

Translation of Structure of Management Information Version 2 (SMIv2) MIB Modules to YANG Modules

Преобразование модулей SMIv2 в модули YANG

Тезисы

Язык моделирования данных YANG служит для создания моделей данных конфигурации и состояния, которыми управляют на основе протокола NETCONF¹, вызовов удаленных процедур NETCONF и уведомления NETCONF. Структура информации управления SMIv2² определяет фундаментальные типы данных, модель объекта и правила создания и пересмотра модулей MIB для использования с протоколом SNMP³. Этот документ определяет трансляцию модулей SMIv2 MIB в модули YANG, обеспечивающие (только) чтение (config false) объектов данных, определенных в модулях SMIv2 MIB, по протоколу NETCONF.

Статус документа

Этот документ относится к категории Internet Standards Track.

Документ является результатом работы IETF⁴ и представляет согласованное мнение сообщества IETF. Документ был вынесен на открытое обсуждение и одобрен для публикации IESG⁵. Дополнительную информацию о стандартах Internet можно найти в разделе 2 документа RFC 5741.

Информация о статусе этого документа, обнаруженных ошибках и способах обратной связи доступна по ссылке <http://www.rfc-editor.org/info/rfc6643>.

Авторские права

Авторские права (с) 2012 принадлежат IETF Trust и лицам, указанным в качестве авторов документа. Все права защищены.

Этот документ является субъектом прав и ограничений, перечисленных в BCP 78 и IETF Trust Legal Provisions и относящихся к документам IETF (<http://trustee.ietf.org/license-info>), на момент публикации данного документа. Прочтите упомянутые документы внимательно, поскольку в них описаны права и ограничения, относящиеся к данному документу. Фрагменты программного кода, включенные в этот документ, распространяются в соответствии с упрощенной лицензией BSD, как указано в параграфе 4.e документа IETF Trust Legal Provisions, без каких-либо гарантий (как указано в Simplified BSD License).

Документ может содержать материалы из IETF Document или IETF Contribution, опубликованных или публично доступных до 10 ноября 2008 года. Лица, контролирующие авторские права на некоторые из таких документов, могли не предоставить IETF Trust права разрешать внесение изменений в такие документы за рамками процессов IETF Standards. Без получения соответствующего разрешения от лиц, контролирующих авторские права этот документ не может быть изменен вне рамок процесса IETF Standards, не могут также создаваться производные документы за рамками процесса IETF Standards за исключением форматирования документа для публикации как RFC или перевода с английского языка на другие языки.

Оглавление

1. Введение.....	2
2. Отображение общеизвестных типов.....	2
3. Трансляция модулей SMIv2 и операторов SMIv2 IMPORT.....	2
3.1. Пример - импорт IF-MIB.....	3
4. Трансляция макроса MODULE-IDENTITY.....	3
4.1. Правила трансляции MODULE-IDENTITY.....	3
4.2. Пример - MODULE-IDENTITY из IF-MIB.....	3
5. Трансляция макроса TEXTUAL-CONVENTION.....	4
5.1. Правила трансляции TEXTUAL-CONVENTION.....	4
5.2. Пример - OwnerString и InterfaceIndex из IF-MIB.....	4
5.3. Пример - IfDirection из DIFFSERV-MIB.....	5
6. Трансляция назначений OBJECT IDENTIFIER.....	5
7. Трансляция макроса OBJECT-TYPE.....	5
7.1. Правила трансляции скаляров и колонок.....	5
7.2. Пример - ifNumber и ifIndex из IF-MIB.....	6
7.3. Правила трансляции нерасширяемых концептуальных таблиц.....	6
7.4. Пример - ifTable из IF-MIB.....	7
7.5. Пример - ifRcvAddressTable из IF-MIB.....	7
7.6. Пример - aiHostTable из RMON2-MIB.....	8

¹Network Configuration Protocol - протокол настройки сети.

²Structure of Management Information.

³Simple Network Management Protocol - простой протокол сетевого управления.

⁴Internet Engineering Task Force.

⁵Internet Engineering Steering Group.

7.7. Правила трансляции расширяемых концептуальных таблиц.....	9
7.8. Пример - ifXTable из IF-MIB.....	9
8. Трансляция макроса OBJECT-IDENTITY.....	10
8.1. Правила трансляции OBJECT-IDENTITY.....	10
8.2. Пример - diffServTbParamSimpleTokenBucket из DIFFSERV-MIB.....	10
9. Трансляция макроса NOTIFICATION-TYPE.....	10
9.1. Правила трансляции NOTIFICATION-TYPE.....	10
9.2. Пример - linkDown NOTIFICATION-TYPE из IF-MIB.....	11
10. Определение расширения языка YANG.....	11
11. Реализация конфигурационных узлов данных.....	13
11.1. Пример - addressMapControlTable из RMON2-MIB.....	14
12. Взаимодействие с IANA.....	15
13. Вопросы безопасности.....	15
14. Благодарности.....	15
15. Литература.....	15
15.1. Нормативные документы.....	15
15.2. Дополнительная литература.....	15
Приложение А. Отображение общеизвестных типов (нормативное).....	16
Приложение В. Генерация префиксов (информативное).....	17

1. Введение

Этот документ описывает трансляцию модулей SMIPv2 [RFC2578], [RFC2579], [RFC2580] MIB в модули YANG [RFC6020], позволяющие (только) считать (config false, как указано в параграфе 7.19.1 RFC 6020) объекты SMIPv2, определенные с модулях SMIPv2 MIB, с помощью протокола NETCONF [RFC6241]. Обсуждение причин, по которым доступные для записи (read-write и read-create) объекты SMIPv2 транслируются в доступные лишь для чтения (config false) объекты YANG, приведено в разделе 11.

Модули YANG, созданные из модулей SMIPv2, не следует изменять. Все, что требуется сделать - изменить исходные модули SMIPv2 (с подобающим обновлением операторов SMIPv2 LAST-UPDATED и REVISION), а затем запустить описанную в этом документе трансляцию. Отметим, что это не влияет на использование дополнений отклонений YANG - модули YANG, полученные из SMIPv2, можно дополнять как любые другие модули YANG, а отклонения могут использоваться для документирования отклонения реализации от полученного при трансляции модуля YANG.

Модули SMIPv1 можно преобразовать в YANG, следуя правилам [RFC3584] для преобразования SMIPv1 в SMIPv2, а затем правилам этого документа для преобразования полученного модуля SMIPv2 в YANG.

Отображение SMIPv2 в YANG иллюстрируется примерами, показывающими трансляцию фрагментов модулей SMIPv2 IF-MIB [RFC2863], DIFFSERV-MIB [RFC3289] и RMON2-MIB [RFC4502].

Ключевые слова **необходимо** (MUST), **недопустимо** (MUST NOT), **требуется** (REQUIRED), **нужно** (SHALL), **не следует** (SHALL NOT), **следует** (SHOULD), **не нужно** (SHOULD NOT), **рекомендуется** (RECOMMENDED), **возможно** (MAY), **необязательно** (OPTIONAL) в данном документе интерпретируются в соответствии с BCP 14 [RFC2119].

2. Отображение общеизвестных типов

Базовые типы SMIPv2 и некоторые общеизвестные текстовые соглашения отображены на типы YANG в соответствии с приложением А. Отображение типа OCTET STRING зависит от контекста. Если строка октетов имеет связанное сообщение DISPLAY-HINT, ей будет соответствовать строка YANG (string). Реализации **должны** форматировать значения OCTET STRING в соответствии с DISPLAY-HINT, как описано в RFC 2579. Если OCTET STRING не имеет связанного DISPLAY-HINT, используется двоичный (binary) тип. Отображение типа INTEGER зависит от его применения в качестве перечисляемого или 32-битового интегрального типа. Реализациям следует поддерживать опции для обработки ситуаций с добавлением DISPLAY-HINT при пересмотре модуля и должна сохраняться совместимость с прежними версиями (т. е. игнорирование DISPLAY-HINT).

Показанные в приложении А отображения могут потребовать импорта модулей YANG ietf-yang-types, ietf-inet-types или ietf-yang-smiv2, поскольку отображения некоторых типов и текстовых соглашений SMIPv2 в типы YANG определены в модулях ietf-yang-types и ietf-inet-types из [RFC6021], а также в модуле ietf-yang-smiv2, определенном в данном документе. Реализации **должны** добавлять импорт всех модулей, требуемых для отображения типов.

3. Трансляция модулей SMIPv2 и операторов SMIPv2 IMPORT

Модули SMIPv2 отображаются в соответствующие модули YANG. Имена создаваемых модулей YANG **должны** совпадать с именами модулей SMIPv2.

Пространство имен YANG **должно** создаваться на основе зарегистрированного IANA префикса urn:ietf:params:xml:ns:yang:smiv2: (раздел 12), за которым следует имя модуля SMIPv2. Поскольку имена модулей SMIPv2 можно считать уникальными (см. раздел 3 в [RFC2578]), полученное пространство имен YANG будет уникальным.

Префикс YANG **может** выводиться из имени модуля SMIPv2 с использованием алгоритма генерации префиксов, описанного в приложении В. Префикс YANG предполагается коротким и должен быть уникальным в рамках набора префиксов, используемых модулем YANG. Такие префиксы генерирует алгоритм, описанный в приложении В.

Операторы SMIPv2 IMPORT преобразуются в операторы YANG import. Одно из основных различий между импортом SMIPv2 и YANG заключается в том, что SMIPv2 IMPORT импортируют конкретные символы из модуля SMIPv2, а операторы YANG import импортируют все символы указанного модуля YANG.

Для создания корректных и полных операторов импорта YANG **должны** выполняться приведенные ниже правила.

- Обработка каждого элемента в каждом SMIPv2 IMPORT в соответствии с приведенным ниже порядком.
 1. Если оператор import для этого модуля SMIPv2 уже был сгенерирован, элемент игнорируется.

2. Если именем модуля SMlv2 является SNMPv2-SMI или SNMPv2-CONF, элемент игнорируется. Отметим, что эти два модуля можно игнорировать полностью, поскольку все определения из них преобразуются правилами трансляции.
 3. Если элемент является текстовым соглашением, соответствующим одному из текстовых соглашений в колонке SMlv2 types приложения A (например, MacAddress, PhysAddress или TimeStamp), элемент игнорируется.
 4. Если элемент используется в SYNTAX для OBJECT-TYPE, где MAX-ACCESS на является accessible-for-notify, оператор import генерируется в соответствии с приведенным ниже описанием.
 5. Если элемент используется в OBJECTS для NOTIFICATION-TYPE, оператор import генерируется в соответствии с приведенным ниже описанием.
 6. Если элемент используется в INDEX или AUGMENTS, оператор import генерируется в соответствии с приведенным ниже описанием.
 7. В остальных случаях элемент игнорируется. Примерами этого являются назначения OBJECT IDENTIFIER и объекты, указанные лишь в MODULE-COMPLIANCE, OBJECT-GROUP или NOTIFICATION-GROUP.
- Генерация дополнительных операторов import в соответствии с типом трансляции и таблицей отображения типов из приложения A. Это требует от транслятора рассмотрения всех типов, использованных в модуле SMlv2, для создания соответствующих операторов import.
 - Генерация оператора import для модуля YANG ietf-yang-smiv2 с префиксом smiv2.

Созданные операторы import используют в качестве аргументов нетранслированные имена SMlv2 или имена общеизвестных модулей YANG. Оператор import должен включать оператор prefix. Префиксы **можно** создавать путем применения алгоритма, описанного в приложении B.

3.1. Пример - импорт IF-MIB

Трансляция IF-MIB [RFC2863] создает модуль YANG с показанными ниже операторами namespace/prefix и import. Префиксом будет перевод имени модуля SMlv2 IF-MIB в нижний регистр (состоит из двух компонент и не требует дальнейшего сокращения).

```
module IF-MIB {
  namespace "urn:ietf:params:xml:ns:yang:smiv2:IF-MIB";
  prefix "if-mib";

  import IANAifType-MIB      { prefix "ianaiftype-mib"; }
  import SNMPv2-TC          { prefix "snmpv2-tc"; }
  import ietf-yang-types     { prefix "yang"; }
  import ietf-yang-smiv2    { prefix "smiv2"; }
}
```

4. Трансляция макроса MODULE-IDENTITY

SMlv2 требует вызова макроса MODULE-IDENTITY для предоставления контактных данных и истории выпусков модуля MIB. Операторы макроса SMlv2 MODULE-IDENTITY **должны** транслироваться в операторы YANG, как описано ниже.

4.1. Правила трансляции MODULE-IDENTITY

- SMlv2 ORGANIZATION отображается в оператор YANG organization.
- SMlv2 CONTACT-INFO отображается в оператор YANG contact.
- SMlv2 DESCRIPTION отображается в оператор YANG description.
- Каждый SMlv2 REVISION отображается в оператор YANG revision. Выпуск указывается аргументом date в SMlv2 REVISION. DESCRIPTION из REVISION отображаются в соответствующие операторы description, вложенные в revision.
- SMlv2 LAST-UPDATED игнорируется, если связанная дата соответствует REVISION. В остальных случаях создается дополнительный оператор revision.
- Создается контейнер YANG верхнего уровня. Именем контейнера служит имя модуля SMlv2 и для контейнера **должно** устанавливаться значение config false. Создание этого контейнера **можно** пропустить, если модуль SMlv2 не определяет никаких объектов для контейнера верхнего уровня (например, определяет лишь текстовые соглашения).
- Значение идентификатора объекта вызова SMlv2 MODULE-IDENTITY транслируется в оператор smiv2:oid, содержащий оператор smiv2:alias, который представляет вызов макроса MODULE-IDENTITY (см. расширение YANG, определенное в разделе 10).

Хотя все корректные модули SMlv2 должны включать точно один вызов макроса MODULE-IDENTITY, имеется несколько важных исключений. Модули, определяющие язык SMlv2 (т. е. SNMPv2-SMI, SNMPv2-TC и SNMPv2-CONF) не вызывают макрос MODULE-IDENTITY. Кроме того, модули SMlv2, созданные из модулей SMlv1 могут пропускать вызов макроса MODULE-IDENTITY. В таких случаях предпочтительно не создавать операторы organization, contact, description или revision.

4.2. Пример - MODULE-IDENTITY из IF-MIB

Трансляция MODULE-IDENTITY из IF-MIB [RFC2863] дает приведенные ниже операторы YANG.

```

organization
  "IETF Interfaces MIB Working Group";

contact
  "Keith McCloghrie
  Cisco Systems, Inc.
  170 West Tasman Drive
  San Jose, CA 95134-1706
  US

  408-526-5260
  kzm@cisco.com";

description
  "The MIB module to describe generic objects for network
  interface sub-layers. This MIB is an updated version of
  MIB-II's ifTable, and incorporates the extensions defined in
  RFC 1229.";

revision "2000-06-14" {
  description
    "Clarifications agreed upon by the Interfaces MIB WG, and
    published as RFC 2863.";
}
revision "1996-02-28" {
  description
    "Revisions made by the Interfaces MIB WG, and published in
    RFC 2233.";
}
revision "1993-11-08" {
  description
    "Initial revision, published as part of RFC 1573.";
}

container IF-MIB {
  config false;
}

```

5. Трансляция макроса TEXTUAL-CONVENTION

SMIv2 использует вызов макроса TEXTUAL-CONVENTION для определения типов, выведенных из базовых типов SMIv2. Вызовы TEXTUAL-CONVENTION **должны** транслироваться в операторы YANG typedef, как описано ниже.

5.1. Правила трансляции TEXTUAL-CONVENTION

Имя вызова макроса TEXTUAL-CONVENTION служит в качестве имени создаваемого оператора typedef. Операторы макроса SMIv2 TEXTUAL-CONVENTION отображаются в операторы YANG, вложенные в оператор typedef, как описано ниже.

- SMIv2 DISPLAY-HINT служит для определения отображений типов, выведенных из типа OCTET STRING, как описано в разделе 2. Кроме того, значение DISPLAY-HINT **может** служить для генерации регулярного выражения в операторе YANG pattern внутри оператора type.
- SMIv2 DISPLAY-HINT транслируется в оператор smiv2:display-hint (см. расширение YANG, определенное в разделе 10).
- SMIv2 STATUS отображается в оператор YANG status. Генерация YANG пропускается, если значение STATUS является текущим.
- SMIv2 DESCRIPTION отображается в оператор YANG description.
- SMIv2 REFERENCE отображается в оператор YANG reference.

Эта трансляция предполагает, что метки именованных чисел и битов не меняются в новых выпусках модуля SMIv2. Это согласуется с правилами выпуска модулей SMIv2, заданными в параграфе 4.9 [RFC4181].

5.2. Пример - OwnerString и InterfaceIndex из IF-MIB

Трансляция текстовых соглашений OwnerString и InterfaceIndex из модуля IF-MIB [RFC2863] показана ниже.

```

typedef OwnerString {
  type string {
- SMIv2 SYNTAX отображается в оператор YANG type. Ограничения диапазона SMIv2 отображаются в
  операторы YANG range, а ограничения размера SMIv2 - в операторы YANG length. Перечисляемые значения
  SMIv2 INTEGER отображаются в операторы YANG enum/value. SMIv2 BITS отображаются в операторы YANG
  bit/position. Для типов OCTET STRING отображаемый в базовый тип YANG string (см. раздел 2), размер,
  заданный в операторе YANG length, должен соответствовать строковому представлению значений. Если
  реализация не может вывести корректные ограничения размера, оператор YANG должен быть опущен.

    length "0..255";
    pattern '\p{IsBasicLatin}{0,255}';
  }
}

```

```

status deprecated;
description
  "This data type is used to model an administratively
  assigned name of the owner of a resource. This information
  is taken from the NVT ASCII character set. It is suggested
  that this name contain one or more of the following: ASCII
  form of the manager station's transport address, management
  station name (e.g., domain name), network management
  personnel's name, location, or phone number. In some cases
  the agent itself will be the owner of an entry. In these
  cases, this string shall be set to a string starting with
  'agent.'";
smiv2:display-hint "255a";
}

typedef InterfaceIndex {
  type int32 {
    range "1..2147483647";
  }
  description
    "A unique value, greater than zero, for each interface or
    interface sub-layer in the managed system. It is
    recommended that values are assigned contiguously starting
    from 1. The value for each interface sub-layer must remain
    constant at least from one re-initialization of the entity's
    network management system to the next re-initialization.";
smiv2:display-hint "d";
}

```

5.3. Пример - IfDirection из DIFFSERV-MIB

Трансляция текстового соглашения IfDirection из DIFFSERV-MIB [RFC3289] приведена ниже.

```

typedef IfDirection {
  type enumeration {
    enum inbound { value 1; }
    enum outbound { value 2; }
  }
  description
    "IfDirection specifies a direction of data travel on an
    interface. 'inbound' traffic is operated on during reception
    from the interface, while 'outbound' traffic is operated on
    prior to transmission on the interface.";
}

```

6. Трансляция назначений OBJECT IDENTIFIER

SMIv2 использует назначения OBJECT IDENTIFIER для ввода имен промежуточных узлов дерева OBJECT IDENTIFIER. Назначения OBJECT IDENTIFIER транслируются в операторы smiv2:alias (см. расширение YANG, определенное в разделе 10).

7. Трансляция макроса OBJECT-TYPE

SMIv2 использует макрос OBJECT-TYPE для определения объектов и структуры концептуальных таблиц. Объекты существуют в виде скаляров (в точности один экземпляр в контексте SNMP) или колонок с концептуальными таблицами (множество, возможно пустое, экземпляров в контексте SNMP). Ряд дополнительных объектов определяет индекс (ключ) концептуальной таблицы. Кроме того, концептуальные таблицы могут дополняться другими концептуальными таблицами. Все эти аспекты должны приниматься во внимание при трансляции вызовов макроса SMIv2 OBJECT-TYPE в YANG. Вызовы OBJECT-TYPE **должны** транслироваться в операторы YANG, как описано ниже.

- SMIv2 SYNTAX отображается в оператор YANG type. Ограничения диапазона SMIv2 отображаются в операторы YANG range, а ограничения размера SMIv2 length - в операторы YANG length. Перечисляемые значения SMIv2 INTEGER отображаются в операторы YANG enum/value, а SMIv2 BITS - в YANG bit/position. Для типов OCTET STRING, отображаемых в базовый тип YANG (раздел 2), размер, задаваемый в операторе YANG length, должен соответствовать строковому представлению значения. Если реализация не может вывести корректные ограничения размера, оператор YANG **должен** быть опущен.

7.1. Правила трансляции скаляров и колонок

Вызовы макроса SMIv2 OBJECT-TYPE, определяющие скаляры или колонки с MAX-ACCESS со значением not-accessible, read-only, read-write и read-create, транслируются в операторы YANG leaf. В дополнение к этому колонки с MAX-ACCESS = accessible-for-notify транслируются в операторы YANG leaf, если объект является частью INDEX для таблицы, содержащей колонку. Именем листа служит имя, связанное с вызовом макроса SMIv2 OBJECT-TYPE. Вызовы макроса SMIv2 OBJECT-TYPE с MAX-ACCESS = accessible-for-notify транслируются не в листья дерева данных YANG, а в листья уведомлений YANG.

Операторы leaf для скалярных объектов создаются в контейнере, представляющем родительский узел скаляра в дереве OID. Этот контейнер именуется после родительского узла скаляра в дереве OID и помещается в контейнер верхнего уровня, представляющий модуль SMIv2 (параграф 4.1). В редких случаях, когда родитель скаляра имеет множество имен, автоматическая трансляция **должна** давать отказ, а конфликт имен нужно изучить и исправить

вручную. Если в предыдущем выпуске модуля SMiv2 неоднозначности не было, **должно** применяться имя из этого выпуска. Операторы leaf, представляющие колонки, создаются в списке, представляющем концептуальную строку (параграф 7.3).

- SMiv2 UNITS отображается в оператор YANG units.
- SMiv2 MAX-ACCESS транслируется в оператор smiv2:max-access (см. расширение YANG, определенное в разделе 10).
- SMiv2 STATUS отображается в оператор YANG status. Генерация оператора YANG пропускается, если значение STATUS является текущим.
- SMiv2 DESCRIPTION отображается в оператор YANG description.
- SMiv2 REFERENCE отображается в оператор YANG reference.
- SMiv2 DEFVAL отображается в оператор smiv2:defval (см. расширение YANG, определенное в разделе 10).
- Значение макроса SMiv2 OBJECT-TYPE транслируется в оператор smiv2:oid (см. расширение YANG, определенное в разделе 10).

Эта трансляция предполагает, что именованные числа и биты не меняются при новом выпуске модуля SMiv2. Это согласуется с правилами выпуска модулей SMiv2, заданными в параграфе 4.9 [RFC4181].

7.2. Пример - ifNumber и ifIndex из IF-MIB

Трансляция скалярного объекта ifNumber и колонки ifIndex из IF-MIB [RFC2863] показана ниже. Поскольку ifNumber является скалярным объектом в ветви interfaces модуля IF-MIB, лист YANG ifNumber заменен контейнером YANG с именем interfaces, который регистрируется в контейнере верхнего уровня IF-MIB.

```
leaf ifNumber {
  type int32;
  description
    "The number of network interfaces (regardless of their
     current state) present on this system.";
  smiv2:max-access "read-only";
  smiv2:oid "1.3.6.1.2.1.2.1";
}

leaf ifIndex {
  type if-mib:InterfaceIndex;
  description
    "A unique value, greater than zero, for each interface. It
     is recommended that values are assigned contiguously
     starting from 1. The value for each interface sub-layer
     must remain constant at least from one re-initialization of
     the entity's network management system to the next re-
     initialization.";
  smiv2:max-access "read-only";
  smiv2:oid "1.3.6.1.2.1.2.2.1.1";
}
```

7.3. Правила трансляции нерасширяемых концептуальных таблиц

Вызов макроса OBJECT-TYPE, определяющего нерасширяемую концептуальную таблицу, транслируется в оператор YANG container с именем вызова макроса OBJECT-TYPE. Этот контейнер создается в контейнере верхнего уровня, представляющем модуль SMiv2. Правила трансляции макроса представлены ниже.

- SMiv2 SYNTAX игнорируется.
- SMiv2 UNITS игнорируется.
- SMiv2 MAX-ACCESS игнорируется.
- SMiv2 STATUS отображается в оператор YANG status. Генерация этого оператора пропускается, если значение STATUS является текущим.
- SMiv2 DESCRIPTION отображается в оператор YANG description.
- SMiv2 REFERENCE отображается в оператор YANG reference.
- Значение вызова макроса SMiv2 OBJECT-TYPE транслируется в оператор smiv2:oid (см. расширение YANG, определенное в разделе 10).

Вызов макроса OBJECT-TYPE, определяющего концептуальную строку, транслируется в оператор YANG list, содержащийся в контейнере YANG, представляющем концептуальную таблицу. Созданный список использует имя вызова макроса OBJECT-TYPE. Правила трансляции макроса представлены ниже.

- SMiv2 SYNTAX игнорируется.
- SMiv2 UNITS игнорируется.
- SMiv2 MAX-ACCESS игнорируется.
- SMiv2 STATUS отображается в оператор YANG status. Генерация этого оператора пропускается, если значение STATUS является текущим.

- SMIV2 DESCRIPTION отображается в оператор YANG description.
- SMIV2 REFERENCE отображается в оператор YANG reference.
- SMIV2 INDEX отображается в YANG key со списком колонок, формирующих ключ YANG list. Если один объект неоднократно указана в INDEX, к дубликатам добавляются суффиксы _<n>, где <n> порядковый номер экземпляра в INDEX, начиная с 2. Для определения создаваемых листьев должны использоваться дополнительные операторы leaf.
- Если SMIV2 INDEX содержит ключевое слово IMPLIED keyword, генерируется оператор smiv2:implied для записи имени объекта, которому предшествует ключевое слово IMPLIED (см. расширение YANG, определенное в разделе 10).
- Значение вызова макроса SMIV2 OBJECT-TYPE транслируется в оператор smiv2:oid (см. расширение YANG, определенное в разделе 10).

Внутри оператора list создаются операторы YANG leaf для колонок, как описано в параграфе 7.1. Для объектов, перечисленных в SMIV2 INDEX, которые не являются частью концептуальной таблицы, создаются операторы YANG leaf типа leafref, указывающие на определение.

7.4. Пример - ifTable из IF-MIB

Трансляция определения ifTable из модуля IF-MIB [RFC2863] показана ниже.

```

container ifTable {
  description
    "A list of interface entries. The number of entries is
     given by the value of ifNumber.";
  smiv2:oid "1.3.6.1.2.1.2.2";

  list ifEntry {
    key "ifIndex";
    description
      "An entry containing management information applicable to a
       particular interface.";
    smiv2:oid "1.3.6.1.2.1.2.2.1";

    leaf ifIndex {
      type if-mib:InterfaceIndex;
      description
        "A unique value, greater than zero, for each interface. It
         is recommended that values are assigned contiguously
         starting from 1. The value for each interface sub-layer
         must remain constant at least from one re-initialization of
         the entity's network management system to the next re-
         initialization.";
      smiv2:max-access "read-only";
      smiv2:oid "1.3.6.1.2.1.2.2.1.1";
    }

    // ...
  }
}

```

7.5. Пример - ifRcvAddressTable из IF-MIB

Ниже показана трансляция определения ifRcvAddressTable из модуля IF-MIB [RFC2863].

```

container ifRcvAddressTable {
  description
    "This table contains an entry for each address (broadcast,
     multicast, or uni-cast) for which the system will receive
     packets/frames on a particular interface, except as follows:

     - for an interface operating in promiscuous mode, entries are
       only required for those addresses for which the system would
       receive frames were it not operating in promiscuous mode.

     - for 802.5 functional addresses, only one entry is required,
       for the address which has the functional address bit ANDed
       with the bit mask of all functional addresses for which the
       interface will accept frames.

     A system is normally able to use any unicast address which
     corresponds to an entry in this table as a source address.";
  smiv2:oid "1.3.6.1.2.1.31.1.4";

  list ifRcvAddressEntry {
    key "ifIndex ifRcvAddressAddress";
    description
      "A list of objects identifying an address for which the

```

```

system will accept packets/frames on the particular
interface identified by the index value ifIndex.";
smiv2:oid "1.3.6.1.2.1.31.1.4.1";

leaf ifIndex {
  type leafref {
    path "/if-mib:IF-MIB/if-mib:ifTable" +
         "/if-mib:ifEntry/if-mib:ifIndex";
  }
}

leaf ifRcvAddressAddress {
  type yang:phys-address;
  description
    "An address for which the system will accept packets/frames
on this entry's interface.";
  smiv2:max-access "not-accessible";
  smiv2:oid "1.3.6.1.2.1.31.1.4.1.1";
}

// ...
}
}

```

7.6. Пример - alHostTable из RMON2-MIB

Ниже показана трансляция определения alHostTable из модуля RMON2-MIB [RFC4502].

```

container alHostTable {
  description
    "A collection of statistics for a particular protocol from a
particular network address that has been discovered on an
interface of this device.

The probe will populate this table for all protocols in the
protocol directory table whose value of
protocolDirHostConfig is equal to supportedOn(3), and
will delete any entries whose protocolDirEntry is deleted or
has a protocolDirHostConfig value of supportedOff(2).

The probe will add to this table all addresses
seen as the source or destination address in all packets with
no MAC errors and will increment octet and packet counts in
the table for all packets with no MAC errors. Further,
entries will only be added to this table if their address
exists in the nlHostTable and will be deleted from this table
if their address is deleted from the nlHostTable.";
  smiv2:oid "1.3.6.1.2.1.16.16.1";

  list alHostEntry {
    key "hlHostControlIndex alHostTimeMark protocolDirLocalIndex "
      + "nlHostAddress protocolDirLocalIndex_2";
    description
      "A conceptual row in the alHostTable.

The hlHostControlIndex value in the index identifies the
hlHostControlEntry on whose behalf this entry was created.
The first protocolDirLocalIndex value in the index identifies
the network-layer protocol of the address.
The nlHostAddress value in the index identifies the network-
layer address of this entry.
The second protocolDirLocalIndex value in the index identifies
the protocol that is counted by this entry.

An example of the indexing in this entry is
alHostOutPkts.1.783495.18.4.128.2.6.6.34.

Note that some combinations of index values may result in an
index that exceeds 128 sub-identifiers in length, which exceeds
the maximum for the SNMP protocol. Implementations should take
care to avoid such combinations.";
    smiv2:oid "1.3.6.1.2.1.16.16.1.1";

    // ...

    leaf protocolDirLocalIndex {
      type leafref {
        path "/rmon2-mib:RMON2-MIB/"

```

```

    + "rmon2-mib:protocolDirTable/"
    + "rmon2-mib:protocolDirEntry/"
    + "rmon2-mib:protocolDirLocalIndex";
  }
}

// ...

leaf protocolDirLocalIndex_2 {
  type leafref {
    path "/rmon2-mib:RMON2-MIB/"
      + "rmon2-mib:protocolDirTable/"
      + "rmon2-mib:protocolDirEntry/"
      + "rmon2-mib:protocolDirLocalIndex";
  }
}

// ...
}
}

```

7.7. Правила трансляции расширяемых концептуальных таблиц

Вызов макроса OBJECT-TYPE, определяющего расширяемую концептуальную таблицу, транслируется в оператор YANG `smiv2:alias` (см. расширение YANG, определенное в разделе 10). Правила трансляции макроса представлены ниже.

- SMIV2 SYNTAX игнорируется.
- SMIV2 UNITS игнорируется.
- SMIV2 MAX-ACCESS игнорируется.
- SMIV2 STATUS отображается в оператор YANG `status`. Генерация этого оператора пропускается, если значение STATUS является текущим.
- SMIV2 DESCRIPTION отображается в оператор YANG `description`.
- SMIV2 REFERENCE отображается в оператор YANG `reference`.
- Значение вызова макроса SMIV2 OBJECT-TYPE транслируется в оператор `smiv2:oid` (см. расширение YANG, определенное в разделе 10).

Вызов макроса OBJECT-TYPE, определяющего концептуальную строку расширения, транслируется в оператор YANG `smiv2:alias` и оператор YANG `augment`, использующий путь к расширенной таблице в качестве аргумента. Правила трансляции макроса представлены ниже.

- SMIV2 SYNTAX игнорируется.
- SMIV2 UNITS игнорируется.
- SMIV2 MAX-ACCESS игнорируется.
- SMIV2 STATUS отображается в оператор YANG `status`. Генерация этого оператора пропускается, если значение STATUS является текущим.
- SMIV2 DESCRIPTION отображается в оператор YANG `description`.
- SMIV2 REFERENCE отображается в оператор YANG `reference`.
- Значение вызова макроса SMIV2 OBJECT-TYPE транслируется в оператор `smiv2:oid` (см. расширение YANG, определенное в разделе 10).

Внутри оператора `augment` создаются операторы YANG `leaf`, как описано в параграфе 7.1.

7.8. Пример - ifXTable из IF-MIB

Ниже показана трансляция определения `ifXTable` из модуля IF-MIB [RFC2863].

```

smiv2:alias "ifXTable" {
  description
    "A list of interface entries. The number of entries is
     given by the value of ifNumber. This table contains
     additional objects for the interface table.";
  smiv2:oid "1.3.6.1.2.1.31.1.1";
}

smiv2:alias "ifXEntry" {
  description
    "An entry containing additional management information
     applicable to a particular interface.";
  smiv2:oid "1.3.6.1.2.1.31.1.1.1";
}

augment "/if-mib:IF-MIB/if-mib:ifTable/if-mib:ifEntry" {

```

```

description
  "An entry containing additional management information
  applicable to a particular interface.";
smiv2:oid "1.3.6.1.2.1.31.1.1.1";

leaf ifName {
  type snmpv2-tc:DisplayString;
  description
    "The textual name of the interface. The value of this
    object should be the name of the interface as assigned by
    the local device and should be suitable for use in commands
    entered at the device's `console'. This might be a text
    name, such as `le0' or a simple port number, such as `1',
    depending on the interface naming syntax of the device. If
    several entries in the ifTable together represent a single
    interface as named by the device, then each will have the
    same value of ifName. Note that for an agent which responds
    to SNMP queries concerning an interface on some other
    (proxied) device, then the value of ifName for such an
    interface is the proxied device's local name for it.

    If there is no local name, or this object is otherwise not
    applicable, then this object contains a zero-length string.";
  smiv2:max-access "read-only";
  smiv2:oid "1.3.6.1.2.1.31.1.1.1";
}

// ...
}

```

8. Трансляция макроса OBJECT-IDENTITY

SMIv2 использует вызовы макроса OBJECT-IDENTITY для определения информации о назначении OBJECT IDENTIFIER. Вызовы макроса OBJECT-IDENTITY **должны** транслироваться в операторы YANG identity, как описано ниже.

8.1. Правила трансляции OBJECT-IDENTITY

Имя вызова макроса OBJECT-IDENTITY служит именем созданного оператора identity, который использует в качестве базы отождествление smiv2:object-identity, определенное в разделе 10. Отображение SMIv2 OBJECT-IDENTITY в операторы YANG показано ниже.

- SMIv2 STATUS отображается в оператор YANG status. Этот оператор не создается, если значение STATUS является текущим.
- SMIv2 DESCRIPTION отображается в оператор YANG description.
- SMIv2 REFERENCE отображается в оператор YANG reference.
- Значение вызова макроса SMIv2 OBJECT-IDENTITY транслируется в оператор smiv2:oid (см. расширение YANG, определенное в разделе 10).

8.2. Пример - diffServTBParamSimpleTokenBucket из DIFFSERV-MIB

Ниже показана трансляция diffServTBParamSimpleTokenBucket из модуля DIFFSERV-MIB [RFC3289] (отметим, что описание должно следовать параграфу 5.1.3 в RFC 3290).

```

identity diffServTBParamSimpleTokenBucket {
  base "smiv2:object-identity";
  description
    "Two Parameter Token Bucket Meter as described in the Informal
    Differentiated Services Model section 5.2.3.";
  smiv2:oid "1.3.6.1.2.1.97.3.1.1";
}

```

9. Трансляция макроса NOTIFICATION-TYPE

SMIv2 обеспечивает макрос NOTIFICATION-TYPE для определения уведомлений о событиях. YANG для этих целей применяет оператор notification. Вызовы макроса NOTIFICATION-TYPE **должны** транслироваться в операторы YANG notification, как описано ниже.

9.1. Правила трансляции NOTIFICATION-TYPE

Имя вызова макроса NOTIFICATION-TYPE служит именем создаваемого оператора notification. Операторы макроса NOTIFICATION-TYPE отображаются в операторы YANG, встроенные в notification, как показано ниже.

- SMIv2 OBJECTS отображается в последовательность контейнеров YANG. Для каждого объекта, указанного в значении OBJECTS, создается оператор YANG container с именем object-<n>, где <n> указывает позицию элемента в значении OBJECTS (первый элемент имеет позицию 1). Если текущий объект относится к концептуальной таблице, создается последовательность операторов leaf для каждого объекта INDEX в концептуальной таблице. Эти листья именованы по объектам INDEX и имеют тип leafref. В заключение создается оператор leaf с именем текущего объекта. Если текущий объект имеет MAX-ACCESS со значением

read-only, read-write или read-create, создается лист типа leafref. В остальных случаях, если текущий объект имеет MAX-ACCESS = accessible-for-notify, лист создается в соответствии с параграфом 7.1.

- SMIV2 STATUS отображается в оператор YANG status. Этот оператор не создается, если значение STATUS является текущим.
- SMIV2 DESCRIPTION отображается в оператор YANG description.
- SMIV2 REFERENCE отображается в оператор YANG reference.
- Значение вызова макроса SMIV2 OBJECT-IDENTITY транслируется в оператор smiv2:oid (см. расширение YANG, определенное в разделе 10).

9.2. Пример - linkDown NOTIFICATION-TYPE из IF-MIB

Ниже показана трансляция уведомления linkDown из модуля IF-MIB [RFC2863].

```
notification linkDown {
  description
    "A linkDown trap signifies that the SNMP entity, acting in
    an agent role, has detected that the ifOperStatus object for
    one of its communication links is about to enter the down
    state from some other state (but not from the notPresent
    state). This other state is indicated by the included value
    of ifOperStatus.";
  smiv2:oid "1.3.6.1.6.3.1.1.5.3";

  container object-1 {
    leaf ifIndex {
      type leafref {
        path "/if-mib:IF-MIB/if-mib:ifTable" +
            "/if-mib:ifEntry/if-mib:ifIndex";
      }
    }
  }

  container object-2 {
    leaf ifIndex {
      type leafref {
        path "/if-mib:IF-MIB/if-mib:ifTable" +
            "/if-mib:ifEntry/if-mib:ifIndex";
      }
    }
    leaf ifAdminStatus {
      type leafref {
        path "/if-mib:IF-MIB/if-mib:ifTable" +
            "/if-mib:ifEntry/if-mib:ifAdminStatus";
      }
    }
  }

  container object-3 {
    leaf ifIndex {
      type leafref {
        path "/if-mib:IF-MIB/if-mib:ifTable" +
            "/if-mib:ifEntry/if-mib:ifIndex";
      }
    }
    leaf ifOperStatus {
      type leafref {
        path "/if-mib:IF-MIB/if-mib:ifTable" +
            "/if-mib:ifEntry/if-mib:ifOperStatus";
      }
    }
  }
}
```

10. Определение расширения языка YANG

В этом разделе определены некоторые операторы расширения YANG, которые могут служить для фиксации информации, присутствующей в модулях SMIV2, которая не транслируется с помощью базовых операторов YANG. Модуль YANG ссылается на [RFC2578] и [RFC2579].

```
<CODE BEGINS> file "ietf-yang-smiv2@2012-06-22.yang"
```

```
module ietf-yang-smiv2 {

  namespace "urn:ietf:params:xml:ns:yang:ietf-yang-smiv2";
  prefix "smiv2";
```

```
organization
  "IETF NETMOD (NETCONF Data Modeling Language) Working Group";
```

contact

```
"WG Web:   <http://tools.ietf.org/wg/netmod/>
WG List:   <mailto:netmod@ietf.org>

WG Chair:  David Kessens
           <mailto:david.kessens@nlnet.nl>

WG Chair:  Juergen Schoenwaelder
           <mailto:j.schoenwaelder@jacobs-university.de>

Editor:    Juergen Schoenwaelder
           <mailto:j.schoenwaelder@jacobs-university.de>";
```

description

"Этот модуль определяет расширения YANG, применяемые для трансляции SMIV2 в YANG.

Авторские права (Copyright (c) 2012) принадлежат IETF Trust и лицам, указанным как авторы. Все права защищены.

Распространение и использование в исходной или двоичной форме с изменениями или без таковых разрешено в соответствии с упрощенной лицензией BSD, как указано в параграфе 4.с документа IETF Trust Legal Provisions применительно к документам IETF (<http://trustee.ietf.org/license-info>).

Модуль является частью RFC 6643, где правовые аспекты указаны более детально.";

revision 2012-06-22 {

```
  description
    "Initial revision.";
  reference
    "RFC 6643: Translation of Structure of Management Information
    Version 2 (SMIV2) MIB Modules to YANG Modules";
}
```

identity object-identity {

```
  description
    "Базовое отождествление для всех SMIV2 OBJECT-IDENTITY.";
}
```

typedef opaque {

```
  type binary;
  description
    "Тип opaque поддерживает возможность передачи произвольного
    синтаксиса ASN.1. Значение кодируется с использованием ASN.1 BER1
    в строку октетов, которая кодируется в OCTET STRING, давая
    'двойное заворачивание' исходного значения ASN.1.

    По значению и его семантике этот тип эквивалентен типу Opaque в
    SMIV2. Этот тип применяется в SMIV2 исключительно для совместимости
    со старыми версиями, что сохраняется и для данного типа YANG.";
  reference
    "RFC 2578: Structure of Management Information Version 2 (SMIV2)";
}
```

extension display-hint {

```
  argument "format";
  description
    "Оператор display-hint принимает в качестве аргумента значение
    DISPLAY-HINT, назначенное текстовому соглашению SMIV2.";
  reference
    "RFC 2579: Textual Conventions for SMIV2";
}
```

extension max-access {

```
  argument "access";
  description
    "Оператор max-access принимает в качестве аргумента значение
    MAX-ACCESS, назначенное определению объекта SMIV2.
```

¹Basic Encoding Rules - базовые правила кодирования.

```

Значение MAX-ACCESS относится лишь к SMIV2 и не влияет на
доступ к объектам YANG по протоколам типа NETCONF.";
reference
"RFC 2578: Structure of Management Information Version 2 (SMIV2)";
}

extension defval {
  argument "value";
  description
  "Оператор defval принимает в качестве аргумента принятое по
  умолчанию значение, определенное в SMIV2 DEFVAL. Отметим, что
  значение находится в пространстве SMIV2, определенном синтаксисом
  SMIV2 соответствующего объекта, а не в пространстве YANG,
  определенном соответствующим типом данных YANG.";
  reference
  "RFC 2578: Structure of Management Information Version 2 (SMIV2)";
}

extension implied {
  argument "index";
  description
  "Если объекту SMIV2 INDEX предшествует ключевое слово IMPLIED,
  оператор implied присутствует в модуле YANG и принимает в
  качестве аргумента имя индексного объекта IMPLIED.";
  reference
  "RFC 2578: Structure of Management Information Version 2 (SMIV2)";
}

extension alias {
  argument "descriptor";
  description
  "Оператор alias вводит дескриптор SMIV2. Предполагается, что тело
  оператора alias содержит оператор oid, который указывает числовой
  идентификатор OID, связанный с дескриптором.";
  reference
  "RFC 2578: Structure of Management Information Version 2 (SMIV2)";
}

extension oid {
  argument "value";
  description
  "Оператор oid принимает в качестве аргумента идентификатор объекта,
  назначенный определению SMIV2. Значение идентификатора объекта
  указывается в десятичной нотации с разделением точками.";
  reference
  "RFC 2578: Structure of Management Information Version 2 (SMIV2)";
}

extension subid {
  argument "value";
  description
  "Оператор subid принимает в качестве аргумента последний
  субидентификатор идентификатора объекта, назначенного определению
  SMIV2. Значением субидентификатора является одно положительное
  десятичное натуральное число. Оператор subid не может служить
  субоператором какого-либо узла верхнего уровня в документе YANG.
  Оператор subid может быть лишь субоператором узла, имеющего
  родителя, определенного в субоператоре smiv2:oid или smiv2:subid.";
  reference
  "RFC 2578: Structure of Management Information Version 2 (SMIV2)";
}
}

}

<CODE ENDS>

```

11. Реализация конфигурационных узлов данных

Результат трансляции модулей SMIV2 MIB в модули YANG содержит объекты YANG, доступные лишь для чтения (config false), даже если объекты SMIV2 имеют доступ read-write или read-create. Одна из причин этого заключается в том, что модели постоянства базовых протоколов SNMP и NETCONF существенно различаются. В SNMP постоянство доступного для записи объекта зависит от определения объекта (т. е. текста в DESCRIPTION) или свойств постоянства концептуальной строки, к которой он относится, иногда управляемых через объект-колонку с использованием текстового соглашения StorageType. В NETCONF постоянство конфигурационных объектов определяется свойствами базового хранилища данных. Кроме того, NETCONF, в соответствии с определением [RFC6241], не обеспечивает стандартных операций для изменения рабочего состояния. Операции <edit-config> и <copy-config> работают только с

данными конфигурации. Поэтому невозможно создать узлы конфигурационных данных YANG из определений SMIV2 автоматически.

Однако для некоторых объектов SMIV2, где семантика постоянства SNMP и NETCONF согласована, разработчики могут реализовать некоторые узлы данных YANG, созданные из определений SMIV2, в качестве узлов данных конфигурации. Такие отклонения от доступных лишь для чтения модулей YANG следует формально документировать в виде отдельных модулей YANG с использованием оператора YANG deviation для изменения свойства config узлов данных, реализованный в качестве конфигурационных, с false на true. Отклонения, которые меняют свойство config false на true без других изменений семантики узла данных, не влияют на соответствие модуля YANG, созданного из SMIV2.

11.1. Пример - addressMapControlTable из RMON2-MIB

Приведенный ниже пример показывает, как некоторые колоночные объекты addressMapControlTable из модуля RMON2-MIB [RFC4502] можно преобразовать в узлы данных конфигурации YANG. Отметим, что отклонения YANG влияют лишь на свойство целевого узла и не наследуются.

```

module acme-RMON2-MIB-deviations {
  namespace "http://acme.example.com/RMON2-MIB-deviations";
  prefix "acme-rmon2-devs";

  import RMON2-MIB {
    prefix "rmon2-mib";
    revision-date 2006-05-02;
  }

  revision 2012-01-11 {
    description
      "First version.";
  }

  deviation "/rmon2-mib:RMON2-MIB" {
    deviate replace {
      config true;
    }
  }

  deviation "/rmon2-mib:RMON2-MIB/"
    + "rmon2-mib:addressMapControlTable" {
    deviate replace {
      config true;
    }
  }

  deviation "/rmon2-mib:RMON2-MIB/"
    + "rmon2-mib:addressMapControlTable/"
    + "rmon2-mib:addressMapControlEntry" {
    deviate replace {
      config true;
    }
  }

  deviation "/rmon2-mib:RMON2-MIB/"
    + "rmon2-mib:addressMapControlTable/"
    + "rmon2-mib:addressMapControlEntry/"
    + "rmon2-mib:addressMapControlIndex" {
    deviate replace {
      config true;
    }
  }

  deviation "/rmon2-mib:RMON2-MIB/"
    + "rmon2-mib:addressMapControlTable/"
    + "rmon2-mib:addressMapControlEntry/"
    + "rmon2-mib:addressMapControlDataSource" {
    deviate replace {
      config true;
    }
  }

  deviation "/rmon2-mib:RMON2-MIB/"
    + "rmon2-mib:addressMapControlTable/"
    + "rmon2-mib:addressMapControlEntry/"
    + "rmon2-mib:addressMapControlOwner" {
    deviate replace {
      config true;
    }
  }
}

```

Сервер NETCONF, реализующий модуль RMON2-MIB с такими отклонениями, будет анонсировать в сообщении <hello> приведенные ниже возможности (пробелы добавлены лишь для форматирования).

```
<capability>
  urn:ietf:params:xml:ns:yang:smiv2:RMON2-MIB?
  module=RMON2-MIB&
  revision=2006-05-02&
  deviations=acme-RMON2-MIB-deviations
</capability>
<capability>
  http://acme.example.com/RMON2-MIB-deviations?
  module=acme-RMON2-MIB-deviations&
  revision=2012-01-11
</capability>
```

12. Взаимодействие с IANA

Этот документ регистрирует два идентификатора URI в реестре IETF XML [RFC3688]. В соответствии с форматом RFC 3688 эти регистрации имеют вид.

```
URI: urn:ietf:params:xml:ns:yang:ietf-yang-smiv2
Registrant Contact: The NETMOD WG of the IETF.
XML: N/A, the requested URI is an XML namespace.
```

```
URI: urn:ietf:params:xml:ns:yang:smiv2
Registrant Contact: The NETMOD WG of the IETF.
XML: N/A, the requested URI is an XML namespace.
```

Документ регистрирует модуль YANG в реестре YANG Module Names [RFC6020].

```
Name:          ietf-yang-smiv2
Namespace:     urn:ietf:params:xml:ns:yang:ietf-yang-smiv2
Prefix:        smiv2
Reference:     RFC 6643
```

13. Вопросы безопасности

Этот документ определяет трансляцию модулей SMIv2 MIB в модули YANG, обеспечивающие лишь возможность чтения (config false) для объектов данных, определенных в SMIv2 MIB, по протоколу NETCONF. Сама трансляция не оказывает влияния на безопасность Internet.

Пользователям моделей данных YANG, созданных из моделей данных SMIv2, которые опубликованы в серии RFC, рекомендуется рассмотреть раздел, посвященный безопасности, в соответствующем RFC. Вопросы безопасности в RFC, содержащих модели данных SMIv2, указывают «чувствительные» и важные объекты. Пользователям NETCONF рекомендуется применять модель управления доступом NETCONF [RFC6536], которая позволяет задать правила для защиты конфиденциальной информации.

14. Благодарности

Автор благодарит за ценные замечания к предварительным вариантам документа Andy Bierman, Benoit Claise, Martin Bjorklund, Leif Johansson, David Reid, Dan Romascanu и David Spakes.

15. Литература

15.1. Нормативные документы

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, [RFC 2119](#), March 1997.
- [RFC2578] McCloghrie, K., Ed., Perkins, D., Ed., and J. Schoenwaelder, Ed., "Structure of Management Information Version 2 (SMIv2)", STD 58, RFC 2578, April 1999.
- [RFC2579] McCloghrie, K., Ed., Perkins, D., Ed., and J. Schoenwaelder, Ed., "Textual Conventions for SMIv2", STD 58, RFC 2579, April 1999.
- [RFC2580] McCloghrie, K., Perkins, D., and J. Schoenwaelder, "Conformance Statements for SMIv2", STD 58, RFC 2580, April 1999.
- [RFC6020] Bjorklund, M., Ed., "YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)", [RFC 6020](#)¹, October 2010.
- [RFC6021] Schoenwaelder, J., "Common YANG Data Types", [RFC 6021](#)², October 2010.

15.2. Дополнительная литература

- [RFC2863] McCloghrie, K. and F. Kastenholz, "The Interfaces Group MIB", RFC 2863, June 2000.
- [RFC3289] Baker, F., Chan, K., and A. Smith, "Management Information Base for the Differentiated Services Architecture", RFC 3289, May 2002.
- [RFC3584] Frye, R., Levi, D., Routhier, S., and B. Wijnen, "Coexistence between Version 1, Version 2, and Version 3 of the Internet-standard Network Management Framework", RFC 3584, August 2003.
- [RFC3688] Mealling, M., "The IETF XML Registry", BCP 81, [RFC 3688](#), January 2004.

¹Этот документ заменен [RFC 7950](#). Прим. перев.

²Этот документ заменен [RFC 6991](#). Прим. перев.

- [RFC4181] Heard, C., "Guidelines for Authors and Reviewers of MIB Documents", BCP 111, RFC 4181, September 2005.
- [RFC4502] Waldbusser, S., "Remote Network Monitoring Management Information Base Version 2", RFC 4502, May 2006.
- [RFC6241] Enns, R., Ed., Bjorklund, M., Ed., Schoenwaelder, J., Ed., and A. Bierman, Ed., "Network Configuration Protocol (NETCONF)", [RFC 6241](#), June 2011.
- [RFC6536] Bierman, A., Ed. and M. Bjorklund, Ed., "Network Configuration Protocol (NETCONF) Access Control Model", [RFC 6536](#), March 2012.

Приложение А. Отображение общеизвестных типов (нормативное)

В этом нормативном приложении описано отображение типов SMIV2 на типы YANG, полностью совместимое с таблицами 1 и 2 в [RFC6021].

```
SMIV2 Module: SNMPv2-SMI
SMIV2 Type:    INTEGER          (используется как перечисляемый тип)
YANG Type:    enumeration

SMIV2 Module: SNMPv2-SMI
SMIV2 Type:    INTEGER          (используется как численный тип)
YANG Type:    int32

SMIV2 Module: SNMPv2-SMI
SMIV2 Type:    Integer32
YANG Type:    int32

SMIV2 Module: SNMPv2-SMI
SMIV2 Type:    OCTET STRING    (используется как двоичная строка)
YANG Type:    binary

SMIV2 Module: SNMPv2-SMI
SMIV2 Type:    OCTET STRING    (используется как строка UTF-8 или ASCII)
YANG Type:    string

SMIV2 Module: SNMPv2-SMI
SMIV2 Type:    OBJECT IDENTIFIER
YANG Module:  ietf-yang-types
YANG Type:    object-identifier-128

SMIV2 Module: SNMPv2-SMI
SMIV2 Type:    BITS
YANG Type:    bits

SMIV2 Module: SNMPv2-SMI
SMIV2 Type:    IpAddress
YANG Module:  ietf-inet-types
YANG Type:    ipv4-address

SMIV2 Module: SNMPv2-SMI
SMIV2 Type:    Counter32
YANG Module:  ietf-yang-types
YANG Type:    counter32

SMIV2 Module: SNMPv2-SMI
SMIV2 Type:    Gauge32
YANG Module:  ietf-yang-types
YANG Type:    gauge32

SMIV2 Module: SNMPv2-SMI
SMIV2 Type:    TimeTicks
YANG Module:  ietf-yang-types
YANG Type:    timeticks

SMIV2 Module: SNMPv2-SMI
SMIV2 Type:    Counter64
YANG Module:  ietf-yang-types
YANG Type:    counter64

SMIV2 Module: SNMPv2-SMI
SMIV2 Type:    Unsigned32
YANG Type:    uint32

SMIV2 Module: SNMPv2-SMI
SMIV2 Type:    Opaque
YANG Module:  ietf-yang-smiv2
YANG Type:    opaque
```

```

SMIv2 Module: SNMPv2-TC
SMIv2 Type: PhysAddress
YANG Module: ietf-yang-types
YANG Type: phys-address

SMIv2 Module: SNMPv2-TC
SMIv2 Type: MacAddress
YANG Module: ietf-yang-types
YANG Type: mac-address

SMIv2 Module: SNMPv2-TC
SMIv2 Type: TruthValue
YANG Type: boolean

SMIv2 Module: SNMPv2-TC
SMIv2 Type: TimeStamp
YANG Module: ietf-yang-types
YANG Type: timestamp

SMIv2 Module: RMON2-MIB
SMIv2 Type: ZeroBasedCounter32
YANG Module: ietf-yang-types
YANG Type: zero-based-counter32

SMIv2 Module: HCNUM-TC
SMIv2 Type: ZeroBasedCounter64
YANG Module: ietf-yang-types
YANG Type: zero-based-counter64

SMIv2 Module: HCNUM-TC
SMIv2 Type: CounterBasedGauge64
YANG Module: ietf-yang-types
YANG Type: gauge64

SMIv2 Module: INET-ADDRESS-MIB
SMIv2 Type: InetAutonomousSystemNumber
YANG Module: ietf-inet-types
YANG Type: as-number

SMIv2 Module: INET-ADDRESS-MIB
SMIv2 Type: InetVersion
YANG Module: ietf-inet-types
YANG Type: ip-version

SMIv2 Module: INET-ADDRESS-MIB
SMIv2 Type: InetPortNumber
YANG Module: ietf-inet-types
YANG Type: port-number

SMIv2 Module: DIFFSERV-DSCP-TC
SMIv2 Type: Dscp
YANG Module: ietf-inet-types
YANG Type: dscp

SMIv2 Module: IPV6-FLOW-LABEL-MIB
SMIv2 Type: IPv6FlowLabel
YANG Module: ietf-inet-types
YANG Type: ipv6-flow-label

SMIv2 Module: URI-TC-MIB
SMIv2 Type: Uri
YANG Module: ietf-inet-types
YANG Type: uri

```

Приложение В. Генерация префиксов (информационное)

В этом приложении описан алгоритм создания префиксов для использования в операторах `import`. Входом алгоритма является набор префиксов (обычно выводятся из имен импортируемых модулей) и имена конкретных модулей преобразуются в префиксы. Описанный ниже алгоритм создает префикс для данного имени модуля, который будет уникален в наборе префиксов.

Таблица 1. Специальные префиксы для общеизвестных модулей YANG.

Модуль YANG	Префикс
ietf-yang-types	yang
ietf-inet-types	inet
ietf-yang-smiv2	smiv2

- Сначала предпринимаются попытки отображения некоторых общеизвестных модулей YANG в короткие префиксы (таблица 1). При наличии правила фиксированной трансляции, не создающей конфликтов имен, используется фиксированная трансляция.
- В остальных случаях префиксы создаются на основе имени модуля YANG с использованием дефиса в качестве разделителя компонент и приведением символов имени к нижнему регистру. Префиксом становится кратчайшая последовательность компонент, разделенных символами дефиса, которая включает не менее двух компонент и обеспечивает уникальность среди используемых в модуле YANG префиксов.

В худшем случае префиксом модуля, созданного из имени модуля SMIv2, становится полное имя этого модуля, приведенное к нижнему регистру. Но обычно создаются более короткие префиксы.

Адрес автора**Juergen Schoenwaelder**

Jacobs University

EMail: j.schoenwaelder@jacobs-university.de**Перевод на русский язык****Николай Малых**nmalykh@protocols.ru