Workgroup: Network Working Group
Internet-Draft:
draft-ietf-regext-rdap-redacted-06
Published: 13 May 2022
Intended Status: Standards Track
Expires: 14 November 2022
Authors: J.G. Gould
VeriSign, Inc.
D.S. Smith
VeriSign, Inc.
J.K. Kolker
GoDaddy Inc.
R.C. Carney
GoDaddy Inc.

Redacted Fields in the Registration Data Access Protocol (RDAP) Response

Abstract

This document describes an RDAP extension for explicitly identifying redacted RDAP response fields, using JSONPath as the default expression language.

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 14 November 2022.

Copyright Notice

Copyright (c) 2022 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 Revised BSD License text as described in

Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Revised BSD License.

Table of Contents

1. Introduction
2. Conventions Used in This Document
3. Redaction Methods
   3.1. Redaction by Removal Method
   3.2. Redaction by Empty Value Method
   3.3. Redaction by Replacement Value Method
4. Redacted RDAP Response
   4.1. RDAP Conformance
   4.2. "redacted" Member
5. JSONPath Considerations
6. IANA Considerations
   6.1. RDAP Extensions Registry
   6.2. JSON Values Registry
7. Implementation Status
   7.1. IIT-CNR/Registro.it RDAP Server
8. Security Considerations
9. Acknowledgements
10. References
   10.1. Informative References
   10.2. Normative References
Appendix A. Change History
   A.1. Change from 00 to 01
   A.2. Change from 01 to 02
   A.3. Change from 02 to 03
   A.4. Change from 03 to 04
   A.5. Change from 04 to 05
   A.6. Change from 05 to 06
Authors' Addresses

1. Introduction

This document describes an RDAP extension for explicitly identifying redacted RDAP response fields, using JSONPath as the default expression language. A redacted RDAP field is one that has data removed from the RDAP response due to the lack of client privilege to receive the field. This extension can be used to identify redacted RDAP fields in any RDAP object class, as defined in [RFC9083], or RDAP fields defined in RDAP extensions. Because an RDAP response may exclude a field due to either the lack of data or based on the lack of RDAP client privileges, this extension is used to explicitly specify which RDAP fields are not included in the RDAP response due to redaction. It thereby provides a capability for disambiguation between redaction and possible other reasons for data or field absence.

In [RFC9082] RDAP supports both lookup and search queries, where a lookup query responds with a single object and a search query responds with a list of objects. This document applies to redaction of a single object of a lookup response and in each of the objects of a search response.

JSONPath, as defined in [I-D.ietf-jsonpath-base], is used as the default expression language to reference RDAP fields that have been redacted. The redacted JSON fields will either be removed or have empty values in the RDAP response. JSON is defined by [RFC8259].

2. Conventions Used in This Document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

The JSON examples include extra line breaks and whitespace. For instance, the JSONPath expressions are broken out into multiple lines when required for illustration.

3. Redaction Methods

Redaction in RDAP can be handled in multiple ways. The use of placeholder text for the values of the RDAP fields, such as the placeholder text "XXXX", MUST NOT be used for redaction. A placeholder text value will not match the format requirements of each of the RDAP fields and provides an inconsistent and unreliable redaction signal. This section covers the redaction methods that can be used with the redaction signaling defined in Section 4.2.

RDAP responses, as defined in [RFC9083], include a mix of JSON objects and JSON arrays, where JSON arrays are heavily used for entity objects with jCard [RFC7095]. jCard [RFC7095] is a JSON representation of vCard [RFC6350] that inherits its dependency on arrays. An example is the vCard [RFC6350] "ADR" property / jCard [RFC7095] "adr" property that defines a sequence of address components. According to [RFC6350], when an "ADR" property component value is missing, the associated component separator MUST still be specified. jCard [RFC7095] extends the use of arrays with each individual vCard property being represented by an array of three fixed elements, followed by one or more additional elements. The mix of JSON objects and JSON arrays impacts the methods used for redaction in RDAP.

The redaction of RDAP fields fall into the two categories of Redaction by Removal Method (Section 3.1) and Redaction by Empty Value Method (Section 3.2), defined in the following sub-sections.

3.1. Redaction by Removal Method

The Redaction by Removal Method is when the RDAP field is removed from the RDAP response, which is the preferred method. The Redaction by Removal Method can be done for all RDAP response fields other than response fields using the position in an array to signal the redacted field (e.g., the JSON arrays used with jCard [RFC7095]). RDAP extensions such as JSContact in Registration Data Access Protocol (RDAP) JSON Responses [I-D.ietf-regext-rdap-jscontact] do not have a dependency on the use of positional JSON arrays and are therefore suited for the Redaction by Removal Method.

When an RDAP object is redacted by removal, all of the RDAP object's child fields are also removed. Only the redacted RDAP object needs to be referenced in the list of redacted fields, as defined in Section 4.2. An example of redacting an RDAP object is removing the administrative contact from the RDAP response and including the following "redacted" member:

```
"redacted": [
  {
    "name": {
      "type": "Administrative Contact"
    },
    "path": "$.entities[?(@.roles[0]=='administrative')]",
    "method": "removal"
  }
]
```

Figure 1: Redacted Administrative Contact

The Redaction by Removal Method MUST NOT be used to remove a field using the position in a fixed length array to signal the redacted field. For example, removal of an individual data field in jCard [RFC7095] will result in a non-conformant jCard [RFC7095] array definition.

3.2. Redaction by Empty Value Method

The Redaction by Empty Value Method is when a redacted field is not removed, but its value is set to an empty value, such as "" for a jCard [RFC7095] Text ("text") property or null for non-Text ("text") properties. The empty jCard [RFC7095] values ("" or null) are referenced in the "redacted" member in place of the jCard [RFC7095] property name, such as referencing the "fn" jCard property value at position 3 instead of referencing the "fn" jCard property name at position 0. The Redaction by Empty Value Method SHOULD be used only when redacting JSON response fields that use the position in an array to signal the redacted field (e.g., jCard [RFC7095] arrays).

Optional jCard [RFC7095] properties SHOULD use the Redaction by Removal Method (Section 3.1) to redact the entire property. The required jCard [RFC7095] "fn" property, defined in section 6.2.1 of vCard [RFC6350], MUST use the Redaction by Empty Value Method to redact the property value. Removing the "fn" property would violate vCard [RFC6350] and removing the property value would violate the fixed array positions defined in jCard [RFC7095].

An example of the redacted "fn" jCard property using the Redaction by Empty Value Method:

```
[
  "fn",
  {},
  "text",
  ""
]
```

Figure 2: Redacted "fn" jCard property using Redaction by Empty Value Method

An example of the "redacted" member for the redacted "fn" jCard property value, which is array position 3:

```
"redacted": [
  {
    "name": {
      "type": "Registrant Name"
    },
    "path": "$.entities[?(@.roles[0]=='registrant')].
      vcardArray[1][?(@[0]=='fn')][3]",
    "pathLang": "jsonpath",
    "method": "emptyValue",
    "reason": {
      "type": "Server policy"
    }
  }
]
```

Figure 3: Redacted Registrant Name using Array Position

3.3. Redaction by Replacement Value Method

The Redaction by Replacement Value Method is when a redacted field is not removed, but its value is replaced with a different value, such as protecting the "email" jCard [RFC7095] property value with an anonymized email "text" value or the use of an alternate "uri" value to a web form. Replacing a property value is a form of

redaction, since it protects the true property value for privacy reasons.

An example of the redacted "email" jCard property using the Redaction by Replacement Value Method with an anonymized email:

```
[
  "email",
  {},
  "text",
  "anonymized123@example.com"
]
```

Figure 4: Redacted "email" jCard property using Redaction by Replacement Value Method with an anonymized email

An example of the "redacted" member for the redacted registrant "email" jCard property value with an anonymized "text" value.

```
"redacted": [
  {
    "name": {
      "type": "Registrant Email"
    },
    "path": "$.entities[?(@.roles[0]=='registrant')].
               vcardArray[1][?(@[0]=='email')][3]",
    "pathLang": "jsonpath",
    "method": "replacementValue",
  }
]
```

Figure 5: Redacted Email using Replacement Value with an anonymized "text" value

An example of the redacted "email" jCard property using the Redaction by Replacement Value Method with a [RFC8605] "contact-uri" jCard property to a web form:

```
[
  "contact-uri",
  {},
  "uri",
  "https://email.example.com/123"
]
```

Figure 6: Redacted "email" jCard property using Redaction by Replacement Value Method with a "contact-uri" jCard property to a web form

An example of the "redacted" member for the redacted registrant "email" jCard property value with a [RFC8605] "contact-uri" jCard property to a web form:

```
"redacted": [
  {
    "name": {
      "type": "Registrant Email"
    },
    "path": "$.entities[?(@.roles[0]=='registrant')].
              vcardArray[1][?(@[0]=='contact-uri')][3]",
    "pathLang": "jsonpath",
    "method": "replacementValue",
  }
]
```

Figure 7: Redacted Email using Replacement Value with a "contact-uri" jCard property to a web form

4. Redacted RDAP Response

4.1. RDAP Conformance

RDAP responses that contain values described in this document MUST indicate conformance with this specification by including an "rdapConformance" ([RFC9083]) value of "redacted_level_0_3". The "redacted_level_0_3" extension identifier is described in Section 6.1.

Example "rdapConformance" member with the redacted extension:

```
"rdapConformance": [
  "rdap_level_0",
  "redacted_level_0_3"
]
```

Figure 8: "rdapConformance" with Redacted Extension

4.2. "redacted" Member

The "redacted" member MUST be added to the RDAP response when there are redacted fields. The "redacted" member is included as a member of the object class in a lookup response, such as the object classes defined in [RFC9083], and as a member of the object instances in a

search response, such as the object instances defined in [RFC9083]. The "redacted" member contains an array of redacted objects with the following child members:

"name": A logical name for the redacted field. The logical name used for the redacted field is up to server policy. The logical name is defined using an object with a "type" field denoting a registered redacted name (see Section 6.2) or a "description" field denoting an unregistered redacted name. The registered redacted names and the chosen unregistered names can meet the needs of different RDAP services or industries.

"path": The JSON expression of the redacted field, using the expression language defined by the "pathLang" member. The JSON expression references a removed JSON field or an empty field value based on Section 3.

"pathLang": OPTIONAL JSON path expression language used, with the default value of "jsonpath" for JSONPath ([I-D.ietf-jsonpath-base]). Other JSON path expression languages MAY be used based on server policy.

"method": OPTIONAL redaction method used with "removal" indicating the Redaction By Removed Method (Section 3.1), "emptyValue" indicating the Redaction by Empty Value Method (Section 3.2), and "replacementValue" indicating the Redaction by Replacement Value Method (Section 3.3). The default value is "removal" when not provided.

"reason": OPTIONAL human readable reason(s) for the redacted field in the language defined by the [RFC9083] "lang" member. The default language is "en" if the [RFC9083] "lang" member is not specified. The reason is defined using an object with an OPTIONAL "type" field denoting a registered redacted reason (see see Section 6.2) and an OPTIONAL "description" field denoting an unregistered redacted reason. The "description" field MUST NOT be a client processing dependency.

Example unredacted version of an RDAP lookup response:

```
{
  "rdapConformance": [
    "rdap_level_0"
  ],
  "objectClassName": "domain",
  "handle": "ABC123",
  "ldhName": "example.com",
  "secureDNS": {
    "delegationSigned": false
  },
  "notices": [
    {
      "title": "Terms of Use",
      "description": [
        "Service subject to Terms of Use."
      ],
      "links": [
        {
          "rel": "self",
          "href": "https://www.example.com/terms-of-use",
          "type": "text/html",
          "value": "https://www.example.com/terms-of-use"
        }
      ]
    }
  ],
  "nameservers": [
    {
      "objectClassName": "nameserver",
      "ldhName": "ns1.example.com"
    },
    {
      "objectClassName": "nameserver",
      "ldhName": "ns2.example.com"
    }
  ],
  "entities": [
    {
      "objectClassName": "entity",
      "handle": "123",
      "roles": [
        "registrar"
      ],
      "publicIds": [
        {
          "type": "IANA Registrar ID",
          "identifier": "1"
        }
      ],
```

```
"vcardArray": [
  "vcard",
  [
    [
      "version",
      {},
      "text",
      "4.0"
    ],
    [
      "fn",
      {},
      "text",
      "Example Registrar Inc."
    ],
    [
      "adr",
      {},
      "text",
      [
        "",
        "Suite 100",
        "123 Example Dr.",
        "Dulles",
        "VA",
        "20166-6503",
        "US"
      ]
    ],
    [
      "email",
      {},
      "text",
      "contact@organization.example"
    ],
    [
      "tel",
      {
        "type": "voice"
      },
      "uri",
      "tel:+1.7035555555;ext=1234"
    ],
    [
      "tel",
      {
        "type": "fax"
      },
      "uri",
```

```
        "tel:+1.7035555556"
      ]
    ]
  ],
  "entities": [
    {
      "objectClassName": "entity",
      "roles": [
        "abuse"
      ],
      "vcardArray": [
        "vcard",
        [
          [
            "version",
            {},
            "text",
            "4.0"
          ],
          [
            "fn",
            {},
            "text",
            "Abuse Contact"
          ],
          [
            "email",
            {},
            "text",
            "abuse@organization.example"
          ],
          [
            "tel",
            {
              "type": "voice"
            },
            "uri",
            "tel:+1.7035555555;ext=1234"
          ]
        ]
      ]
    }
  ]
},
{
  "objectClassName": "entity",
  "handle": "XXXX",
  "roles": [
    "registrant"
```

```
],
"vcardArray": [
  "vcard",
  [
    [
      "version",
      {},
      "text",
      "4.0"
    ],
    [
      "fn",
      {},
      "text",
      "Registrant User"
    ],
    [
      "org",
      {},
      "text",
      "Example Inc."
    ],
    [
      "adr",
      {},
      "text",
      [
        "",
        "Suite 1235",
        "4321 Rue Somewhere",
        "Quebec",
        "QC",
        "G1V 2M2",
        "Canada"
      ]
    ],
    [
      "email",
      {},
      "text",
      "registrant.user@example.com"
    ],
    [
      "tel",
      {
        "type": "voice"
      },
      "uri",
      "tel:+1-555-555-1235;ext=123"
```

```
      ],
      [
        "tel",
        {
          "type": "fax"
        },
        "uri",
        "tel:+1-555-555-5321"
      ]
    ]
  ]
},
{
  "objectClassName": "entity",
  "handle": "YYYY",
  "roles": [
    "technical"
  ],
  "vcardArray": [
    "vcard",
    [
      [
        "version",
        {},
        "text",
        "4.0"
      ],
      [
        "fn",
        {},
        "text",
        "Technical User"
      ],
      [
        "org",
        {},
        "text",
        "Example Inc."
      ],
      [
        "adr",
        {},
        "text",
        [
          "",
          "Suite 1234",
          "4321 Rue Somewhere",
          "Quebec",
          "QC",
```

```
          "G1V 2M2",
          "Canada"
        ]
      ],
      [
        "email",
        {},
        "text",
        "technical.user@example.com"
      ],
      [
        "tel",
        {
          "type": "voice"
        },
        "uri",
        "tel:+1-555-555-1234;ext=321"
      ],
      [
        "tel",
        {
          "type": "fax"
        },
        "uri",
        "tel:+1-555-555-4321"
      ]
    ]
  ]
},
{
  "objectClassName": "entity",
  "handle": "ZZZZ",
  "roles": [
    "administrative"
  ],
  "vcardArray": [
    "vcard",
    [
      [
        "version",
        {},
        "text",
        "4.0"
      ],
      [
        "fn",
        {},
        "text",
        "Administrative User"
```

```
      ],
      [
        "org",
        {},
        "text",
        "Example Inc."
      ],
      [
        "adr",
        {},
        "text",
        [
          "",
          "Suite 1236",
          "4321 Rue Somewhere",
          "Quebec",
          "QC",
          "G1V 2M2",
          "Canada"
        ]
      ],
      [
        "email",
        {},
        "text",
        "administrative.user@example.com"
      ],
      [
        "tel",
        {
          "type": "voice"
        },
        "uri",
        "tel:+1-555-555-1236;ext=789"
      ],
      [
        "tel",
        {
          "type": "fax"
        },
        "uri",
        "tel:+1-555-555-6321"
      ]
    ]
  ]
},
{
  "objectClassName": "entity",
  "handle": "WWWW",
```

```
      "roles": [
        "billing"
      ],
      "vcardArray": [
        "vcard",
        [
          [
            "version",
            {},
            "text",
            "4.0"
          ],
          [
            "fn",
            {},
            "text",
            "Billing User"
          ],
          [
            "email",
            {},
            "text",
            "billing.user@example.com"
          ]
        ]
      ]
    }
  ],
  "events": [
    {
      "eventAction": "registration",
      "eventDate": "1997-06-03T00:00:00Z"
    },
    {
      "eventAction": "last changed",
      "eventDate": "2020-05-28T01:35:00Z"
    },
    {
      "eventAction": "expiration",
      "eventDate": "2021-06-03T04:00:00Z"
    }
  ],
  "status": [
    "server delete prohibited",
    "server update prohibited",
    "server transfer prohibited",
    "client transfer prohibited"
  ]
}
```

Figure 9: Unredacted RDAP Lookup Response

Example redacted version of an RDAP lookup response:

```
{
  "rdapConformance": [
    "rdap_level_0",
    "redacted_level_0_3"
  ],
  "objectClassName": "domain",
  "ldhName": "example.com",
  "secureDNS": {
    "delegationSigned": false
  },
  "notices": [
    {
      "title": "Terms of Use",
      "description": [
        "Service subject to Terms of Use."
      ],
      "links": [
        {
          "rel": "self",
          "href": "https://www.example.com/terms-of-use",
          "type": "text/html",
          "value": "https://www.example.com/terms-of-use"
        }
      ]
    }
  ],
  "nameservers": [
    {
      "objectClassName": "nameserver",
      "ldhName": "ns1.example.com"
    },
    {
      "objectClassName": "nameserver",
      "ldhName": "ns2.example.com"
    }
  ],
  "entities": [
    {
      "objectClassName": "entity",
      "handle": "123",
      "roles": [
        "registrar"
      ],
      "publicIds": [
        {
          "type": "IANA Registrar ID",
          "identifier": "1"
        }
      ],
```

```
"vcardArray": [
  "vcard",
  [
    [
      "version",
      {},
      "text",
      "4.0"
    ],
    [
      "fn",
      {},
      "text",
      "Example Registrar Inc."
    ],
    [
      "adr",
      {},
      "text",
      [
        "",
        "Suite 100",
        "123 Example Dr.",
        "Dulles",
        "VA",
        "20166-6503",
        "US"
      ]
    ],
    [
      "email",
      {},
      "text",
      "contact@organization.example"
    ],
    [
      "tel",
      {
        "type": "voice"
      },
      "uri",
      "tel:+1.7035555555"
    ],
    [
      "tel",
      {
        "type": "fax"
      },
      "uri",
```

```
        "tel:+1.7035555556"
      ]
    ]
  ],
  "entities": [
    {
      "objectClassName": "entity",
      "roles": [
        "abuse"
      ],
      "vcardArray": [
        "vcard",
        [
          [
            "version",
            {},
            "text",
            "4.0"
          ],
          [
            "fn",
            {},
            "text",
            "Abuse Contact"
          ],
          [
            "email",
            {},
            "text",
            "abuse@organization.example"
          ],
          [
            "tel",
            {
              "type": "voice"
            },
            "uri",
            "tel:+1.7035555555"
          ]
        ]
      ]
    }
  ]
},
{
  "objectClassName": "entity",
  "handle": "XXXX",
  "roles": [
    "registrant"
```

```
  ],
  "vcardArray": [
    "vcard",
    [
      [
        "version",
        {},
        "text",
        "4.0"
      ],
      [
        "fn",
        {},
        "text",
        ""
      ],
      [
        "adr",
        {},
        "text",
        [
          "",
          "",
          "",
          "",
          "QC",
          "",
          "Canada"
        ]
      ]
    ]
  ]
},
{
  "objectClassName": "entity",
  "handle": "YYYY",
  "roles": [
    "technical"
  ],
  "vcardArray": [
    "vcard",
    [
      [
        "version",
        {},
        "text",
        "4.0"
      ],
      [
```

```
          "fn",
          {},
          "text",
          ""
        ],
        [
          "org",
          {},
          "text",
          "Example Inc."
        ],
        [
          "adr",
          {},
          "text",
          [
            "",
            "Suite 1234",
            "4321 Rue Somewhere",
            "Quebec",
            "QC",
            "G1V 2M2",
            "Canada"
          ]
        ]
      ]
    ]
  }
],
"events": [
  {
    "eventAction": "registration",
    "eventDate": "1997-06-03T00:00:00Z"
  },
  {
    "eventAction": "last changed",
    "eventDate": "2020-05-28T01:35:00Z"
  },
  {
    "eventAction": "expiration",
    "eventDate": "2021-06-03T04:00:00Z"
  }
],
"status": [
  "server delete prohibited",
  "server update prohibited",
  "server transfer prohibited",
  "client transfer prohibited"
],
```

```
"redacted": [
  {
    "name": {
      "type": "Registry Domain ID"
    },
    "path": "$.handle",
    "pathLang": "jsonpath",
    "method": "removal",
    "reason": {
      "type": "Server policy"
    }
  },
  {
    "name": {
      "type": "Registrant Name"
    },
    "path": "$.entities[?(@.roles[0]=='registrant')].
      vcardArray[1][?(@[0]=='fn')][3]",
    "pathLang": "jsonpath",
    "method": "emptyValue",
    "reason": {
      "type": "Server policy"
    }
  },
  {
    "name": {
      "type": "Registrant Organization"
    },
    "path": "$.entities[?(@.roles[0]=='registrant')].
      vcardArray[1][?(@[0]=='org')]",
    "pathLang": "jsonpath",
    "method": "removal",
    "reason": {
      "type": "Server policy"
    }
  },
  {
    "name": {
      "type": "Registrant Street"
    },
    "path": "$.entities[?(@.roles[0]=='registrant')].
      vcardArray[1][?(@[0]=='adr')][3][:3]",
    "pathLang": "jsonpath",
    "method": "emptyValue",
    "reason": {
      "type": "Server policy"
    }
  },
  {
```

```
    "name": {
      "type": "Registrant City"
    },
    "path": "$.entities[?(@.roles[0]=='registrant')].
      vcardArray[1][?(@[0]=='adr')][3][3]",
    "pathLang": "jsonpath",
    "method": "emptyValue",
    "reason": {
      "type": "Server policy"
    }
  },
  {
    "name": {
      "type": "Registrant Postal Code"
    },
    "path": "$.entities[?(@.roles[0]=='registrant')].
      vcardArray[1][?(@[0]=='adr')][3][5]",
    "pathLang": "jsonpath",
    "method": "emptyValue",
    "reason": {
      "type": "Server policy"
    }
  },
  {
    "name": {
      "type": "Registrant Email"
    },
    "path": "$.entities[?(@.roles[0]=='registrant')].
      vcardArray[1][?(@[0]=='email')]",
    "method": "removal",
    "reason": {
      "type": "Server policy"
    }
  },
  {
    "name": {
      "type": "Registrant Phone"
    },
    "path": "$.entities[?(@.roles[0]=='registrant')].
      vcardArray[1][?(@[1].type=='voice')]",
    "method": "removal",
    "reason": {
      "type": "Server policy"
    }
  },
  {
    "name": {
      "type": "Technical Name"
    },
```

```
  "path": "$.entities[?(@.roles[0]=='technical')].
    vcardArray[1][?(@[0]=='fn')][3]",
  "method": "emptyValue",
  "reason": {
    "type": "Server policy"
  }
},
{
  "name": {
    "type": "Technical Email"
  },
  "path": "$.entities[?(@.roles[0]=='technical')].
    vcardArray[1][?(@[0]=='email')]",
  "method": "removal",
  "reason": {
    "type": "Server policy"
  }
},
{
  "name": {
    "type": "Technical Phone"
  },
  "path": "$.entities[?(@.roles[0]=='technical')].
    vcardArray[1][?(@[1].type=='voice')]",
  "method": "removal",
  "reason": {
    "type": "Server policy"
  }
},
{
  "name": {
    "type": "Technical Fax"
  },
  "path": "$.entities[?(@.roles[0]=='technical')].
    vcardArray[1][?(@[1].type=='fax')]",
  "reason": {
    "type": "Client request",
    "description": "Client requested the field redacted"
  }
},
{
  "name": {
    "description": "Administrative Contact"
  },
  "path": "$.entities[?(@.roles[0]=='administrative')]",
  "method": "removal",
  "reason": {
    "description": "Refer to the technical contact"
  }
```

```
    }
    {
      "name": {
        "description": "Billing Contact"
      },
      "path": "$.entities[?(@.roles[0]=='billing')]",
      "method": "removal",
      "reason": {
        "description": "Refer to the registrant contact"
      }
    }
  ]
}
```

Figure 10: Redacted RDAP Lookup Response

Example unredacted version of an RDAP search response:

```
{
  "rdapConformance": [
    "rdap_level_0"
  ],
  "domainSearchResults":[
    {
      "objectClassName": "domain",
      "handle": "ABC121",
      "ldhName": "example1.com",
      "links":[
        {
          "value":"https://example.com/rdap/domain/example1.com",
          "rel":"self",
          "href":"https://example.com/rdap/domain/example1.com",
          "type":"application/rdap+json"
        },
        {
          "value":"https://example.com/rdap/domain/example1.com",
          "rel":"related",
          "href":"https://example.com/rdap/domain/example1.com",
          "type":"application/rdap+json"
        }
      ]
    },
    {
      "objectClassName": "domain",
      "handle": "ABC122",
      "ldhName": "example2.com",
      "links":[
        {
          "value":"https://example.com/rdap/domain/example2.com",
          "rel":"self",
          "href":"https://example.com/rdap/domain/example2.com",
          "type":"application/rdap+json"
        },
        {
          "value":"https://example.com/rdap/domain/example2.com",
          "rel":"related",
          "href":"https://example.com/rdap/domain/example2.com",
          "type":"application/rdap+json"
        }
      ]
    }
  ]
}
```


Figure 11: Unredacted RDAP Search Response

Example redacted version of an RDAP search response:

```
{
  "rdapConformance": [
    "rdap_level_0",
    "redacted_level_0_3"
  ],
  "domainSearchResults":[
    {
      "objectClassName": "domain",
      "ldhName": "example1.com",
      "links":[
        {
          "value":"https://example.com/rdap/domain/example1.com",
          "rel":"self",
          "href":"https://example.com/rdap/domain/example1.com",
          "type":"application/rdap+json"
        },
        {
          "value":"https://example.com/rdap/domain/example1.com",
          "rel":"related",
          "href":"https://example.com/rdap/domain/example1.com",
          "type":"application/rdap+json"
        }
      ],
      "redacted": [
        {
          "name": {
            "type": "Registry Domain ID"
          },
          "path": "$.domainSearchResults[0].handle",
          "pathLang": "jsonpath",
          "method": "removal",
          "reason": {
            "type": "Server policy"
          }
        }
      ]
    },
    {
      "objectClassName": "domain",
      "ldhName": "example2.com",
      "links":[
        {
          "value":"https://example.com/rdap/domain/example2.com",
          "rel":"self",
          "href":"https://example.com/rdap/domain/example2.com",
          "type":"application/rdap+json"
        },
        {
          "value":"https://example.com/rdap/domain/example2.com",
```

```
          "rel":"related",
          "href":"https://example.com/rdap/domain/example2.com",
          "type":"application/rdap+json"
        }
      ],
      "redacted": [
        {
          "name": {
            "type": "Registry Domain ID"
          },
          "path": "$.domainSearchResults[1].handle",
          "pathLang": "jsonpath",
          "method": "removal",
          "reason": {
            "type": "Server policy"
          }
        }
      ]
    }
  ]
}
```

Figure 12: Redacted RDAP Search Response

5. JSONPath Considerations

JSONPath [I-D.ietf-jsonpath-base] is the default JSON path expression language. This section covers considerations for servers using [I-D.ietf-jsonpath-base] to identify redacted RDAP fields with the "path" member of redacted objects in the "redacted" member. The list of JSONPath considerations include:

1. Use absolute paths with the '$' JSONPath element. An example is "$.handle" for the "Registry Domain ID" in a lookup response or "$.domainSearchResults[0].handle" in a search response.
2. Validate a JSONPath expression using a non-redacted RDAP response, where evaluating the expression results in returning the redacted field.
3. Reference the removed object field when redacting an entire object by the Redaction by Removal Method (Section 3.1), where all of the object's child fields are explicitly removed. An example is "$.entities[?(@.roles[0]=='administrative')]" for the entire "Administrative Contact".
4. When an entity has multiple roles, include "redacted" members for each role using the role index. This will result in duplicate "redacted" members, but will enable the client to treat redaction consistently when there is a single role per entity or multiple roles per entity. An example is when the "roles" member has the value '["registrant","administrative"]',

redacting the "name" member of the entity will result in two "redacted" members with the JSONPath expressions "$.entities[?(@.roles[0]=='registrant')].vcardArray[1][?(@[0]=='fn')][3]" and "$.entities[?(@.roles[1]=='administrative')].vcardArray[1][?(@[0]=='fn')][3]".

5. When there are multiple entities with the same role, include "redacted" members for each entity using the entity index instead of the role. A JSONPath can be created that identifies the entity based on an index of a role selector nodelist, such as "$.entities[?(@.roles[0]=='technical')][0]" for the first entity with the "technical" role. Using the entity index, such as "$.entities[1]", is simpler and recommended.
6. Reference the removed field when using the Redaction by Removal Method (Section 3.1). An example is "$.handle" for the "Registry Domain ID".
7. Reference index 0 of the jCard [RFC7095] property array, which is the jCard [RFC7095] "name" property, with a filter expression containing the name of the field, when redacting a jCard [RFC7095] field using the Redaction by Removal Method (Section 3.1). An example is "$.entities[?(@.roles[0]=='registrant')].vcardArray[1][?(@[0]=='email')]" for the "Registrant Email".
8. Reference jCard [RFC7095] field value or values redacted by array index 3 and greater, when redacting a jCard [RFC7095] field using the Redaction by Empty Value Method (Section 3.2). The jCard [RFC7095] property array index 3 and greater contain the property values, where the property values set with an empty value are referenced directly in place of the jCard [RFC7095] property name. Servers can then systematically redact jCard [RFC7095] field value or values based on the JSONPath expressions and clients will directly know which jCard [RFC7095] property values have been redacted. An example is "$.entities[?(@.roles[0]=='registrant')].vcardArray[1][?(@[0]=='fn')][3]" for the "Registrant Name" or "$.entities[?(@.roles[0]=='registrant')].vcardArray[1][?(@[0]=='adr')][3][5]" for the "Registrant Postal Code".
9. RDAP extensions should define any special JSONPath considerations required to identify redacted RDAP fields if these considerations are insufficient.

6. IANA Considerations

6.1. RDAP Extensions Registry

IANA is requested to register the following value in the RDAP Extensions Registry:

Extension identifier: redacted_level_0_3
Registry operator: Any

Published specification: This document.
Contact: IESG <iesg@ietf.org>
Intended usage: This extension identifier is used for an "rdapConformance" value when returning the "redacted" member in the JSON response.
Extension identifier: redacted
Registry operator: Any
Published specification: This document.
Contact: IESG <iesg@ietf.org>
Intended usage: This extension prefix identifier is used for the "redacted" member returned in the JSON response.

6.2. JSON Values Registry

Section 10.2 of [RFC9083] defines the JSON Values Registry with pre-defined Type field values and the use of the "Expert Review" policy defined in [RFC8126]. Two new JSON Values Registry Type field values are used to register pre-defined redacted name and reason values:

"redacted name": Redacted name being registered. The registered redacted name is referenced using the "type" field of the redacted "name" field.

"redacted reason": Redacted reason being registered. The registered redacted reason is referenced using the "type" field of the redacted "reason" field.

7. Implementation Status

Note to RFC Editor: Please remove this section and the reference to RFC 7942 [RFC7942] before publication.

This section records the status of known implementations of the protocol defined by this specification at the time of posting of this Internet-Draft, and is based on a proposal described in RFC 7942 [RFC7942]. The description of implementations in this section is intended to assist the IETF in its decision processes in progressing drafts to RFCs. Please note that the listing of any individual implementation here does not imply endorsement by the IETF. Furthermore, no effort has been spent to verify the information presented here that was supplied by IETF contributors. This is not intended as, and must not be construed to be, a catalog of available implementations or their features. Readers are advised to note that other implementations may exist.

According to RFC 7942 [RFC7942], "this will allow reviewers and working groups to assign due consideration to documents that have the benefit of running code, which may serve as evidence of valuable experimentation and feedback that have made the implemented

protocols more mature. It is up to the individual working groups to use this information as they see fit".

7.1. IIT-CNR/Registro.it RDAP Server

Responsible Organization: Institute of Informatics and Telematics of National Research Council (IIT-CNR)/Registro.it

Location: https://rdap.pubtest.nic.it/

Description: This implementation includes support for RDAP queries using data from the public test environment of .it ccTLD. The "redacted" array can be returned in the response to the domain lookup that is the only available to anonymous users.

Level of Maturity: This is an "alpha" test implementation.

Coverage: This implementation includes all of the features described in this specification.

Contact Information: Mario Loffredo, mario.loffredo@iit.cnr.it

8. Security Considerations

The server including a redacted signal provides an unauthorized client additional information related to the existence of data. Servers MAY exclude the redacted members for RDAP fields that are considered a privacy issue in providing a data existence signal.

9. Acknowledgements

The authors wish to thank the following persons for their feedback and suggestions: Marc Blanchet, Scott Hollenbeck, Mario Loffredo, Gustavo Lozano, and Rick Wilhelm.

10. References

10.1. Informative References

[I-D.ietf-regext-rdap-jscontact] Loffredo, M. and G. Brown, "Using JSContact in Registration Data Access Protocol (RDAP) JSON Responses", Work in Progress, Internet-Draft, draft-ietf-regext-rdap-jscontact-12, 2 May 2022, <https://datatracker.ietf.org/doc/html/draft-ietf-regext-rdap-jscontact-12>.

[RFC8605] Hollenbeck, S. and R. Carney, "vCard Format Extensions: ICANN Extensions for the Registration Data Access Protocol (RDAP)", RFC 8605, DOI 10.17487/RFC8605, May 2019, <https://www.rfc-editor.org/info/rfc8605>.

10.2. Normative References

[I-D.ietf-jsonpath-base] Gössner, S., Normington, G., and C. Bormann, "JSONPath: Query expressions for JSON", Work in Progress, Internet-Draft, draft-ietf-jsonpath-base-05, 25 April 2022, <https://datatracker.ietf.org/doc/html/draft-ietf-jsonpath-base-05>.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC6350] Perreault, S., "vCard Format Specification", RFC 6350, DOI 10.17487/RFC6350, August 2011, <https://www.rfc-editor.org/info/rfc6350>.

[RFC7095] Kewisch, P., "jCard: The JSON Format for vCard", RFC 7095, DOI 10.17487/RFC7095, January 2014, <https://www.rfc-editor.org/info/rfc7095>.

[RFC7942] Sheffer, Y. and A. Farrel, "Improving Awareness of Running Code: The Implementation Status Section", BCP 205, RFC 7942, DOI 10.17487/RFC7942, July 2016, <https://www.rfc-editor.org/info/rfc7942>.

[RFC8126] Cotton, M., Leiba, B., and T. Narten, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 8126, DOI 10.17487/RFC8126, June 2017, <https://www.rfc-editor.org/info/rfc8126>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC8259] Bray, T., Ed., "The JavaScript Object Notation (JSON) Data Interchange Format", STD 90, RFC 8259, DOI 10.17487/RFC8259, December 2017, <https://www.rfc-editor.org/info/rfc8259>.

[RFC9082] Hollenbeck, S. and A. Newton, "Registration Data Access Protocol (RDAP) Query Format", STD 95, RFC 9082, DOI 10.17487/RFC9082, June 2021, <https://www.rfc-editor.org/info/rfc9082>.

[RFC9083] Hollenbeck, S. and A. Newton, "JSON Responses for the Registration Data Access Protocol (RDAP)", STD 95, RFC 9083, DOI 10.17487/RFC9083, June 2021, <https://www.rfc-editor.org/info/rfc9083>.

Appendix A. Change History

A.1. Change from 00 to 01

1. Changed rdapConformance to use pointed "redacted_0.1" value to support structural changes of the extension up to the target of "redacted_1.0".
2. Updates based on the Gustavo Lozano feedback:
    1. Updated the language to change the special treatment of jCard to be more generic for future RDAP extensions that leverage fixed length JSON arrays.
    2. Added "RDAP extensions should define any special JSONPath considerations required to identify redacted RDAP fields if the these considerations are insufficient." to the JSONPath Considerations section to generalize it.
3. Updates based on the Marc Blanchet feedback:
    1. Added a reference to draft-ietf-regext-rdap-jscontact as an example of an RDAP extension that is suited for the Redaction by Removal Method based on the lack of dependency on positional JSON arrays.
    2. Added support for registered and unregistered (free-form) redaction reasons by changing the "reason" property to be a JSON object with the "type" and "description" properties. The "type" property includes registration in the IANA JSON Values Registry.
    3. Added a "JSON Values Registry" section in the IANA Considersations section to define the "redaction reason" JSON Values Registry Type values to support the registration of redaction reasons.
4. Updates based on the Mario Loffredo feedback:
    1. Added support for registered and unregistered (free-form) redaction names by changing the "reason" property to be a JSON object with the "type" and "description" properties. The "type" property includes registration in the IANA JSON Values Registry.
    2. Added a "JSON Values Registry" section in the IANA Considersations section to define the "redaction name" JSON Values Registry Type values to support the registration of redaction names.
    3. Added a JSONPath Considerations item associated with handling entities with multiple roles.
    4. Added language to restrict the extension to responses.

A.2. Change from 01 to 02

1. Updates to add support for RDAP search responses:

    1. Replaced "RDAP lookup response" with "RDAP response" throughout the draft to expand the scope to include search.
    2. Updated the description in the second paragraph of the Introduction to cover both a lookup response and a search response.
    3. Added an example of the use of an absoluate path for a search response to the "JSONPath Considerations" section.
    4. Added a description of the placement of the "redacted" member in a lookup response and a search response in the ""redacted" Member" section.
    5. Added an example of an unredacted search response and a redacted search response in the ""redacted" Member" section.

A.3. Change from 02 to 03

1. Fixed mismatch of the extension identifier, which was updated to "redacted_0.1" throughout the draft based on feedback from Mario Loffredo.
2. Added the JSONPath Considerations item associated with redacting fields for multiple entities with the same role based on implementation feedback from Mario Loffredo.
3. Added the Implementation Status section that includes the server implementation by Mario Loffredo.
4. Added use of numbered figures for easy reference for JSON Values Registry registrations.
5. Updated the example unredacted and redacted lookup responses to include the "objectClassName" and "handle" members.
6. Changed RFC7482 and RFC7483 references to RFC9082 and RFC9083, respectively.

A.4. Change from 03 to 04

1. Changed the extension identifier to be "redacted" instead of a versioned value, which will be leveraged for both the rdapConformance value and the JSON Values.
2. Changed the RDAP Conformance to be "redacted_level_0.2", which leveraged the extension identifier as a prefix along with "_level_" and a pointed version number. The version number will become "1.0" once the draft passes WGLC.
3. Added the Redaction by Replacement Value Method.

A.5. Change from 04 to 05

1. Update the RDAP Extensions Registry entries to include the identifier that is used for the RDAP conformance value and to include the "redacted" prefix indentifier to use for the JSON response member.
2. Changed the RDAP Conformance to be "redacted_level_0_3", which is registered in the RDAP Extensions Registry. The RDAP Conformance value will become "redacted_level_1" once the draft passes WGLC.

A.6. Change from 05 to 06

1. Fixed a couple nits.
2. Updated the Redaction by Replacement Value Method email web form examples to use the "contact-uri" jCard property of RFC 8605.

Authors' Addresses

James Gould
VeriSign, Inc.
12061 Bluemont Way
Reston, VA 20190
United States of America

Email: jgould@verisign.com
URI: http://www.verisigninc.com

David Smith
VeriSign, Inc.
12061 Bluemont Way
Reston, VA 20190
United States of America

Email: dsmith@verisign.com
URI: http://www.verisigninc.com

Jody Kolker
GoDaddy Inc.
14455 N. Hayden Rd. #219
Scottsdale, AZ 85260
United States of America

Email: jkolker@godaddy.com
URI: http://www.godaddy.com

Roger Carney
GoDaddy Inc.
14455 N. Hayden Rd. #219

Scottsdale, AZ 85260
United States of America

Email: rcarney@godaddy.com
URI: http://www.godaddy.com