property identifier, a sequence of ASCII characters.

> Tag values MAY contain ASCII characters a through z and the numbers 0 through 9. Tag values MUST NOT contain any other characters. Matching of tag values is case insensitive.

Value A sequence of octets representing the property value. Property values are encoded as binary values and MAY employ sub-formats.

> The length of the value field is specified implicitly as the remaining length of the enclosing Resource Record data field.

2.3.1. Presentation Format

The presentation format of the GSRV and ESRV resource records is as follows:
GSRV <flags> <tag> <data>
ESRV <flags> <tag> <data>
Where:

flags Is an unsigned integer between 0 and 15.

tag Is a non-zero sequence of ASCII letter and numbers in lower case.

data The parameter data for the property specified as either a quoted text string or an unquoted Base64 Encoding [RFC4648] of the value.

2.4. ESRV Processing Rules

The purpose of extended service discovery is to refine an abstract specification of a DNS host name and protocol prefix to a concrete specification for the name of a specific network host, a specific network port number and a specific network protocol and associated protocol parameters.

2.4.1. Service Discovery

The GSRV and ESRV Resource Record is used in combination with service discovery records (e.g. SRV, URI NAPTR) to perform extended service discovery. An API call for extended service discovery has the following signature:

name (input/output) The DNS name of the service.

protocol (input/output) The DNS prefix of the protocol.

port (output) The IP port number to conect to at the host

uri (output) The canonical uri of the service to connect to

properties (output) A list of attribute value pairs that specify characteristics of the connection to the service.

2.4.2. Abstract Service Discovery

The GSRV and ESRV Resource Records may be used to perform abstract service discovery providing a list of supported protocols that implement the abstract service. An API call for abstract service discovery has the following signature:

name (input/output)
The DNS name of the service.

protocol (input) The DNS prefix of the abstract protocol.

implementations (output) A list of DNS prefixes for the protocols implementing the specified protocol that are supported at the specified domain.

3. Properties

3.1. Property Values

GSRV/ESRV properties take different parameter values according to the specific label.

[domain_name] A DNS domain name.

[protocol_specifier] An ASCII string containing a protocol prefix string or the wildcard character '*'.

[tls_specifier] A sequence of one or more TLS attributes or attribute value pairs.

3.2. Processing Properties

Processing properties direct the process of property resolution. Since a processing property is used to direct the resolution process, they only have effect when encountered at a relevant stage of resolution. A service property that directs a client to resolve for service properties has no effect.
Three processing properties are defined.

canonical [domain_name] The canonical property directs the client to use the specified property value as the canonical domain name to be used in service resolution.

The canonical property only has effect if declared as a domain property.

service [protocol_specifier] The service property directs the client to perform service resolution if the property value matches the query protocol.

The service property only has effect if declared as a domain property.

instance [protocol_specifier] The service property directs the client to perform instance resolution if the property value matches the query protocol.

The instance property only has effect if declared as a domain property or as a service property.

3.3. Discovery Properties

Discovery Properties specify the resolution mechanism(s) to be used for service resolution. Discovery properties only have effect when encountered in the Domain resolution phase and always take a protocol_specifier as the property type.
If multiple discovery properties are specified, the most specific property or properties are to be used.
If no discovery property is specified, 'a' record service discovery (i.e. IPv4) is indicated.

a [protocol_specifier] Specifies discovery by means of A record lookup; the traditional means of service resolution for IPv4.

Since A record lookup is the default, this property is most likely to be used to specify a protocol specific exception to a wildcard discovery property.

aaaa [protocol_specifier] Specifies discovery by means of AAAA record lookup; the traditional means of service resolution for IPv6.

srv [protocol_specifier] Specifies discovery using the SRV service discovery mechanism.

uri [protocol_specifier] Specifies discovery using the URI service discovery mechanism.

naptr [protocol_specifier] Reserved for specifying discovery using the URI service discovery mechanism.

Since NAPTR records are designed to support URN resolution rather than service resolution, the manner of using NAPTR records within the GSRV/ESRV framework is left unspecified in this version of the specification.

prot [protocol_specifier] Specifies that the specified protocol MAY be resolved to discover a protocol related to the specified protocol.

The prot Discovery Property is used to support abstract service resolution and alternative service resolution. When encountered in the domain resolution phase, the prot property advises the resolver that an alternative to the requested protocol is available.

In the case of an abstract protocol such as a prefix representing 'email', there is no concrete service to resolve and the only discovery properties that are valid are prot discovery properties.

In the case of a concrete protocol such as _http._tcp, a prot specifier MAY be used to advise the resolver that use of an alternative protocol is available (e.g. _httpng._tcp).

3.4. Security Properties

Security properties allow a service provider to describe the security critieria for a service. When an ESRV record is secured using an appropriate means (e.g. DNSSEC), the security properties MAY be used to prevent a downgrade attack on the security enhancements offered.

tls [tls_specifier] Specifies the use of tls. Valid property values are refused, optional and required.

If the property value refused is specified, the corresponding service does not support use of the TLS security enhancement.

If the property value optional is specified, the corresponding service always offers use of the TLS security enhancement.

If the property value optional is specified, the corresponding service requires of use of the TLS security enhancement.

http-tls [tls_specifier] Specifies the use of tls with HTTP. Valid values are refused, optional and required with the same semantics as for the tis property.

The http-tls property allows security properties to be declared for the http protocol as domain properties, thus allowing the properties to be resolved in a single round trip.

It is anticipated that the list of tags will be expanded at a future date to support other Internet security protocols such as IPSEC and WS-Security.

3.4.1. tls_specifier Attributes

A tls_specifier property value consists of a sequence of case insensitive atributes or attribute-value pairs separated by spaces as follows:
[EBNF To Be Specified]

refused
The specified service or service instance does not support TLS.

optional[=<port]> The specified service or service instance always offered on an alternative IP port but does not require use of TLS.

If the port parameter is specified, it speficies the port number for the TLS service in decimal.

If no port number is specified the TLS service is provided on the default port for using TLS with the specified protocol.

required[=<port]> The specified service or service instance requires use of TLS.

If the port parameter is specified, it speficies the port number for the TLS service in decimal.

If no port number is specified the TLS service is provided on the default port for using TLS with the specified protocol.

upgrade The specified service or service instance always accepts a client request to upgrade the connection to use TLS.

4. Relation to Existing Work

Extended Service Discrpition (ESRV) DNS Resource Records extend the principles of named service discovery first proposed in SRV RFC 2782 *[RFC2782]* and later extended in NAPTR RFC 3403 *[RFC3403]* and related specifications. The ESRV record provides an extensible container that MAY be used to provide a description of an abstract Internet service bound to a domain name or a specific instance of that Internet Service on a specific host.

Existing SRV records provide a means of allowing a client to discover a host and port number for a specific Internet protocol. ESRV records allow a further layer of abstraction in which the discovery is of an Internet Service and the service provider MAY declare a range of supported protocol options.

For example, POP RFC 1939 *[RFC1939]* and IMAP RFC 2060 *[RFC2060]* both provide means by which a mail client can access mail messages but the only means by which a client might discover that both protocols are supported is to attempt to connect to each in turn.

Although discovery by polling is practical when there are only two options, it is impractical in application areas such as federated authentication (also known as 'identity') where the number of protocols that might be employed is very large (e.g. Kerberos, SAML, OpenID, etc.) and the number of ways in which those protocols might be employed is even larger still. Not only is polling inefficient in such circumstances, a client that fails to find a means of connection has no way to know how it might have succeeded.

ESRV records provide a means by which Internet clients and Servers can negotiate choice of protocols and protocol properties. In particular, when publication of the ESRV record is appropriately secure (e.g. through a use of DNSSEC RFC 4033 *[RFC4033]*), the ESRV record provides a means of securely negotiating critical security properties.

Today use of Internet security is the exception rather than the rule. As a result, an attacker can frequently bypass security enhancements by persuading the parties that they do not exist.

This form of attack is a downgrade attack. While protocols such as SSL RFC 5246 *[RFC5246]* and S/MIME have measures that are intended to prevent downgrade attacks in which weaker algorithms are substituted for strong, there is currently no in-band mechanism for specifying that these enhancements are available or that they should or must be used.

While it is possible to infer such information from existing DNS records such as the port number specification in an SRV record, such approaches represent heuristics and as such are not appropriate as a means of achieving an essential security objective.

5. Security Considerations

TBS

6. IANA Considerations

TBS

7. References

7.1. Normative References

[RFC2119]	Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
[RFC3403]	Mealling, M., "Dynamic Delegation Discovery System (DDDS) Part Three: The Domain Name System (DNS) Database", RFC 3403, October 2002.
[RFC2782]	Gulbrandsen, A., Vixie, P. and L. Esibov, "A DNS RR for specifying the location of services (DNS SRV)", RFC 2782, February 2000.
[RFC4648]	Josefsson, S., "The Base16, Base32, and Base64 Data Encodings", RFC 4648, October 2006.
[RFC4871]	Allman, E., Callas, J., Delany, M., Libbey, M., Fenton, J. and M. Thomas, "DomainKeys Identified Mail (DKIM) Signatures", RFC 4871, May 2007.
[RFC5246]	Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2", RFC 5246, August 2008.
[RFC5280]	Cooper, D., Santesson, S., Farrell, S., Boeyen, S., Housley, R. and W. Polk, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 5280, May 2008.
[X.509]	International Telecommunication Union , "ITU-T Recommendation X.509 (11/2008): Information technology - Open systems interconnection - The Directory: Public-key and attribute certificate frameworks ", ITU-T Recommendation X.509, November 2008.

[X.680]	International Telecommunication Union , "ITU-T Recommendation X.680 (11/2008): Information technology - Abstract Syntax Notation One (ASN.1): Specification of basic notation ", ITU-T Recommendation X.680, November 2008.
[X.690]	International Telecommunication Union , "ITU-T Recommendation X.690 (11/2008): Information technology - Abstract Syntax Notation One (ASN.1): Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER) ", ITU-T Recommendation X.690, November 2008.

7.2. Non-Normative References

[RFC1035]	Mockapetris, P., "Domain names - implementation and specification", STD 13, RFC 1035, November 1987.
[RFC2905]	Vollbrecht, J., Calhoun, P., Farrell, S., Gommans, L., Gross, G., de Bruijn, B., de Laat, C., Holdrege, M. and D. Spence, "AAA Authorization Application Examples", RFC 2905, August 2000.
[RFC1939]	Myers, J.G. and M.T. Rose, "Post Office Protocol - Version 3", STD 53, RFC 1939, May 1996.
[RFC2060]	Crispin, M., "Internet Message Access Protocol - Version 4rev1", RFC 2060, December 1996.
[RFC4033]	Arends, R., Austein, R., Larson, M., Massey, D. and S. Rose, "DNS Security Introduction and Requirements", RFC 4033, March 2005.
[RFC3851]	Ramsdell, B., "Secure/Multipurpose Internet Mail Extensions (S/MIME) Version 3.1 Message Specification", RFC 3851, July 2004.

Authors' Addresses

Phillip Hallam-Baker Hallam-Baker Comodo Inc. EMail: philliph@comodo.com

Brian Smith Smith Comodo Inc. EMail: brian@dns.com