Internet Engineering Task Force (IETF)
Request for Comments: 9472
Category: Standards Track
ISSN: 2070-1721

E. Lear
Cisco Systems
S. Rose
NIST

October 2023

A YANG Data Model for Reporting Software Bills of Materials (SBOMs) and Vulnerability Information

Модель данных YANG для представления спецификаций программного обеспечения (SBOM) и сведений об уязвимостях

Аннотация

Для повышения уровня кибербезопасности требуется автоматизация, позволяющая определить, какое программное обеспечение (ПО) применяется в устройстве, имеет ли это ПО уязвимости, а также получить рекомендации поставщиков (при наличии). Этот документ расширяет схему YANG пользовательского описания от производителя (Manufacturer User Description или MUD) для указания местоположения спецификаций ПО (software bills of materials или SBOM) и сведений об уязвимостях путём внесения «схемы прозрачности».

Статус документа

Документ относится к категории Internet Standards Track.

Документ является результатом работы IETF[1] и представляет согласованный взгляд сообщества IETF. Документ прошёл открытое обсуждение и был одобрен для публикации IESG[2]. Дополнительную информацию о стандартах Internet можно найти в разделе 2 в RFC 7841.

Информацию о текущем статусе документа, ошибках и способах обратной связи можно найти по ссылке https://www.rfc-editor.org/info/rfc9472.

Оглавление

1. Введение

Предпринято много усилий по улучшению видимости программ, работающих в системе, а также уязвимостей, которые могут быть в этих программах [EO2021].

Этот документ призван ответить на два вопроса для большого числа разнотипных устройств:

[1] Internet Engineering Task Force - комиссия по решению инженерных задач Internet.
[2] Internet Engineering Steering Group - комиссия по инженерным разработкам Internet.

- имеется ли в системе определённая уязвимость?
- какие устройства в конкретной среде имеют уязвимости, требующие определённых действий?

Документ не задаёт формат этих сведений и лишь указывает, как находить и извлекать такую информацию. Таким образом, модель предназначена для упрощения поиска и сама по себе не предоставляет доступа к базовым данным.

Спецификации ПО (SBOM) являются описаниями ПО, включающими сведения о версии и зависимостях, связанные с устройством. Имеются разные форматы SBOM, такие как [SPDX[1]] и CycloneDX [CycloneDX15].

Уязвимости системы можно описать аналогично, используя несколько форматов данных, таких как упомянутый CycloneDX, [CVRF[2]], [CSAF[3]]. Эти сведения обычно служат для информирования администратором об известных уязвимостях системы.

SBOM и сведения об уязвимостях могут применяться вместе с данными об уязвимостях из других источников. Средства управления сетью могут обнаружить в системе определённый набор используемых программных компонентов, выполнить поиск в национальной базе данных об уязвимостях для определения известных уязвимостей и применить полученные от изготовителя сведения об уязвимостях с помощью этого механизма для создания отчёта об уязвимостях. Этот отчёт может применяться для указания версии ПО (при её наличии), устраняющей уязвимость, а также наличия в системе уязвимого кода.

Оба класса элементов информации являются необязательными в модели, представленной в этом документе. Можно представлять SBOM, сведения об уязвимостях или то и другое сразу.

Отметим, что форматы SBOM позволяют передавать другие сведения и наиболее распространены среди них условия лицензирования. Поскольку эта спецификация нейтральна к содержимому сведений, выбор набора поддерживаемых атрибутов остаётся за разработчиками форматов, такими как Linux Foundation, OASIS, ISO.

Этот документ не задаёт способы непосредственного извлечения сведения об уязвимостях из конечных точек. Это обусловлено тем, что сведения об уязвимостях могут меняться при обновлении программ. Однако некоторые форматы SBOM могут включать также информацию об уязвимостях.

SBOM и сведения об уязвимостях анонсируются и извлекаются с помощью дополнения YANG для модели MUD [RFC8520]. Отметим, что схема создаёт группировку, которую можно применять и независимо от MUD. Кроме того, не требуется наличия других функций MUD, таких как контроль доступа.

Описанные в этом документе механизмы предназначены для решения двух вариантов применения.

- Система управления сетевого уровня, извлекающая информацию из устройств IoT[4] в рамках жизненного цикла. Такие устройства могут (но не обязаны) иметь интерфейсы запросов.
- Система управления прикладного уровня, извлекающая сведения об уязвимостях или SBOM для оценки состояния сервера приложений. Такие серверы сами могут быть контейнерами или гипервизорами. Обнаружение топологии серверов выходит за рамки этого документа.

Для реализации этих вариантов применений объекты могут быть найдены одним из трёх методов:

1. на самих устройствах;
2. на web-сайте (например, по URI);
3. через контакт с поставщиком по отдельному каналу (out-of-band).

При использовании первого метода устройства будут иметь интерфейсы, разрешающие извлечение данных напрямую. Примерами таких интерфейсов могут быть конечные точки HTTP [RFC9110] или протокола приложений с ограничениями (Constrained Application Protocol или CoAP) [RFC7252] для извлечения данных, а также приватные интерфейсы.

При использовании второго метода, когда у устройства нет подходящего интерфейса для извлечения данных напрямую, но он доступен непосредственно от изготовителя, URI для этих сведений обнаруживается через такие интерфейсы, как MUD по протоколу DHCP или механизмы начальной загрузки и передачи прав владения.

При использовании третьего метода поставщик может сделать SBOM или сведения об уязвимостях доступными при определённых обстоятельствах и ему может потребоваться индивидуальная оценка запросов. Результатом такой оценки может быть SBOM, сведения об уязвимости, URL с ограничениями или отсутствие доступа.

Для обеспечения возможности обнаружения на прикладном уровне этот документ определяет общеизвестный (well-known) идентификатор URI [RFC8615]. Средства управления или организации (оркестровки) могут обращаться к общеизвестному URI для извлечения данных SBOM. Могут потребоваться дополнительные запросы в зависимости от структуры и содержимого первого отклика.

1.1. Уровни требований

Ключевые слова **необходимо** (MUST), **недопустимо** (MUST NOT), **требуется** (REQUIRED), **нужно** (SHALL), **не следует** (SHALL NOT), **следует** (SHOULD), **не нужно** (SHOULD NOT), **рекомендуется** (RECOMMENDED), **не рекомендуется** (NOT RECOMMENDED), **возможно** (MAY), **необязательно** (OPTIONAL) в данном документе интерпретируются в соответствии с BCP 14 [RFC2119] [RFC8174] тогда и только тогда, когда они выделены шрифтом, как показано здесь.

[1]Software Package Data Exchange - обмен данными о программных пакетах
[2]Common Vulnerability Reporting Framework - базовая модель отчётов об уязвимостях.
[3]Common Security Advisory Format - базовый формат сведений о безопасности.
[4]Internet of Things - Internet вещей.

1.2. Как извлекается информация

В разделе 4 описана модель данных для расширения формата файлов MUD для передачи SBOM и сведений об уязвимостях. В параграфе 1.5 [RFC8520] описаны механизмы, с помощью которых устройства могут выдавать URL для указания такого файла. Кроме того, устройства могут предоставлять URL в документации или с помощью QR-кода на коробке (корпусе). В разделе 2 описан общеизвестный идентификатор URL, по которому можно получить SBOM от локального устройства.

Отметим, что сведения об уязвимостях и SBOM могут изменяться с разной скоростью. Узел cache-validity для MUD предоставляет изготовителям возможность контроля частоты проверки этих изменений через узел cache-validity.

1.3. Форматы

Имеется много способов представления SBOM и сведений об уязвимостях. При получении этих данных от устройства или удалённого web-сервера инструментам нужно просматривать заголовок Content-Type для точного определения формата передачи. Поскольку возможности устройств IoT обычно ограничены, применять конкретный заголовок Accept: в HTTP или Accept Option в CoAP **не рекомендуется**. Вместо этого инструментам backend рекомендуется поддерживать все известные форматы, а данные SBOM, переданные с неизвестным типом носителя **следует** отбрасывать без уведомления отправителя.

Если в одном файле предполагается поддерживать несколько SBOM, тип носителя должен соответствующим образом отражать это. Например, можно использовать application/{someformat}+json-seq. Подходящая регистрация в таких случаях отдаётся на усмотрение тех, кто поддерживает форматы.

Некоторые форматы могут поддерживать сведения об уязвимостях и инвентаризации ПО. При доступности обоих типов сведений по одному URL это **должны** указывать как sbom-url, так и элементы списка vuln-url. Системы управления сетью **должны** учитывать доступность SBOM и сведений об уязвимостях через один ресурс и не извлекать его дважды.

2. Общеизвестная конечная точка

Определяется общеизвестная конечная точка /.well-known/sbom для извлечения SBOM. Как отмечено выше, точный формат отклика определяется представленным Content-Type.

3. Расширение mud-transparency

Здесь дано формальное определение расширения mud-transparency, включающего 2 части. Первой частью является имя расширения transparency, включаемое в массив extensions файла MUD. Это расширение схемы предназначено для использования везде, где это уместно (не только в MUD). Второй частью является контейнер mud, дополненный списком источников SBOM.

```
module: ietf-mud-transparency

  augment /mud:mud:
    +--rw transparency
       +--rw (sbom-retrieval-method)?
       |  +--:(cloud)
       |  |  +--rw sboms* [version-info]
       |  |     +--rw version-info    string
       |  |     +--rw sbom-url?       inet:uri
       |  +--:(local-well-known)
       |  |  +--rw sbom-local-well-known?   identityref
       |  +--:(sbom-contact-info)
       |     +--rw sbom-contact-uri?        inet:uri
       +--rw sbom-archive-list?             inet:uri
       +--rw (vuln-retrieval-method)?
          +--:(cloud)
          |  +--rw vuln-url*                inet:uri
          +--:(vuln-contact-info)
             +--rw vuln-contact-uri?        inet:uri
```

Описание деревьев YANG содержится в [RFC8340].

4. Дополнение mud-sbom для модели данных YANG MUD

Этот модуль YANG ссылается на [RFC6991], [RFC7231], [RFC7252], [RFC8520], [RFC9110].

```
<CODE BEGINS> file "ietf-mud-transparency@2023-10-10.yang"
module ietf-mud-transparency {
  yang-version 1.1;
  namespace "urn:ietf:params:xml:ns:yang:ietf-mud-transparency";
  prefix mudtx;

  import ietf-inet-types {
    prefix inet;
    reference
      "RFC 6991: Common YANG Data Types";
  }
  import ietf-mud {
    prefix mud;
    reference
      "RFC 8520: Manufacturer Usage Description Specification";
  }

  organization
    "IETF OPSAWG (Ops Area) Working Group";
```

```
    contact
      "WG Web: <https://datatracker.ietf.org/wg/opsawg/>
       WG List: <opsawg@ietf.org>

       Editor: Eliot Lear <lear@cisco.com>
       Editor: Scott Rose <scott.rose@nist.gov>";
    description
      "Этот модуль YANG дополняет модель ietf-mud для предоставления
       SBOM и сведений об уязвимостях.

       Ключевые слова ДОЛЖНО, НЕДОПУСТИМО, ТРЕБУЕТСЯ, НУЖНО, НЕ НУЖНО,
       СЛЕДУЕТ, НЕ СЛЕДУЕТ, РЕКОМЕНДУЕТСЯ, НЕ РЕКОМЕНДУЕТСЯ, МОЖНО,
       НЕОБЯЗАТЕЛЬНО в этом документе трактуются в соответствии с
       BCP 14 (RFC 2119) (RFC 8174) тогда и только тогда, когда они
       указаны заглавными буквами, как показано здесь.
       Авторские права (Copyright (c) 2023) принадлежат IETF Trust
       и лицам, указанным в качестве авторов кода. Все права защищены.

       Распространение и использование в исходной или двоичной форме с
       изменениями или без таковых разрешено в соответствии с лицензией
       Simplified BSD, изложенной в разделе 4  IETF Trust's Legal
       Provisions применительно к документам IETF
       (http://trustee.ietf.org/license-info).

       Эта версия данного модуля YANG является частью RFC 9472
       (https://www.rfc-editor.org/info/rfc9472), где
       правовые вопросы рассмотрены более полно.";

    revision 2023-10-10 {
      description
        "Исходный вариант предложенного стандарта.";
      reference
        "RFC 9472: A YANG Data Model for Reporting Software Bills
         of Materials (SBOMs) and Vulnerability Information";
    }

    identity local-type {
      description
        "Базовый идентификатор для локальных общеизвестных вариантов.";
    }

    identity http {
      base mudtx:local-type;
      description
        "Используется http (RFC 7231) (без защиты) для извлечения SBOM.
         Этот метод НЕ РЕКОМЕНДУЕТСЯ, но может оказаться неизбежным для
         некоторых вариантов внедрения, где нет TLS.";
        reference
          "RFC 7231: Hypertext Transfer Protocol (HTTP/1.1):
           Semantics and Content";
    }

    identity https {
      base mudtx:local-type;
      description
        "Применяется https (с защитой) для извлечения SBOM (RFC 9110)";
        reference
          "RFC 9110: HTTP Semantics";
    }

    identity coap {
      base mudtx:local-type;
      description
        "Для извлечения SBOM применяется COAP (RFC 7252, без защиты).
         Этот метод НЕ РЕКОМЕНДУЕТСЯ, но может оказаться неизбежным для
         некоторых вариантов внедрения, где нет TLS.";
        reference
          "RFC 7252: The Constrained Application Protocol (CoAP)";
    }

    identity coaps {
      base mudtx:local-type;
      description
        "Для извлечения SBOM применяется COAPS (RFC 7252. с защитой).";
    }

    grouping transparency-extension {
      description
        "Эта группировка обеспечивает средства описания размещения SBOM
         и описаний уязвимостей.";
      container transparency {
        description
          "Методы получения SBOM и данных об уязвимостях.";
        choice sbom-retrieval-method {
          description
            "Как найти данные SBOM.";
```

```
          case cloud {
            list sboms {
              key "version-info";
              description
                "Список SBOM, связанных с разными аппаратными или
                 программными версиями.";
              leaf version-info {
                type string;
                description
                  "Версия, к которой относится SBOM.";
              }
              leaf sbom-url {
                type inet:uri {
                  pattern '((coaps?)|(https?)):.*';
                }
                description
                  "Статически размещённый идентификатор URL.";
              }
            }
          }
          case local-well-known {
            leaf sbom-local-well-known {
              type identityref {
                base mudtx:local-type;
              }
              description
                "Коммуникационный протокол для выбора.";
            }
          }
          case sbom-contact-info {
            leaf sbom-contact-uri {
              type inet:uri {
                pattern '((mailto)|(https?)|(tel)):.*';
              }
              description
                "Это ДОЛЖНА быть схема uri tel, http, https или mailto,
                 которую клиенты могут применять для получения SBOM.";
            }
          }
        }
        leaf sbom-archive-list {
          type inet:uri;
          description
            "Этот идентификатор URI возвращает JSON-список URL с SBOM,
             которые ранее опубликованы для этого устройства. Даты
             публикации содержатся в SBOM.";
        }
        choice vuln-retrieval-method {
          description
            "Как найти сведения об уязвимостях.";
          case cloud {
            leaf-list vuln-url {
              type inet:uri;
              description
                "Список статически размещённых URL, указывающих
                 сведения об уязвимостях.";
            }
          }
          case vuln-contact-info {
            leaf vuln-contact-uri {
              type inet:uri {
                pattern '((mailto)|(https?)|(tel)):.*';
              }
              description
                "Это ДОЛЖНА быть схема uri tel, http, https или mailto,
                 которую клиенты могут применять для получения сведений
                 об уязвимостях.";
            }
          }
        }
      }
    }

    augment "/mud:mud" {
      description
        "Добавление расширения.";
      uses transparency-extension;
    }
  }
  <CODE ENDS>
```

5. Примеры

В приведённых примерах файлов MUD, использующих облачный сервис, modelX представляет местоположение SBOM в форме URL. Отметим, что списки управления доступом (Access Control List или ACL) в файле MUD **не** требуются, хотя для устройств на основе IP их применение является хорошей идеей.

5.1. Без ACL

Приведённый ниже файл MUD показывает, как получить SBOM и сведения об уязвимостях без ACL.

```
{
 "ietf-mud:mud": {
   "mud-version": 1,
   "extensions": [
     "transparency"
   ],
   "mudtx:transparency": {
     sboms: [ {
     "version-info": "1.2",
     "sbom-url": "https://iot.example.com/info/modelX/sbom.json"
     } ],
     "vuln-url" : [
       "https://iotd.example.com/info/modelX/csaf.json"
     ]
   },
   "mud-url": "https://iot.example.com/modelX.json",
   "mud-signature": "https://iot.example.com/modelX.p7s",
   "last-update": "2022-01-05T13:29:12+00:00",
   "cache-validity": 48,
   "is-supported": true,
   "systeminfo": "retrieving vuln and SBOM info via a cloud service",
   "mfg-name": "Example, Inc.",
   "documentation": "https://iot.example.com/doc/modelX",
   "model-name": "modelX"
 }
}
```

Следующий пример демонстрирует извлечение данных SBOM из облака.

```
{
 "ietf-mud:mud": {
   "mud-version": 1,
   "extensions": [
     "transparency"
   ],
   "mudtx:transparency": {
     sboms: [ {
     "version-info": "1.2",
     "sbom-url": "https://iot.example.com/info/modelX/sbom.json"
     } ],
   },
   "mud-url": "https://iot.example.com/modelX.json",
   "mud-signature": "https://iot.example.com/modelX.p7s",
   "last-update": "2022-01-05T13:29:12+00:00",
   "cache-validity": 48,
   "is-supported": true,
   "systeminfo": "retrieving vuln and SBOM info via a cloud service",
   "mfg-name": "Example, Inc.",
   "documentation": "https://iot.example.com/doc/modelX",
   "model-name": "modelX"
 }
}
```

5.2. SBOM на устройстве

В приведённом ниже примере SBOM размещается на устройстве, а сведений об уязвимостях не предоставдяется.

```
{
  "ietf-mud:mud": {
    "mud-version": 1,
    "extensions": [
      "transparency"
    ],
    "mudtx:transparency": {
      "sbom-local-well-known": "https"
    },
    "mud-url": "https://iot.example.com/modelX.json",
    "mud-signature": "https://iot.example.com/modelX.p7s",
    "last-update": "2022-01-05T13:29:47+00:00",
    "cache-validity": 48,
    "is-supported": true,
    "systeminfo": "retrieving SBOM info from a local source",
    "mfg-name": "Example, Inc.",
    "documentation": "https://iot.example.com/doc/modelX",
    "model-name": "modelX"
  }
}
```

В следующем примере SBOM извлекается из устройства, а сведения об уязвимостях - из облака. Вероятно, это распространённый случай, так как производители могут получать данные об уязвимостях чаще, чем обновляется ПО.

```
{
 "ietf-mud:mud": {
   "mud-version": 1,
   "extensions": [
     "transparency"
```

```
    ],
    "mudtx:transparency": {
      "sbom-local-well-known": "https",
      "vuln-url" : [
        "https://iotd.example.com/info/modelX/csaf.json"
      ]
    },
    "mud-url": "https://iot-device.example.com/modelX.json",
    "mud-signature": "https://iot-device.example.com/modelX.p7s",
    "last-update": "2022-01-05T13:25:14+00:00",
    "cache-validity": 48,
    "is-supported": true,
    "systeminfo": "mixed example: SBOM on device, vuln info in cloud",
    "mfg-name": "Example, Inc.",
    "documentation": "https://iot-device.example.com/doc/modelX",
    "model-name": "modelX"
  }
}
```

5.3. Нужны дополнительные контакты

В приведённом ниже примере менеджер сети должен предпринять дополнительные действия для извлечения SBOM. Доступны также сведения об уязвимостях.

```
{
"ietf-mud:mud": {
"mud-version": 1,
"extensions": [
  "transparency"
],
"mudtx:transparency": {
  "contact-info": "https://iot-device.example.com/contact-info.html",
    "vuln-url" : [
      "https://iotd.example.com/info/modelX/csaf.json"
    ]
},
"mud-url": "https://iot-device.example.com/modelX.json",
"mud-signature": "https://iot-device.example.com/modelX.p7s",
"last-update": "2021-07-09T06:16:42+00:00",
"cache-validity": 48,
"is-supported": true,
"systeminfo": "retrieving vuln and SBOM info via a cloud service",
"mfg-name": "Example, Inc.",
"documentation": "https://iot-device.example.com/doc/modelX",
"model-name": "modelX"
}
}
```

5.4. С ACL

Ниже представлен пример, где устройство предоставляет SBOM и сведения об уязвимостях, а также имеются списки управления доступом.

```
{
 "ietf-mud:mud": {
   "mud-version": 1,
   "extensions": [
     "transparency"
   ],
   "mudtx:transparency": {
     "sbom-local-well-known": "https",
     "vuln-url" : [
       "https://iotd.example.com/info/modelX/csaf.json"
     ]
   },
   "mud-url": "https://iot.example.com/modelX.json",
   "mud-signature": "https://iot.example.com/modelX.p7s",
   "last-update": "2022-01-05T13:30:31+00:00",
   "cache-validity": 48,
   "is-supported": true,
   "systeminfo": "retrieving vuln and SBOM info via a cloud service",
   "mfg-name": "Example, Inc.",
   "documentation": "https://iot.example.com/doc/modelX",
   "model-name": "modelX",
   "from-device-policy": {
     "access-lists": {
       "access-list": [
         {
           "name": "mud-65443-v4fr"
         }
       ]
     }
   },
   "to-device-policy": {
     "access-lists": {
       "access-list": [
         {
```

```
            "name": "mud-65443-v4to"
          }
        ]
      }
    }
  },
  "ietf-access-control-list:acls": {
    "acl": [
      {
        "name": "mud-65443-v4to",
        "type": "ipv4-acl-type",
        "aces": {
          "ace": [
            {
              "name": "cl0-todev",
              "matches": {
                "ipv4": {
                  "ietf-acldns:src-dnsname": "iotserver.example.com"
                }
              },
              "actions": {
                "forwarding": "accept"
              }
            }
          ]
        }
      },
      {
        "name": "mud-65443-v4fr",
        "type": "ipv4-acl-type",
        "aces": {
          "ace": [
            {
              "name": "cl0-frdev",
              "matches": {
                "ipv4": {
                  "ietf-acldns:dst-dnsname": "iotserver.example.com"
                }
              },
              "actions": {
                "forwarding": "accept"
              }
            }
          ]
        }
      }
    ]
  }
}
```

Далее система управления может попытаться извлечь SBOM, определить применяемый формат по заголовку Content-Type в отклике на запрос GET, независимо повторить процесс для сведений об уязвимостях и применить ACL, если это уместно.

6. Вопросы безопасности

В этом документе описана схема для обнаружения местоположения сведений о «прозрачности» ПО и не задаётся модуль доступа к таким данным. В частности, заданный в документе модуль YANG не обязательно использовать для доступа по обычным протоколам управления сетью, таким как NETCONF [RFC6241] и RESTCONF [RFC8040], поэтому обычные вопросы безопасности для такого применения здесь не рассматриваются.

Ниже описаны меры защиты, относящиеся как к обнаружению, так и к SBOM и сведениям об уязвимостях.

Модель предусматривает средства извлечения информации как с шифрованием, так и без такового. Это вопрос прагматизма. Нешифрованное взаимодействие позволяет манипулировать извлекаемыми данными, поэтому разработчикам **рекомендуется** предоставлять средства настройки конечных точек для применения TLS или DTLS.

Модуль ietf-mud-transparency не оказывает оперативного влияния на сам элемент и применяется для обнаружения данных о состоянии, которые могут быть доступны в элементе или вне его. В той мере, в какой сам модуль становится доступным для записи, это указывает лишь на изменение способа извлечения элементов, доступных только для чтения. Например, нет средств выгрузки SBOM в элемент. Ниже обсуждаются дополнительные риски, применимые ко всем узлам внутри контейнера transparency.

Если злоумышленник изменяет элементы, он может направить автоматизацию на извлечение набора URL, отличающегося от предусмотренного разработчиком. Это вызывает два конкретных варианта рисков:

- извлекаемая информация становится ложной;
- URL указывают на вредоносных код (malware).

Для устранения этих рисков и иных фальсификаций URL следует:

- проверять все URL, связанные с облаком, на предмет доверия к ним;
- предоставить администратору возможность изменять последующую обработку, когда полномочия заведомо не заслуживают доверия.

В SBOM содержится перечень ПО. Сведения о конкретных программах, загруженных в систему, могут помочь атакующему найти подходящий эксплойт для известной уязвимости или создать новый для этой системы. Однако при доступности ПО злоумышленнику он может самостоятельно создать очень близкий перечень программ. Если такие сведения размещаются на самом устройстве, конечной точке **не следует** предоставлять по умолчанию неограниченный доступ к well-known URL.

Другие серверы, предоставляющие сведения, **могут** ограничивать доступ к SBOM, применяя подобающую проверку полномочий в HTTP. Одним из способов является выпуск сертификата для клиента после прохождения им регистрации. Другим вариантом является комбинированное применение OAuth. В частности, если система пытается извлечь SBOM через HTTP или CoAP и полномочия клиента не подтверждены, сервер **должен** выдавать подходящую ошибку с инструкциями по регистрации конкретного клиента.

Другой риск связан с несоответствием данных SBOM фактическому перечню ПО на устройстве (контейнере). Например, изготовитель может обновить SBOM на своём сервере, ещё не обновив отдельные устройства. Это может приводить к некорректному применению правил на устройстве. Однозначное сопоставления версий программ на устройстве с SBOM позволяет минимизировать этот риск.

Для дополнительного смягчения атак на устройства изготовителям **следует** рекомендовать средства контроля доступа через сеть.

Сведения об уязвимостях обычно делаются доступными в базах данных, таких как NIST National Vulnerability Database [NISTNVD]. Возможно предоставление производителями таких сведений некоторым заказчикам заранее. Этот вопрос здесь не обсуждается, однако в таких случаях для этих сведения **следует** применять подходящие средства контроля доступа и проверки полномочий.

7. Взаимодействие с IANA

7.1. Расширение MUD

Агентство IANA добавило transparency в реестр MUD Extensions [RFC8520].

```
Value:  transparency
Reference:  RFC 9472
```

7.2. Регистрация YANG

Агентство IANA зарегистрировало указанный ниже модуль YANG в реестре YANG Module Names [RFC6020].

```
Name:  ietf-mud-transparency
Namespace:  urn:ietf:params:xml:ns:yang:ietf-mud-transparency
Maintained by IANA:  N
Prefix:  mudtx
Reference:  RFC 9472
```

Приведённый ниже дескриптор URI зарегистрирован в реестре IETF XML Registry [RFC3688].

```
URI:  urn:ietf:params:xml:ns:yang:ietf-mud-transparency
Registrant Contact:  IESG
XML:  None.  Namespace URIs do not represent an XML specification.
```

7.3. Общеизвестный префикс

Агентство IANA добавило указанный ниже суффикс URI в реестр Well-Known URIs в соответствии с [RFC8615].

```
URI Suffix:  sbom
Change Controller:  IETF
Reference:  RFC 9472
Status:  permanent
Related Information:  See ISO/IEC 5962:2021 and SPDX.org
```

8. Литература

8.1. Нормативные документы

[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>.

[RFC3688] Mealling, M., "The IETF XML Registry", BCP 81, RFC 3688, DOI 10.17487/RFC3688, January 2004, <https://www.rfc-editor.org/info/rfc3688>.

[RFC6020] Bjorklund, M., Ed., "YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)", RFC 6020, DOI 10.17487/RFC6020, October 2010, <https://www.rfc-editor.org/info/rfc6020>.

[RFC6241] Enns, R., Ed., Bjorklund, M., Ed., Schoenwaelder, J., Ed., and A. Bierman, Ed., "Network Configuration Protocol (NETCONF)", RFC 6241, DOI 10.17487/RFC6241, June 2011, <https://www.rfc-editor.org/info/rfc6241>.

[RFC6991] Schoenwaelder, J., Ed., "Common YANG Data Types", RFC 6991, DOI 10.17487/RFC6991, July 2013, <https://www.rfc-editor.org/info/rfc6991>.

[RFC7231] Fielding, R., Ed. and J. Reschke, Ed., "Hypertext Transfer Protocol (HTTP/1.1): Semantics and Content", RFC 7231, DOI 10.17487/RFC7231, June 2014, <https://www.rfc-editor.org/info/rfc7231>.

[RFC7252] Shelby, Z., Hartke, K., and C. Bormann, "The Constrained Application Protocol (CoAP)", RFC 7252, DOI 10.17487/RFC7252, June 2014, <https://www.rfc-editor.org/info/rfc7252>.

[RFC8040] Bierman, A., Bjorklund, M., and K. Watsen, "RESTCONF Protocol", RFC 8040, DOI 10.17487/RFC8040, January 2017, <https://www.rfc-editor.org/info/rfc8040>.

[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>.

[RFC8520] Lear, E., Droms, R., and D. Romascanu, "Manufacturer Usage Description Specification", RFC 8520, DOI 10.17487/RFC8520, March 2019, <https://www.rfc-editor.org/info/rfc8520>.

[RFC8615] Nottingham, M., "Well-Known Uniform Resource Identifiers (URIs)", RFC 8615, DOI 10.17487/RFC8615, May 2019, <https://www.rfc-editor.org/info/rfc8615>.

[RFC9110] Fielding, R., Ed., Nottingham, M., Ed., and J. Reschke, Ed., "HTTP Semantics", STD 97, RFC 9110, DOI 10.17487/RFC9110, June 2022, <https://www.rfc-editor.org/info/rfc9110>.

8.2. Дополнительная литература

[CSAF] Rock, L., Ed., Hagen, S., Ed., and T. Schmidt, Ed., "Common Security Advisory Framework Version 2.0", OASIS Standard, November 2022, <https://docs.oasis-open.org/csaf/csaf/v2.0/csaf-v2.0.html>.

[CVRF] Hagen, S., Ed., "CSAF Common Vulnerability Reporting Framework (CVRF) Version 1.2", Committee Specification 01, September 2017, <https://docs.oasis-open.org/csaf/csaf-cvrf/v1.2/csaf-cvrf-v1.2.pdf>.

[CycloneDX15] CycloneDX, "CycloneDX v1.5 JSON Reference", Version 1.5.0, <https://cyclonedx.org/docs/1.5/json>.

[EO2021] Biden, J., "Executive Order on Improving the Nation's Cybersecurity", EO 14028, May 2021.

[NISTNVD] NIST, "National Vulnerability Database", <https://nvd.nist.gov>.

[RFC8340] Bjorklund, M. and L. Berger, Ed., "YANG Tree Diagrams", BCP 215, RFC 8340, DOI 10.17487/RFC8340, March 2018, <https://www.rfc-editor.org/info/rfc8340>.

[SPDX] The Linux Foundation, "The Software Package Data Exchange (SPDX) Specification", Version 2.3, 2022, <https://spdx.github.io/spdx-spec/v2.3/>.

Благодарности

Спасибо Russ Housley, Dick Brooks, Tom Petch, Nicolas Comstedt за их обзорные комментарии.

Адреса авторов

Eliot Lear
Cisco Systems
Richtistrasse 7
CH-8304 Wallisellen
Switzerland
Phone: +41 44 878 9200
Email: lear@cisco.com

Scott Rose
NIST
100 Bureau Dr.
Gaithersburg, MD 20899
United States of America
Phone: +1 301-975-8439
Email: scott.rose@nist.gov

Перевод на русский язык

Николай Малых
nmalykh@protokols.ru