Abstract

This document discusses how private use fields in IETF protocols are used.

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 https://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 April 19, 2020.

Copyright Notice

Copyright (c) 2019 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust’s Legal Provisions Relating to IETF Documents (https://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

This document may contain material from IETF Documents or IETF Contributions published or made publicly available before November
Protocols having options reserved for testing and experimentation have been found to be beneficial to the community as discussed in "Assigning Experimental and Testing Numbers Considered Useful"
As with any protocol detail, the effectiveness of private use fields depends upon a shared understanding of their syntax and semantics by all participating implementations. For open use in the Internet, this requires that such fields be fully specified in openly available documents.

This taxonomy uses examples of some protocols to discuss how some private use options are used.

1.1. Nomenclature

In this document, the following terms are defined to prevent ambiguity. Some of these words have not been used in the referenced works but their meanings can be ascertained and applied.

- **Standard Option** - a field in a protocol frame that may only use values that are strictly defined within a specification.

- **Private Use Option** - a field in a protocol frame that is reserved for private, experimental, testing, or local use only.

- **Namespace** - a fully qualified Standard Option or Private Use Option.

1.2. Note on References

In many cases throughout this document, RFCs are referenced even though they are not the most current version of their respective protocol. This is only done to reference the first occurrence of a private use option, or to point out a distinct feature in that specification. When an RFC is referenced that is not the current version, the reference will be followed by the RFC number of the current version in curly braces.

2. Origins of the Private Use Namespace

Some standards permit private use options in different ways, while others do not. The "Time Protocol" [RFC0868] is an example of a protocol that only conveys standardized information; there is no way to use private options and no way to add anything other than what is specified in the document. In a more open way, "DOD STANDARD TRANSMISSION CONTROL PROTOCOL" [RFC0761] [RFC0793] [RFC7805] does have "options" but they must be registered through the IANA [IANAtcp] before use, which does not leave any room for optional information supplied by equipment vendors, network operators, or experimenters. An even more open way may be seen in "Vendor-Identifying Vendor Options for Dynamic Host Configuration Protocol version 4 (DHCPv4)"
[RFC3925], which allows for vendor specific options that do not need to be registered with anyone.

For the case of TCP [RFC0761] {{RFC0793}} {{RFC7805}}, standard options are expected; senders may use them and receivers may be configured to act upon that information, or to ignore it. If an experimenter wants to add an option, they will have to create a new IETF RFC with specific details, or obtain approval from the IESG to have the IANA add to the registry [IANAtcp]. Similarly, if equipment vendors Foo and Bar were to have a need for a similar option within TCP, they would each have to go through the process to add to the registry. They may then need to negotiate how they would interpret each other's options for some level of interoperability. On the other hand, if a properly crafted multipurpose private use option were to be registered, such as in the case of multiple vendor instances within "DHCPv4" [RFC3925], then vendors and experimenters would each be able to use it for their own purpose as long as all network participants could easily differentiate between the entities using the option.

"Guidelines for Writing an IANA Considerations Section in RFCs" [RFC2434] {{RFC8126}} describes that values of specific namespaces may either be registered with the IANA, or not. In most cases, there are well defined values for their respective namespaces. However, as the document explains, not all namespaces require centralized administration. In that document, it seems to be assumed that private use namespaces will be domain specific and it will be up to the administrators of any domain to avoid conflicts. The first example given about private use namespaces refers to "Dynamic Host Configuration Protocol" [RFC2131] and presumably "DHCP Options and BOOTP Vendor Extensions" [RFC2132]. In this the example states that "site-specific options in DHCP have significance only within a single site". As noted below this became a problem that was rectified in a later revision of DHCP.

Later works identified a need to place a scope on private use namespaces. Another example of private use namespace in the IANA guidelines [RFC2434] {{RFC8126}} is from "STANDARD FOR THE FORMAT OF ARPA INTERNET TEXT MESSAGES" [RFC0822] {{RFC5322}} which describes X-headers. There was no effort made to control their scope, and the use of the namespace was removed when the specification was updated in 2001 in "Internet Message Format" [RFC2822] {{RFC5322}}.

3. Observed Characteristics of Private Use Options

This section summarizes the observed characteristics of some private use options that have been developed and deployed.
3.1. Parts and Identification of the Authority

There appear to be three identifiable parts of private use namespaces:

- Authority
- Path
- Value

3.1.1. Authority

A private use option requires an Authority that can create and maintain the option. Presumably, the goal for an Authority is to regulate, codify, and disambiguate each namespace. Therefore, the referent most often seen has been globally unique, and not dependent upon local interpretation. For example, several vendors have published their RADIUS VSAs on web pages, which are easy to find. From that, anyone creating or updating a RADIUS server would have access to, and be able to incorporate the information available.

Likely, the first private use option with a globally unique source was defined in the "Structure and Identification of Management Information for TCP/IP-based Internets" [RFC1155] which was first used in "A Simple Network Management Protocol" [RFC1067] (SNMP). The globally unique Authority in SNMP is the International Standards Organization [ISO] which is accredited by the United Nations to maintain this structure.

While SNMP used the entire OBJECT IDENTIFIER with the prefix, other protocols truncated this to only used the Private Enterprise Number [IANApen] (PEN) as an identification of an Authority. This reduced the length of the identifier but continued to provide a unique Authority through a globally managed scope.

The PEN is sourced by the Internet Assigned Numbers Authority (IANA). PENs may be viewed as being similar to domain names in that they are acquired by individuals, corporations, or other organizations. However, a notable difference is that when domain names fall into disuse they may be acquired and used by entirely different people or organizations - as per the conditions set forth by the Internet Corporation for Assigned Names and Numbers [ICANN]. The structure of the PEN registry does not place any limits on the time that a PEN will be active or associated with the requester. This is no different from many other registries maintained by the IANA; they are just a snapshot at the time of the reservation based on the information required by the IANA and provided by the applicant. This
eternal association of the PEN, versus the ephemeral association of domain names, has not been shown to present any problems to private use options. This may, in fact, be a feature as this methodology ensures that these namespaces will stay unique for the foreseeable future.

Some additional information on the PEN may be found in "Enterprise Number for Documentation Use" [RFC5612], and in "Private Enterprise Number (PEN) practices and Internet Assigned Numbers Authority (IANA) registration considerations" [I-D.liang-iana-pen].

One observed alternative to using a numerical indicator such as the OBJECT IDENTIFIER or PEN, is to use textual strings such as names. In some cases, domain names have been used for this purpose. However, as noted above, domain names may be more ephemeral than eternal. Unlike PENs that usually become unserviceable when their owning organization ceases operation, domain names that fall into disuse may be acquired and used by entirely different organizations. Similar to the use of PENs however, there have not been any problems reported from this in normal use.

Uniform Resource Names (URNs) have also been used to convey options. They seem to provide flexibility for many different needs. URNs were first defined in "Uniform Resource Names (URN) Namespace Definition Mechanisms" [RFC3406] ([RFC8141]). "An IETF URN Sub-namespace for Registered Protocol Parameters" [RFC3553] provides guidance for ways to use URNs as protocol parameters and how to define an Authority.

### 3.1.2. Path and Value

Once the Authority is established as a globally unique source, an actual option, or in some cases multiple options, may be specified. This has usually been observed to be an indicator of what Value is expected. Within the scope established by the Authority, the Path to each Value has been seen to be unique.

In a very simple example, the namespace of a private use option may consist of "Authority"+"Path"="Value". Since the Authority is unique, each individual Path will be unique as long as the Authority maintains that uniqueness; e.g., it would be poor form for an Authority to define a namespace, then to redefine it in a conflicting way at a later time.
3.2. Incomplete Understanding

Guidance has frequently been provided on how to deal with incomplete understanding when private use options are not understood by a receiver. Within the example protocol specifications given in this discussion, some behavior has usually been established about what to do if the receiver does not understand a namespace. Some protocols have defined that a receiver will silently discard packets that contain private use options they do not understand. Other protocols have defined that they will only discard the private use option rather than the entire packet. On the other hand some other protocols have no need for the receiver to have any understanding of any private use options when it receives any.

In some cases, guidance has given describing appropriate error message responses for incomplete understanding or processing that cannot be performed.

3.3. Bounds and Extensibility

The Values of private use options have frequently followed the same guidance given for standard options in their respective specifications. In most of the examples given, the Value of each private use option has been well defined and bounded.

Private use options may be extensible if they are clearly designed to be so.

3.4. Use and Reuse

In some cases, a unique option may only be used once within the context of an exchange. This may define a Value of an option once and will not change that Value during the remainder of the session. RADIUS and DHCP seem to either state this or strongly imply it. However, while it is not explicitly discussed, it appears that nothing prevents this within Syslog, and it seems to be acceptable behavior to resend unique options multiple times within EPP.

4. Examples of Private Use Options

This section contains a review of RFCs that allow the use of private use options.

4.1. SNMP

SNMP is syntactically complex but has features in the GetRequest PDU that are consistent with the observed characteristics of private use options. The structure of management information (SMI) is currently
defined by the "Structure of Management Information Version 2 (SMIv2)" [RFC2578]. SMI is a well described tree of OBJECT IDENTIFIERS (OIDs). OIDs have an Authority and Path for defined object identifiers which this document describes as standard options. The specification also allows for experimental and vendor specific object identifiers, which are described as private use options in this document. The IANA maintains a registry of these Network Management Parameters [IANAsmi].

As was noted, the globally unique Authority in SNMP is the International Standards Organization [ISO].

The Internet subtree of experimental OBJECT IDENTIFIERS starts with the prefix: 1.3.6.1.3.

The Internet subtree of private enterprise OBJECT IDENTIFIERS starts with the prefix: 1.3.6.1.4.1. and is followed by a Private Enterprise Number [IANApen] (PEN) and then the objects defined by that enterprise. After the vendor identifier (the PEN) in the management information base (MIB), a vendor may create many different trees to identify objects. This may result in a very large number of OBJECT IDENTIFIERS, each of which is an identifier, or Value, of a Path. Each of these are uniquely identified by the vendor and do not require registration with any coordinating authority.

The last part of each OBJECT IDENTIFIER is the Path and a placeholder for its Value; the varbind. In a GetRequest the SNMP Manager (the client) fills the first part of the varbind with the object identifier. The other portion is transmitted with an ASN.1 NULL value. In a typical case, the SNMP Agent (the server) responds by replacing the NULL with the actual Value in the response. Since this namespace is defined by the vendor, it may actually be a concatenation of Values.

The SetRequest PDU is similar to the GetRequest PDU in that it has an OID and may use a PID to identify the objects, however, the varbind is populated differently than in a GetRequest PDU. The other PDUs also use the OID and may use a PID, but behave differently than the GetRequest PID.

The SNMP namespace is extensible. A varbind may be considered to be a TLV wherein the Value may be another TLV.

Specific codes, known as error-indexes, are used to indicate when a request cannot be processed because a device does not understand a request.
GetRequests and SetRequests may be sent repetitively, even with the same Path and with the same or different Values. For GetRequests, a client may be monitoring a server to chronologically record parameters of interest. In some cases, the analysis of the Values obtained by GetRequests may trigger an event that causes one or more SetRequests to be sent.

4.2. RADIUS

There are many attributes defined in "The Remote Authentication Dial In User Service (RADIUS)" [RFC2058] [[RFC2865]], which may be considered to be standard options. Each of these attributes is specified within a "type length value" (TLV) container. For this protocol, the "type" attribute is a specific numerical value, which differentiates it other types.

[ RFC2058 ] documented how to use just the PEN (without the rest of the SMI path to the root) to allow "vendors" to articulate their own options. In that document, these are called Vendor-Specific Attributes (VSA).

One example of a RADIUS standard option is Type 26, which denotes the Vendor Specified Attribute. It is "available to allow vendors to support their own extended Attributes not suitable for general usage". The PEN of the "vendor" is the Authority that starts the namespace. The remainder of the namespace after the PEN is deliberately undefined in the specification. It is practically suggested that the field contain embedded TLVs. This may be seen as the Path and Value.

The values for each RADIUS type are bounded by the length of the attribute. In some cases, it is feasible that a value has no length. In that case, the transmission of the type alone has been seen to be a signal of some sort to the receiver.

The original specification of [RFC2058] [[RFC2865]] provided guidance that invalid packets were to be silently discarded. That was augmented in [RFC2865], along with guidance about reusing the attributes.

o Servers not equipped to interpret the vendor-specific information sent by a client MUST ignore it (although it may be reported).

o Clients which do not receive desired vendor-specific information SHOULD make an attempt to operate without it, although they may do so (and report they are doing so) in a degraded mode.
The Attribute-Specific field is dependent on the vendor’s definition of that attribute.

It SHOULD be encoded as a sequence of vendor type / vendor length / value fields.

Multiple subattributes MAY be encoded within a single Vendor-Specific Attribute, although they do not have to be.

### 4.3. Mobile IP

"Mobile IP Vendor Specific Extensions" [RFC3115] defines two extensions that can be used for making organization specific extensions by vendors/organizations for their own specific purposes for Mobile IP [RFC2002] ([RFC5944]). These are the Critical Vendor/Organization Specific Extension (CVSE) and the Normal Vendor/Organization Specific Extension (NVSE). These are collectively called Vendor/Organization Specific Extensions (VSE).

The structure of the namespace of the VSEs for "Mobile IP" [RFC3115] is similar to that of RADIUS. The PEN is the Authority, and types and values (the Path and Value) may be stacked in TLVs. The values are constrained by the respective lengths of the types or subtypes.

Guidance is given for incomplete understanding in [RFC3115], which is consistent with the guidance given in the original Mobile IP specification [RFC2002] ([RFC5944]).

When the Critical Vendor/Organization Specific Extension (CVSE) is encountered but not recognized, the message containing the extension MUST be silently discarded.

When a Normal Vendor/Organization Specific Extension (NVSE) is encountered but not recognized, the extension SHOULD be ignored, but the rest of the Extensions and message data MUST still be processed.

Error codes are provided in responses to registration requests that are denied because of incomplete understanding.

Multiple TLV’s with the types CVSE-TYPE-NUMBER and NVSE-TYPE-NUMBER can be included in a message. RFC 3115 is silent on reusing the same VSE in subsequent messages.
4.4. DHCP

"Dynamic Host Configuration Protocol" [RFC2131] specified that there was to be a single instance of the vendor type, and the receiver was to use that namespace to set and limit the scope for the fields in the vendor-specific information option. This early version of DHCP did not allow for multiple Authorities; only a single Authority was permitted where the Path and Value were to be defined referring exclusively to that scope. Evidently this was found to be unworkable when different vendors needed to expand private use options in the protocol.

"Dynamic Host Configuration Protocol for IPv6 (DHCPv6)" [RFC3315] ([RFC8415]) was created to provide DHCP for IPv6. This used the PEN as the way to identify the Authority of each private use option. This methodology was subsequently adopted in "Vendor-Identifying Vendor Options for Dynamic Host Configuration Protocol version 4 (DHCPv4)" [RFC3925], which provided for multiple vendors to identify and set their own private use options. TLVs were used in this instance with its inherent bounds and extensibility.

[RFC3925] provides guidance on actions to take if servers and clients do not comprehend a request or a response: servers must ignore options they are not equipped to comprehend and clients should make an attempt to get along without any desired vendor specific response they expect.

[RFC2131] allowed options to be sent only once. However, it acknowledged that multiple values for an option may be transmitted. This may be, for example, for a list of routers where the list is too long to fit within a single option. Guidance is given that the client must concatenate the values into a single list. This sentiment is echoed in [RFC3925], which states that behavior is undefined if a sequence of vendors options reuses the same PEN.

4.5. Syslog

"The Syslog Protocol" [RFC5424] also uses the PEN within structured data (SD) to uniquely qualify the namespace for private use options. The format for options, called SD-ELEMENTs, consists of an SD-ID and SD-PARAMs. For standard options the "@" character cannot be used in the SD-ID. Private use options must have the PEN following the "@" character in the SD-ID. This allows a vendor or experimenter to have disambiguated Paths and Values.

Simply put, a standard option is an SD-ID that does not have the "@" character in it, while a private use option is an SD-ID that does contain the "@" character.
For example, the standard option of the SD-ID timeQuality may only have PARAM-VALUEs of "0" and "1" for the tzKnown PARAM-NAME. The SD-ELEMENT Authority for the standard option timeQuality is then the IANA. However, the SD-ID timeQuality@32473 is a private use option controlled by the Authority that controls enterprise number 32473. Therefore, the tzKnown SD-PARAM may have any PARAM-VALUE assigned to it by the owner of enterprise number 32473.

Syslog transport receivers are supposed to accept all correctly formatted Syslog messages. Unlike RADIUS, the receiving Syslog application does not have to have immediate knowledge of all variable options to continue operations. If a private use option is not immediately known to the receiving application, it may still store the message and an Operator or Administrator may look it up at a later time.

An SD-ID may not be reused within a Syslog message.

Bounds are given in [RFC5424].

4.6. Secure Shell

"The Secure Shell (SSH) Protocol Architecture" [RFC4251] uses character strings rather than PENs to establish Authority. Similar to Syslog, but actually predating it, standard options must not have the "@" character in them. Private use options will have an Authority identifier preceding an "@" character followed by a Value field. For example, in "The Secure Shell (SSH) Connection Protocol" [RFC4254] SSH channels may be opened by specifying a channel type when sending the SSH_MSG_CHANNEL_OPEN message. Standard options for the channel type include "session" and "x11". A private use option for a channel type could be "example_session@example.com".

The character strings are domain names as defined in [RFC1034] and [RFC1035]. This is specified in "The Secure Shell (SSH) Protocol Architecture" [RFC4251]. The rationale for choosing the manner of providing a format for private use options is given in Section 4.2 of [RFC4251].

We have chosen to identify algorithms, methods, formats, and extension protocols with textual names that are of a specific format. DNS names are used to create local Paths and Values where experimental or classified extensions can be defined without fear of conflicts with other implementations.

In the SSH protocol [RFC4250], the Authority is a domain name and the path and value of the option is dependent upon context. For example, ourcipher-cbc@example.com can only be used when negotiating ciphers,
while example_session@example.com can only be used when negotiating channel types, per the examples in [RFC4250].

Guidance is given throughout the SSH series of RFCs (4250 – 4254) for incomplete understanding. The guidance differs based upon the context; in some cases, the guidance is to ignore a private use option when it cannot be understood, while in other cases, a negative response must be sent to indicate that a received private use option could not be understood.

Similarly, reuse of a private use option is dependent upon the context. The same is true for checking bounds of any private use option.

4.7. YANG and NETCONF

One example of a protocol utilizing URNs is "Network Configuration Protocol (NETCONF)" [RFC6241]. NETCONF may utilize "YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)" [RFC6020] as a means to describe and convey options.


Section 8.3 of YANG [RFC6020] describes the parsing of the YANG payload. It contains a good deal of information about how to process elements or values that are not recognized.

Similarly, NETCONF [RFC6241] contains much information about processing requests that cannot be completed because elements or values are not recognized.

Both YANG [RFC6020] and NETCONF [RFC6241] use URIs to enumerate private use options of a device. The use of this comes from XPATH [W3C.REC-xpath-19991116]. In both of these, the start of authority is the domain name in the URI and the Authority is the full URI path. Many private use options may be described within YANG. From that, each private use option may be populated in NETCONF.

4.8. Extensible Provisioning Protocol

The "Extensible Provisioning Protocol (EPP)" [RFC5730] is another example of a protocol that utilizes URN Path and Values. From the Protocol Description section 2:
EPP uses XML namespaces to provide an extensible object management framework and to identify schemas required for XML instance parsing and validation. These namespaces and schema definitions are used to identify both the base protocol schema and the schemas for managed objects.

The specification provides clear guidance and an example on how to extend the base protocol and how to map new objects through the use of separate documents. However, commands and responses may be extended through the use of an <extension> element. In this protocol, and the extensions, the start of authority is the domain name in the URI and the focus is the full URI path.

Guidance has been provided about incomplete understanding. First, a section is provided on responses for received messages that are not understandable, are beyond boundaries, or are not in compliance with policy. Additionally, guidance is given about incomplete understanding of a response:

Command success or failure MUST NOT be assumed if no response is returned or if a returned response is malformed. Protocol idempotency ensures the safety of retrying a command in cases of response-delivery failure.


5. Observations

Private use options are a way to allow vendors, network operators, and experimenters to convey dynamic information without going through any process to register each variable. However, there is no one size fits all. The use of a very specific and fixed format works for RADIUS which requires speed in processing. On the other hand, the open nature of the private use options in Syslog are appropriate for that protocol where all event messages need not be fully parsed at the time of reception.

As with all good things, the use of private use options comes with a cost. Adding any extra fields to a protocol will require additional processing for both the sender and the receiver. Also, larger packets will take up more bandwidth in transmission. In another aspect, the code needed to deal with private use options may be considered wasteful if it is not used.
Clear documentation has been seen to achieve uniformity and interoperability in these features. Obviously implementers will need to adhere closely to these standards for complete interoperability.

6. Authoritative Guidance

This document is not an encouragement or recommendation to define private use fields in IETF protocols. Rather, since private use options are being used by the community, this document is an attempt to document the ways in which they have been used.

However, "Design Considerations for Protocol Extensions" [RFC6709] is intended to provide guidance on designing protocol extensions. It has several examples and pointers to other material that will aid in the development of protocol extensions.

"Procedures for Protocol Extensions and Variations" [RFC4775] is a companion document to [RFC6709] and provides the procedures for review and standardization for extensions to be added to protocols.

Finally, the usage of any private use values on the wire before any namespace is properly reserved with the IANA is entirely inadvisable.

7. Authors Notes

This section will be removed prior to publication.

This is version -18. I have received ISE feedback and am integrating that into this document. Unfortunately, that’s going to take a while as life and the day job keep getting in the way.

I’m revising the flow of the document to be consistent and to accomodate the feedback that I’ve received.

8. Security Considerations

This document reviews ways that options are being used in various protocols. As such, there are no security considerations inherent in this document.

While it is not a problem that can be technically addressed, fraudulently purporting to be an owner of a domain name, a PEN, or other identifier may allow the misuse of private namespaces.

Readers and implementers should be aware of the context of implementing options in protocols they are using or that are being developed.
9. IANA Considerations

This document does not propose a standard and does not require the IANA to do anything.

10. Acknowledgments

The idea for documenting this came from questions asked in the SIP-CLF Working Group and the author is grateful for the discussion around this topic.

The following people have contributed to this document. Listing their names here does not mean that they agree with or endorse the document, but that they have contributed to its substance:

David Harrington, Dan Romascanu, Bert Wijnen, Ralph Droms, Juergen Schoenwalder, Nevil Brownlee, Klaas Wierenga, Brian Carpenter, Randy Presuhn, and Dave Crocker.

11. References

11.1. Normative References


11.2. Informative References


Author’s Address

Chris Lonvick
1307 Kent Oak Dr.
Houston, Texas 77077
US

Email: lonvick.ietf@gmail.com