

Internet Engineering Task Force (IETF)
Request for Comments: 8347
Category: Standards Track
ISSN: 2070-1721

X. Liu, Ed.
A. Kyparlis
Jabil
R. Parikh
VMware
A. Lindem
Cisco Systems
M. Zhang
Huawei Technologies
March 2018

A YANG Data Model for the Virtual Router Redundancy Protocol (VRRP)

Модель данных YANG для протокола резервирования виртуальных маршрутизаторов (VRRP)

Аннотация

Этот документ описывает модель данных для протокола резервирования виртуальных маршрутизаторов (VRRP) версий 2 и 3.

Статус документа

Документ относится к категории Internet Standards Track.

Документ является результатом работы IETF¹ и представляет согласованный взгляд сообщества IETF. Документ прошёл открытое обсуждение и был одобрен для публикации IESG². 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/rfc8347>.

Авторские права

Copyright (c) 2018. Авторские права принадлежат IETF Trust и лицам, указанным в качестве авторов документа. Все права защищены.

К документу применимы права и ограничения, указанные в BCP 78 и IETF Trust Legal Provisions и относящиеся к документам IETF (<http://trustee.ietf.org/license-info>), на момент публикации данного документа. Прочтите упомянутые документы внимательно. Фрагменты программного кода, включённые в этот документ, распространяются в соответствии с упрощённой лицензией BSD, как указано в параграфе 4.e документа IETF Trust Legal Provisions, без каких-либо гарантий (как указано в Simplified BSD License).

Оглавление

1. Введение.....	1
1.1. Терминология.....	2
1.2. Диаграммы деревьев.....	2
1.3. Префиксы в именах узлов данных.....	2
2. Устройство модели данных.....	2
2.1. Область действия модели.....	2
2.2. Связи с моделями интерфейса и IP.....	2
2.3. Конфигурация протокола.....	3
2.4. Состояния протокола.....	3
2.5. Уведомления.....	4
3. Структура дерева.....	5
4. Модуль YANG.....	6
5. Взаимодействие с IANA.....	18
6. Вопросы безопасности.....	19
7. Литература.....	19
7.1. Нормативные документы.....	19
7.2. Дополнительная литература.....	20
Приложение А. Пример дерева данных.....	20
Адреса авторов.....	22

1. Введение

Этот документ добавляет модель данных YANG [RFC6020] [RFC7950] для протокола VRRP [RFC3768] [RFC5798], обеспечивающего отказоустойчивость за счёт указания протокола выбора, который динамически назначает ответственность за виртуальный маршрутизатор одному из маршрутизаторов VRRP в ЛВС.

Заданный здесь модуль YANG поддерживает версии 2 и 3 протокола VRRP. Версия VRRP 2 (задана в [RFC3768]) поддерживает IPv4, версия 3 (задана в [RFC5798]) - IPv4 и IPv6.

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

1.1. Терминология

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.

Ниже указаны термины, определённые в [RFC7950] и не переопределяемые здесь

- augment - дополнение;
- data model - модель данных;
- data node - узел данных.

1.2. Диаграммы деревьев

В документе применяется графическое представление моделей данных, описанное в [RFC8340].

1.3. Префиксы в именах узлов данных

В этом документе имена узлов данных, действий и других объектов модели часто указываются без префиксов, если из контекста понятно, в каком модуле YANG они определены. В остальных случаях применяется префикс соответствующего модуля YANG, как показано в таблице 1.

Таблица 1. Префиксы и соответствующие модули YANG.

Префикс	Модуль YANG	Документ
yang	ietf-yang-types	[RFC6991]
inet	ietf-inet-types	[RFC6991]
if	ietf-interfaces	[RFC8343]
ip	ietf-ip	[RFC8344]

2. Устройство модели данных

2.1. Область действия модели

Модель охватывает VRRP версии 2 [RFC3768] и VRRP версии 3 [RFC5798]. Модель предназначена для реализации на устройствах, где реализован протокол VRRP версии 2 или 3. С подходящим протоколом управления модель может служить:

- для настройки VRRP версии 2 или 3;
- для управления поведением операций протоколов;
- для определения рабочего состояния протокола;
- для получения протокольных уведомлений.

2.2. Связи с моделями интерфейса и IP

Эта модель дополняет модели данных интерфейса ietf-interfaces [RFC8343] и управления IP ietf-ip [RFC8344]. Связи и дополнения показаны ниже.

```

module: ietf-interfaces
  +--rw interfaces
    +--rw interface* [name]
      ...
      +--rw ip:ipv4!
        | +--rw ip:address* [ip]
        | ...
        | +--rw vrrp:vrrp
        |   +--rw vrrp:vrrp-instance* [vrid]
        |   +--rw vrrp:vrid
        |   +--rw vrrp:virtual-ipv4-addresses
        |   ...
      +--rw ip:ipv6!
        +--rw ip:address* [ip]
        +--rw vrrp:vrrp
          +--rw vrrp:vrrp-instance* [vrid]
          +--rw vrrp:vrid
          +--rw vrrp:virtual-ipv6-addresses
          ...
  
```

В приведённом выше дереве узел без префикса взят из модели ietf-interfaces, узел с префиксом ip: - из модели ietf-ip, узел с префиксом vrrp: - из описанной здесь модели VRRP.

Контейнер vrrp содержит список узлов vrrp-instance, которые создаются в ветви интерфейса для заданного семейства адресов (IPv4 или IPv6). Каждый узел vrrp-instance представляет состояние конечного автомата маршрутизатора VRRP, как описано в параграфе 6.4 [RFC5798], обеспечивая сведения о конфигурации и состоянии для процесса выбора виртуального маршрутизатора. Адреса IP добавленного интерфейса являются реальными адресами, через которые работает маршрутизатор VRRP. Адрес IPv4 или IPv6 или адреса, связанные с виртуальным маршрутизатором (раздел 1 в [RFC5798]) моделируются как список адресов IPv4 или IPv6 в ветви vrrp-instance.

2.3. Конфигурация протокола

Структура модели для настройки протокола показана ниже.

```

augment /if:interfaces/if:interface/ip:ipv4:
  +--rw vrrp
    +--rw vrrp-instance* [vrid]
      +--rw vrid                               uint8
      |   ...
      +--rw track
        | +--rw interfaces
        | | +--rw interface* [interface]
        | |   +--rw interface                 if:interface-ref
        | |   ...
        | +--rw networks
        |   +--rw network* [prefix]
        |   +--rw prefix                       inet:ipv4-prefix
        |   ...
      +--rw virtual-ipv4-addresses
        +--rw virtual-ipv4-address* [ipv4-address]
        +--rw ipv4-address                 inet:ipv4-address

augment /if:interfaces/if:interface/ip:ipv6:
  +--rw vrrp
    +--rw vrrp-instance* [vrid]
      +--rw vrid                               uint8
      |   ...
      +--rw track
        | +--rw interfaces
        | | +--rw interface* [interface]
        | |   +--rw interface                 if:interface-ref
        | |   ...
        | +--rw networks
        |   +--rw network* [prefix]
        |   +--rw prefix                       inet:ipv6-prefix
        |   ...
      +--rw virtual-ipv6-addresses
        +--rw virtual-ipv6-address* [ipv6-address]
        +--rw ipv6-address                 inet:ipv6-address

```

Модель позволяет настраивать:

- экземпляр VRRP (версии 2 или 3), представляющий маршрутизатор VRRP;
- виртуальный адрес IPv4 или IPv6, связанный с виртуальным маршрутизатором;
- интерфейс отслеживания для обнаружения отказов связности интерфейса;
- сеть отслеживания для обнаружения отказов связности с сетью.

2.4. Состояния протокола

Ниже показана структура модели для состояний протокола.

```

module: ietf-vrrp
  +--ro vrrp
    | // global operational states
    +--ro virtual-routers?  uint32
    +--ro interfaces?      uint32
    +--ro statistics       // global statistics
      +--ro discontinuity-datetime?  yang:date-and-time
      +--ro checksum-errors?         yang:counter64
      +--ro version-errors?          yang:counter64
      +--ro vrid-errors?             yang:counter64
      +--ro ip-ttl-errors?           yang:counter64

augment /if:interfaces/if:interface/ip:ipv4:
  +--rw vrrp
    +--rw vrrp-instance* [vrid]
      +--rw vrid                               uint8
      |   ...
      +--rw track
        | +--rw interfaces
        | | +--rw interface* [interface]
        | |   +--rw interface                 if:interface-ref
        | |   ...
        | +--rw networks
        |   +--rw network* [prefix]
        |   +--rw prefix                       inet:ipv4-prefix
        |   ...
      +--rw virtual-ipv4-addresses
        | +--rw virtual-ipv4-address* [ipv4-address]
        | +--rw ipv4-address                 inet:ipv4-address
        |
        | // per-instance operational states
      +--ro state?                            identityref
      +--ro is-owner?                          boolean
      +--ro last-adv-source?                    inet:ip-address

```

```

+--ro up-datetime?          yang:date-and-time
+--ro master-down-interval? uint32
+--ro skew-time?           uint32
+--ro last-event?         identityref
+--ro new-master-reason?   new-master-reason-type
+--ro statistics           // per-instance statistics
  +--ro discontinuity-datetime? yang:date-and-time
  +--ro master-transitions?    yang:counter32
  +--ro advertisement-rcvd?    yang:counter64
  +--ro advertisement-sent?    yang:counter64
  +--ro interval-errors?      yang:counter64
  | {validate-interval-errors}?
  +--ro priority-zero-pkts-rcvd? yang:counter64
  +--ro priority-zero-pkts-sent? yang:counter64
  +--ro invalid-type-pkts-rcvd? yang:counter64
  +--ro address-list-errors?   yang:counter64
  | {validate-address-list-errors}?
  +--ro packet-length-errors?  yang:counter64

augment /if:interfaces/if:interface/ip:ipv6:
+--rw vrrp
  +--rw vrrp-instance* [vrid]
    +--rw vrid          uint8
    + ...
  +--rw track
    | +--rw interfaces
    | | +--rw interface* [interface]
    | | +--rw interface  if:interface-ref
    | | ...
    | +--rw networks
    |   +--rw network* [prefix]
    |   +--rw prefix    inet:ipv6-prefix
    |   ...
  +--rw virtual-ipv6-addresses
    | +--rw virtual-ipv6-address* [ipv6-address]
    | +--rw ipv6-address          inet:ipv6-address
    |
    | // per-instance operational states
  +--ro state?          identityref
  +--ro is-owner?       boolean
  +--ro last-adv-source? inet:ip-address
  +--ro up-datetime?    yang:date-and-time
  +--ro master-down-interval? uint32
  +--ro skew-time?     uint32
  +--ro last-event?    identityref
  +--ro new-master-reason? new-master-reason-type
  +--ro statistics     // per-instance statistics
    +--ro discontinuity-datetime? yang:date-and-time
    +--ro master-transitions?    yang:counter32
    +--ro advertisement-rcvd?    yang:counter64
    +--ro advertisement-sent?    yang:counter64
    +--ro interval-errors?      yang:counter64
    | {validate-interval-errors}?
    +--ro priority-zero-pkts-rcvd? yang:counter64
    +--ro priority-zero-pkts-sent? yang:counter64
    +--ro invalid-type-pkts-rcvd? yang:counter64
    +--ro address-list-errors?   yang:counter64
    | {validate-address-list-errors}?
    +--ro packet-length-errors?  yang:counter64

```

Эта модель соответствует архитектуре хранилищ данных управления сетью (Network Management Datastore Architecture или NMDA) [RFC8342]. Данные рабочего состояния объединяются с соответствующими данными конфигурации в одной иерархии [YANG-Guidelines]. Когда состояния протокола извлекаются из рабочего хранилища NMDA, охватываются все узлы config true (rw) и config false (ro), определённые в схеме.

Модель позволяет извлекать состояния протокола на указанных ниже уровнях.

- Экземпляр VRRP (версии 2 или 3), представляющий маршрутизатор VRRP.
- Виртуальный адрес IPv4 или IPv6, связанный с виртуальным маршрутизатором.
- Интерфейс отслеживания для обнаружения отказов связности интерфейса.
- Сеть отслеживания для обнаружения отказов связности с сетью.
- Глобальные состояния и статистика со сводкой для всех экземпляров.

2.5. Уведомления

Эта модель определяет указанные ниже уведомления, связанные с VRRP.

```

notifications:
+---n vrrp-new-master-event
| +--ro master-ip-address  inet:ip-address
| +--ro new-master-reason  new-master-reason-type
+---n vrrp-protocol-error-event
| +--ro protocol-error-reason  identityref
+---n vrrp-virtual-router-error-event

```

```

+--ro interface                               if:interface-ref
+--ro (ip-version)
| +--:(ipv4)
| | +--ro ipv4
| |   +--ro vrid      leafref
| +--:(ipv6)
|   +--ro ipv6
|     +--ro vrid      leafref
+--ro virtual-router-error-reason  identityref

```

Каждый тип уведомления служит для указания типа смены состояния VRRP или возникновения ошибки.

vrrp-new-master-event

Выбран новый ведущий VRRP.

vrrp-protocol-error-event

Ошибка протокола VRRP для сообщения, которое не может достичь экземпляра VRRP для своей обработки.

vrrp-virtual-router-error-event

Ошибка виртуального маршрутизатора VRRP для сообщения, обрабатываемого экземпляром VRRP.

В дополнение к указанным уведомлениям могут применяться механизмы, определённые в [Subscribed-Notifications] и [YANG-Push], которые позволяют пользователю:

- подписаться на уведомления на уровне клиента;
- задать фильтры ветвей XML Path Language (XPath) для отправки лишь интересующего содержимого;
- задать периодические уведомления или уведомления по запросу.

3. Структура дерева

Дерево модели данных YANG для VRRP, заданной этим документом, показано ниже.

```

module: ietf-vrrp
+--ro vrrp
+--ro virtual-routers?  uint32
+--ro interfaces?      uint32
+--ro statistics
  +--ro discontinuity-datetime?  yang:date-and-time
  +--ro checksum-errors?        yang:counter64
  +--ro version-errors?         yang:counter64
  +--ro vrid-errors?           yang:counter64
  +--ro ip-ttl-errors?         yang:counter64
augment /if:interfaces/if:interface/ip:ipv4:
+--rw vrrp
+--rw vrrp-instance* [vrid]
+--rw vrid                uint8
+--rw version             identityref
+--rw log-state-change?   boolean
+--rw preempt
| +--rw enabled?         boolean
| +--rw hold-time?      uint16
+--rw priority?          uint8
+--rw accept-mode?       boolean
+--rw (advertise-interval-choice)?
| +--:(v2)
| | +--rw advertise-interval-sec?      uint8
| +--:(v3)
|   +--rw advertise-interval-centi-sec? uint16
+--rw track
| +--rw interfaces
| | +--rw interface* [interface]
| |   +--rw interface      if:interface-ref
| |   +--rw priority-decrement?  uint8
| +--rw networks
|   +--rw network* [prefix]
|     +--rw prefix         inet:ipv4-prefix
|     +--rw priority-decrement?  uint8
+--rw virtual-ipv4-addresses
| +--rw virtual-ipv4-address* [ipv4-address]
|   +--rw ipv4-address      inet:ipv4-address
+--ro state?                identityref
+--ro is-owner?             boolean
+--ro last-adv-source?      inet:ip-address
+--ro up-datetime?         yang:date-and-time
+--ro master-down-interval? uint32
+--ro skew-time?           uint32
+--ro last-event?          identityref
+--ro new-master-reason?
new-master-reason-type
+--ro statistics
  +--ro discontinuity-datetime?  yang:date-and-time
  +--ro master-transitions?     yang:counter32
  +--ro advertisement-rcvd?     yang:counter64
  +--ro advertisement-sent?     yang:counter64
  +--ro interval-errors?       yang:counter64
  | {validate-interval-errors}?
  +--ro priority-zero-pkts-rcvd? yang:counter64
  +--ro priority-zero-pkts-sent? yang:counter64

```

```

+--ro invalid-type-pkts-rcvd? yang:counter64
+--ro address-list-errors?   yang:counter64
|   {validate-address-list-errors}?
+--ro packet-length-errors?  yang:counter64
augment /if:interfaces/if:interface/ip:ipv6:
+--rw vrrp
+--rw vrrp-instance* [vrid]
+--rw vrid                uint8
+--rw version              identityref
+--rw log-state-change?   boolean
+--rw preempt
|   +--rw enabled?        boolean
|   +--rw hold-time?     uint16
+--rw priority?           uint8
+--rw accept-mode?        boolean
+--rw advertise-interval-centi-sec? uint16
+--rw track
|   +--rw interfaces
|   |   +--rw interface* [interface]
|   |   |   +--rw interface          if:interface-ref
|   |   |   +--rw priority-decrement? uint8
|   +--rw networks
|   |   +--rw network* [prefix]
|   |   |   +--rw prefix              inet:ipv6-prefix
|   |   |   +--rw priority-decrement? uint8
+--rw virtual-ipv6-addresses
|   +--rw virtual-ipv6-address* [ipv6-address]
|   |   +--rw ipv6-address          inet:ipv6-address
+--ro state?                    identityref
+--ro is-owner?                  boolean
+--ro last-adv-source?           inet:ip-address
+--ro up-datetime?               yang:date-and-time
+--ro master-down-interval?     uint32
+--ro skew-time?                 uint32
+--ro last-event?                identityref
+--ro new-master-reason?
new-master-reason-type
+--ro statistics
+--ro discontinuity-datetime?    yang:date-and-time
+--ro master-transitions?       yang:counter32
+--ro advertisement-rcvd?       yang:counter64
+--ro advertisement-sent?       yang:counter64
+--ro interval-errors?          yang:counter64
|   {validate-interval-errors}?
+--ro priority-zero-pkts-rcvd?  yang:counter64
+--ro priority-zero-pkts-sent?  yang:counter64
+--ro invalid-type-pkts-rcvd?   yang:counter64
+--ro address-list-errors?      yang:counter64
|   {validate-address-list-errors}?
+--ro packet-length-errors?     yang:counter64

notifications:
+---n vrrp-new-master-event
|   +--ro master-ip-address      inet:ip-address
|   +--ro new-master-reason      new-master-reason-type
+---n vrrp-protocol-error-event
|   +--ro protocol-error-reason  identityref
+---n vrrp-virtual-router-error-event
+--ro interface                  if:interface-ref
+--ro (ip-version)
|   +---: (ipv4)
|   |   +--ro ipv4
|   |   |   +--ro vrid          leafref
|   +---: (ipv6)
|   |   +--ro ipv6
|   |   |   +--ro vrid          leafref
+--ro virtual-router-error-reason identityref

```

4. Модуль YANG

Этот модуль ссылается на [RFC2787], [RFC3768], [RFC5798], [RFC6527].

```
<CODE BEGINS> file "ietf-vrrp@2018-03-13.yang"
```

```

module ietf-vrrp {
  yang-version 1.1;
  namespace "urn:ietf:params:xml:ns:yang:ietf-vrrp";
  prefix "vrrp";

  import ietf-inet-types {
    prefix "inet";
  }

  import ietf-yang-types {
    prefix "yang";
  }

```

```

import ietf-interfaces {
  prefix "if";
}

import ietf-ip {
  prefix "ip";
}

organization
  "IETF Routing Area Working Group (RTGWG)";
contact
  "WG Web: <https://datatracker.ietf.org/wg/rtgwg/>
  WG List: <mailto:rtgwg@ietf.org>

  Editor: Xufeng Liu
         <mailto:xufeng.liu.ietf@gmail.com>

  Editor: Athanasios Kyparlis
         <mailto:Athanasios\_Kyparlis@jabil.com>
  Editor: Ravi Parikh
         <mailto:parikhr@vmware.com>

  Editor: Acee Lindem
         <mailto:acee@cisco.com>

  Editor: Mingui Zhang
         <mailto:zhangmingui@huawei.com>";

description
  "Этот модуль YANG задаёт модель для управления протоколом VRRP
  версий 2 и 3.

  Авторские права (Copyright (c) 2018) 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 8347, где
  правовые вопросы рассмотрены более полно.";

revision 2018-03-13 {
  description
    "Исходный выпуск.";
  reference
    "RFC 8347: A YANG Data Model for the Virtual Router Redundancy
    Protocol (VRRP)
    RFC 2787: Definitions of Managed Objects for the Virtual
    Router Redundancy Protocol
    RFC 3768: Virtual Router Redundancy Protocol (VRRP)
    RFC 5798: Virtual Router Redundancy Protocol (VRRP)
    Version 3 for IPv4 and IPv6
    RFC 6527: Definitions of Managed Objects for the Virtual
    Router Redundancy Protocol Version 3 (VRRPv3)";
}

/*
 * Свойства (функции)
 */

feature validate-interval-errors {
  description
    "Указывает, что система проверяет, соответствует ли интервал
    анонсов в полученных пакетах интервалу, заданному для
    локального маршрутизатора VRRP.";
}

feature validate-address-list-errors {
  description
    "Указывает, что система проверяет соответствие списка адресов в
    полученных пакетах настроенному локально списку для
    маршрутизатора VRRP.";
}

/*
 * Определения типов
 */
typedef new-master-reason-type {
  type enumeration {
    enum not-master {
      description
        "Виртуальный маршрутизатор никогда не был ведущим.";
    }
  }
}

```

```
enum priority {
    description
        "Приоритет был выше.";
}
enum preempted {
    description
        "Ведущий (master) был вытеснен (сменён).";
}
enum no-response {
    description
        "Прежний ведущий не отвечает.";
}
}
description
    "Указывает переход виртуального маршрутизатора в
    состояние master.";
} // new-master-reason-type

/*
 * Отождествления (идентификаторы)
 */

/* vrrp-event-type и производные от него. */
identity vrrp-event-type {
    description
        "Тип события протокола VRRP.";
}
identity vrrp-event-none {
    base vrrp-event-type;
    description
        "Незначимое событие.";
}
identity vrrp-event-startup {
    base vrrp-event-type;
    description
        "Экземпляр маршрутизатора VRRP создан протоколом.";
}
identity vrrp-event-shutdown {
    base vrrp-event-type;
    description
        "Маршрутизатор VRRP закрыт (down) протоколом.";
}
identity vrrp-event-higher-priority-backup {
    base vrrp-event-type;
    description
        "Резервный (backup) маршрутизатор имеет приоритет выше, чем
        имеющийся ведущий (master).";
}
identity vrrp-event-master-timeout {
    base vrrp-event-type;
    description
        "Текущий master не передал анонса в интервале
        master-down-interval.";
}
identity vrrp-event-interface-up {
    base vrrp-event-type;
    description
        "Интерфейс со включенным VRRP перешёл в состояние
        operational up.";
}
identity vrrp-event-interface-down {
    base vrrp-event-type;
    description
        "Интерфейс со включенным VRRP перешёл в состояние
        operational down.";
}
identity vrrp-event-no-primary-ip-address {
    base vrrp-event-type;
    description
        "Основной адрес IP на интерфейсе с VRRP утратил доступность.";
}
identity vrrp-event-primary-ip-address {
    base vrrp-event-type;
    description
        "Основной адрес IP на интерфейсе с VRRP стал доступным.";
}
identity vrrp-event-no-virtual-ip-addresses {
    base vrrp-event-type;
    description
        "На виртуальном маршрутизаторе нет виртуального адреса IP.";
}
identity vrrp-event-virtual-ip-addresses {
    base vrrp-event-type;
    description
        "На виртуальном маршрутизаторе есть виртуальные адреса IP.";
}
}
```



```
identity vrrp-event-preempt-hold-timeout {
    base vrrp-event-type;
    description
        "Указывает, что заданное время удержания вытеснения прошло.";
}
identity vrrp-event-lower-priority-master {
    base vrrp-event-type;
    description
        "Указывает наличие VRRP master с меньшим приоритетом.";
}
identity vrrp-event-owner-preempt {
    base vrrp-event-type;
    description
        "Владелец сделал ведущим (master) другой маршрутизатор.";
}

/* vrrp-error-global и производные от него. */
identity vrrp-error-global {
    description
        "Тип ошибки VRRP, возникшей для пакета до попадания
        на маршрутизатор VRRP.";
}
identity checksum-error {
    base vrrp-error-global;
    description
        "Получен пакет с ошибкой контрольной суммы VRRP.";
}
identity ip-ttl-error {
    base vrrp-error-global;
    description
        "Получен пакет с IP TTL (Time-To-Live), отличным от 255.";
}
identity version-error {
    base vrrp-error-global;
    description
        "Получен пакет неизвестной или неподдерживаемой версии.";
}
identity vrid-error {
    base vrrp-error-global;
    description
        "Получен пакет с идентификатором виртуального маршрутизатора
        (VRID), не действительным на этом маршрутизаторе.";
}

/* vrrp-error-virtual-router и производные от него. */
identity vrrp-error-virtual-router {
    description
        "Ошибка VRRP, возникшая после попадания пакета на
        маршрутизатор VRRP.";
}
identity address-list-error {
    base vrrp-error-virtual-router;
    description
        "Получен пакет со списком адресов, не соответствующим
        заданному локально списку для виртуального маршрутизатора.";
}
identity interval-error {
    base vrrp-error-virtual-router;
    description
        "Получен пакет с интервалом анонсирования, отличным от
        заданного для локального виртуального маршрутизатора.";
}
identity packet-length-error {
    base vrrp-error-virtual-router;
    description
        "Получен пакет с размером меньше размера заголовка VRRP.";
}

/* vrrp-state-type и производные от него. */
identity vrrp-state-type {
    description
        "Состояние виртуального маршрутизатора.";
}
identity initialize {
    base vrrp-state-type;
    description
        "Виртуальный маршрутизатор ждёт стартового события.";
}
identity backup {
    base vrrp-state-type;
    description
        "Виртуальный маршрутизатор отслеживает доступность ведущего.";
}
identity master {
    base vrrp-state-type;
    description
```

```
"Указывает, что виртуальный маршрутизатор пересылает пакеты для
адресов IP, связанных с этим виртуальным маршрутизатором.";
}

/* vrrp-version и производные от него. */
identity vrrp-version {
  description
    "Версия VRRP.";
}
identity vrrp-v2 {
  base vrrp-version;
  description
    "VRRP версии 2.";
}
identity vrrp-v3 {
  base vrrp-version;
  description
    "VRRP версии 3.";
}

/*
 * Группировки
 */

grouping vrrp-common-attributes {
  description
    "Группа общих атрибутов VRRP для версий 2 и 3.";

  leaf vrid {
    type uint8 {
      range "1..255";
    }
    description
      "Идентификатор виртуального маршрутизатора (VRID).";
  }

  leaf version {
    type identityref {
      base vrrp:vrrp-version;
    }
    mandatory true;
    description
      "VRRP версии 2 или 3.";
  }

  leaf log-state-change {
    type boolean;
    default "false";
    description
      "Задаёт генерацию сообщения о смене состояния VRRP при каждом
      изменении статуса экземпляра VRRP (up в down или обратно).";
  }

  container preempt {
    description
      "Позволяет резервному маршрутизатору VRRP с более высоким
      приоритетом вытеснить VRRP master с меньшим приоритетом.";
    leaf enabled {
      type boolean;
      default "true";
      description
        "Значение true разрешает вытеснение.";
    }
    leaf hold-time {
      type uint16;
      units seconds;
      default 0;
      description
        "Время (в секундах), в течение которого backup-маршрутизатор
        VRRP с более высоким приоритетом должен ждать перед
        вытеснением VRRP master.";
    }
  }

  leaf priority {
    type uint8 {
      range "1..254";
    }
    default 100;
    description
      "Приоритет выбора VRRP для резервного виртуального
      маршрутизатора.";
  }

  leaf accept-mode {
    when "derived-from-or-self(current()/../version, 'vrrp-v3')" {

```

```

description
  "Применимо лишь для версии 3.";
}
type boolean;
default "false";
description
  "Задаёт, будет ли виртуальный маршрутизатор master
  воспринимать пакеты, направленные владельцу адреса IPvX, как
  свои, если адрес IPvX не принадлежит ему. По умолчанию
  задано false. Развёртывания, полагающиеся, например, на
  ping по адресу владельца IPvX могут захотеть установки
  accept-mode true.

  Примечание. Сообщения IPv6 Neighbor Solicitations и Neighbor
  Advertisements НЕДОПУСТИМО отбрасывать при accept-mode
  false.";
}
} // vrrp-common-attributes

grouping vrrp-ipv4-attributes {
  description
    "Группа атрибутов VRRP для IPv4.";

  uses vrrp-common-attributes;

  choice advertise-interval-choice {
    description
      "Опции для интервала, с которым анонсы VRRPv2 или VRRPv3
      передаются с указанного интерфейса.";

    case v2 {
      when "derived-from-or-self(version, 'vrrp-v2')" {
        description
          "Применимо лишь для версии 2.";
      }
      leaf advertise-interval-sec {
        type uint8 {
          range "1..254";
        }
        units seconds;
        default 1;
        description
          "Интервал, с которым анонсы VRRPv2 передаются с
          указанного интерфейса.";
      }
    }

    case v3 {
      when "derived-from-or-self(version, 'vrrp-v3')" {
        description
          "Применимо лишь для версии 3.";
      }
      leaf advertise-interval-centi-sec {
        type uint16 {
          range "1..4095";
        }
        units centiseconds;
        default 100;
        description
          "Интервал, с которым анонсы VRRPv3 передаются с
          указанного интерфейса.";
      }
    }
  }
} // advertise-interval-choice

container track {
  description
    "Разрешает указанному экземпляру VRRP отслеживать
    интерфейсы или сети.";
  container interfaces {
    description
      "Разрешает заданному экземпляру VRRPv2 или VRRPv3
      отслеживать интерфейсы. Это предотвращает потери трафика
      за счёт обнаружения доступности интерфейсов. Рабочие
      состояния других интерфейсов связаны с приоритетом
      маршрутизатора VRRP. Когда отслеживаемый интерфейс
      становится недоступным (или operational down), приоритет
      маршрутизатора VRRP декрементируется. Когда доступность
      интерфейса восстанавливается, приоритет маршрутизатора
      VRRP инкрементируется на то же значение.";

    list interface {
      key "interface";
      description
        "Интерфейс для отслеживания.";
      leaf interface {

```

```
type if:interface-ref;
must "/if:interfaces/if:interface[if:name=current()]/"
  + "ip:ipv4" {
  description
    "Интерфейс IPv4.";
  }
description
  "Интерфейс для отслеживания.";
}

leaf priority-decrement {
  type uint8 {
    range "1..254";
  }
  default 10;
  description
    "Значение декремента приоритета экземпляра VRRP при
    отключении (down) интерфейса.";
}
} // interface
} // interfaces

container networks {
  description
    "Позволяет экземпляру маршрутизатора VRRPv2 или VRRPv3
    отслеживать сети по заданному префиксу IPv4. Отслеживание
    сетей предотвращает потери трафика за счёт обнаружения
    отказов связности. Состояния связности с некоторыми сетями
    привязываются к приоритету маршрутизатора VRRP. Когда
    связность с сетью, представленной префиксом, теряется,
    приоритет маршрутизатора VRRP декрементируется. При
    восстановлении связности приоритет инкрементируется на
    то же значение.";
  list network {
    key "prefix";
    description
      "Позволяет экземпляру маршрутизатора VRRPv2 или VRRPv3
      отслеживать сеть IPv4 путём задания её префикса IPv4.";

    leaf prefix {
      type inet:ipv4-prefix;
      description
        "Префикс IPv4 для отслеживаемой сети.";
    }

    leaf priority-decrement {
      type uint8 {
        range "1..254";
      }
      default 10;
      description
        "Значение декремента приоритета экземпляра VRRP при
        отказе сети IPv4.";
    }
  } // network
} // networks
} // track

container virtual-ipv4-addresses {
  description
    "Задаёт виртуальный адрес IPv4 для интерфейса VRRP.";

  list virtual-ipv4-address {
    key "ipv4-address";
    max-elements 16;
    description
      "Виртуальный адрес IPv4 для одного экземпляра VRRP. Для
      владеющего VRRP маршрутизатора виртуальный адрес должен
      совпадать с одним из адресов IPv4 на интерфейсе,
      соответствующем виртуальному маршрутизатору.";

    leaf ipv4-address {
      type inet:ipv4-address;
      description
        "Адрес IPv4, связанный с виртуальным маршрутизатором.";
      reference
        "RFC 5798: Virtual Router Redundancy Protocol (VRRP)
        Version 3 for IPv4 and IPv6. Section 1.2";
    }
  } // virtual-ipv4-address
} // virtual-ipv4-addresses
} // vrrp-ipv4-attributes

grouping vrrp-ipv6-attributes {
  description
    "Группа атрибутов VRRP для IPv6.";
```

```

uses vrrp-common-attributes;

leaf advertise-interval-centi-sec {
  type uint16 {
    range "1..4095";
  }
  units centiseconds;
  default 100;
  description
    "Интервал, с которым анонсы VRRPv3 передаются с
    указанного интерфейса.";
}

container track {
  description
    "Разрешает указанному экземпляру VRRP отслеживать
    интерфейсы или сети.";
  container interfaces {
    description
      "Разрешает заданному экземпляру VRRPv2 или VRRPv3
      отслеживать интерфейсы. Это предотвращает потери трафика
      за счёт обнаружения доступности интерфейсов. Рабочие
      состояния других интерфейсов связаны с приоритетом
      маршрутизатора VRRP. Когда отслеживаемый интерфейс
      становится недоступным (или operational down), приоритет
      маршрутизатора VRRP декрементируется. Когда доступность
      интерфейса восстанавливается, приоритет маршрутизатора
      VRRP инкрементируется на то же значение.";
    list interface {
      key "interface";
      description
        "Интерфейс для отслеживания.";

      leaf interface {
        type if:interface-ref;
        must "/if:interfaces/if:interface[if:name=current()]/"
          + "ip:ipv6" {
          description
            "Интерфейс IPv6.";
        }
        description
          "Интерфейс для отслеживания.";
      }

      leaf priority-decrement {
        type uint8 {
          range "1..254";
        }
        default 10;
        description
          "Значение декремента приоритета экземпляра VRRP при
          отключении (down) интерфейса.";
      }
    } // interface
  } // interfaces

  container networks {
    description
      "Позволяет экземпляру маршрутизатора VRRPv2 или VRRPv3
      отслеживать сети по заданному префиксу IPv6. Отслеживание
      сетей предотвращает потери трафика за счёт обнаружения
      отказов связности. Состояния связности с некоторыми сетями
      привязываются к приоритету маршрутизатора VRRP. Когда
      связность с сетью, представленной префиксом, теряется,
      приоритет маршрутизатора VRRP декрементируется. При
      восстановлении связности приоритет инкрементируется на
      то же значение.";
    list network {
      key "prefix";
      description
        "Позволяет экземпляру маршрутизатора VRRPv2 или VRRPv3
        отслеживать сеть IPv6 путём задания её префикса IPv6.";

      leaf prefix {
        type inet:ipv6-prefix;
        description
          "Префикс IPv6 для отслеживаемой сети.";
      }

      leaf priority-decrement {
        type uint8 {
          range "1..254";
        }
        default 10;
        description

```

```

"Значение декремента приоритета экземпляра VRRP при
отказе сети IPv6.";
    }
  } // network
} // networks
} // track

container virtual-ipv6-addresses {
  description
    "Задаёт виртуальный адрес IPv6 для интерфейса VRRP.";
  list virtual-ipv6-address {
    key "ipv6-address";
    max-elements 2;
    description
      "Разрешено два адреса IPv6, первый из которых должен быть
      link-local, а второй может быть глобальным или
      link-local.";

    leaf ipv6-address {
      type inet:ipv6-address;
      description
        "Адрес IPv6, связанный с виртуальным маршрутизатором.";
      reference
        "RFC 5798: Virtual Router Redundancy Protocol (VRRP)
        Version 3 for IPv4 and IPv6. Section 1.3";
    }
  } // virtual-ipv6-address
} // virtual-ipv6-addresses
} // vrrp-ipv6-attributes

grouping vrrp-state-attributes {
  description
    "Группа атрибутов состояния VRRP.";

  leaf state {
    type identityref {
      base vrrp:vrrp-state-type;
    }
    config false;
    description
      "Рабочее состояние.";
  }

  leaf is-owner {
    type boolean;
    config false;
    description
      "Значение true, если этот виртуальный маршрутизатор
      является владельцем.";
  }

  leaf last-adv-source {
    type inet:ip-address;
    config false;
    description
      "Анонсированный последним адрес источника IPv4/IPv6.";
  }

  leaf up-datetime {
    type yang:date-and-time;
    config false;
    description
      "Дата и время выхода виртуального маршрутизатора
      из состояния init.";
  }

  leaf master-down-interval {
    type uint32;
    units centiseconds;
    config false;
    description
      "Интервал для резервного маршрутизатора при
      декларировании master down.";
  }

  leaf skew-time {
    type uint32;
    units microseconds;
    config false;
    description
      "Рассчитывается на основе параметров приоритета и интервала
      анонсирования, заданных командой. См. RFC 3768.";
  }

  leaf last-event {
    type identityref {

```

```
    base vrrp:vrrp-event-type;
  }
  config false;
  description
    "Сообщенное последним событие.";
}

leaf new-master-reason {
  type new-master-reason-type;
  config false;
  description
    "Причина перехода виртуального маршрутизатора
     в состояние master.";
}

container statistics {
  config false;
  description
    "Статистика VRRP.";

  leaf discontinuity-datetime {
    type yang:date-and-time;
    description
      "Время последнего разрыва одного или нескольких счетчиков
       статистики VRRP. Если разрывов не было с момента
       реинициализации локальной подсистемы управления, этот узел
       содержит время реинициализации этой подсистемы.";
  }

  leaf master-transitions {
    type yang:counter32;
    description
      "Общее число переходов этого виртуального маршрутизатора
       в состояние master.";
  }

  leaf advertisement-rcvd {
    type yang:counter64;
    description
      "Общее число анонсов VRRP, полученных этим виртуальным
       маршрутизатором.";
  }

  leaf advertisement-sent {
    type yang:counter64;
    description
      "Общее число анонсов VRRP, переданных этим виртуальным
       маршрутизатором.";
  }

  leaf interval-errors {
    if-feature validate-interval-errors;
    type yang:counter64;
    description
      "Общее число пакетов с анонсами VRRP, полученных с
       интервалом, отличным от заданного для локального
       виртуального маршрутизатора.";
  }

  leaf priority-zero-pkts-rcvd {
    type yang:counter64;
    description
      "Общее число пакетов VRRP, полученных виртуальным
       маршрутизатором с приоритетом 0.";
  }

  leaf priority-zero-pkts-sent {
    type yang:counter64;
    description
      "Общее число пакетов VRRP, переданных виртуальным
       маршрутизатором с приоритетом 0.";
  }

  leaf invalid-type-pkts-rcvd {
    type yang:counter64;
    description
      "Общее число пакетов VRRP, полученных виртуальным
       маршрутизатором с непригодным значением поля type.";
  }

  leaf address-list-errors {
    if-feature validate-address-list-errors;
    type yang:counter64;
    description
      "Общее число пакетов VRRP, полученных со списком адресов,
       не совпадающим с локально настроенным списком адресов";
  }
}
```

```
        для виртуального маршрутизатора.";
    }

    leaf packet-length-errors {
        type yang:counter64;
        description
            "Общее число пакетов VRRP, полученных с размером меньше
            размера заголовка VRRP.";
    }
} // statistics
} // vrrp-state-attributes

grouping vrrp-global-state-attributes {
    description
        "Группа атрибутов глобального состояния VRRP.";

    leaf virtual-routers {
        type uint32;
        description
            "Число настроенных виртуальных маршрутизаторов.";
    }

    leaf interfaces {
        type uint32;
        description
            "Число интерфейсов с настроенным VRRP.";
    }

    container statistics {
        description
            "Глобальная статистика VRRP.";

        leaf discontinuity-datetime {
            type yang:date-and-time;
            description
                "Время последнего разрыва одного или нескольких значений
                checksum-errors, version-errors, vrid-errors,
                ip-ttl-errors. Если разрывов не было с момента
                реинициализации локальной подсистемы управления, этот узел
                содержит время реинициализации этой подсистемы."
        }

        leaf checksum-errors {
            type yang:counter64;
            description
                "Общее число пакетов VRRP, полученных с ошибкой
                контрольной суммы VRRP.";
            reference
                "RFC 5798: Virtual Router Redundancy Protocol (VRRP)
                Version 3 for IPv4 and IPv6. Section 5.2.8";
        }

        leaf version-errors {
            type yang:counter64;
            description
                "Общее число пакетов VRRP, полученных с неизвестным
                или неподдерживаемым номером версии.";
            reference
                "RFC 5798: Virtual Router Redundancy Protocol (VRRP)
                Version 3 for IPv4 and IPv6. Section 5.2.1";
        }

        leaf vrid-errors {
            type yang:counter64;
            description
                "Общее число пакетов VRRP, полученных с VRID, не
                подходящим ни одному виртуальному маршрутизатору
                на этом маршрутизаторе.";
            reference
                "RFC 5798: Virtual Router Redundancy Protocol (VRRP)
                Version 3 for IPv4 and IPv6. Section 5.2.3";
        }

        leaf ip-ttl-errors {
            type yang:counter64;
            description
                "Общее число пакетов VRRP, полученных виртуальным
                маршрутизатором с IP TTL (IPv4) или Hop Limit (IPv6),
                отличным от 255.";
            reference
                "RFC 5798: Virtual Router Redundancy Protocol (VRRP)
                Version 3 for IPv4 and IPv6.
                Sections 5.1.1.3 and 5.1.2.3";
        }
    } // statistics
} // vrrp-global-state-attributes
```



```

/*
 * Узлы данных конфигурации и рабочего состояния
 */

augment "/if:interfaces/if:interface/ip:ipv4" {
  description
    "Дополняет интерфейс IPv4.";

  container vrrp {
    description
      "Настраивает VRRP версии 2 или 3 для IPv4.";

    list vrrp-instance {
      key "vrid";
      description
        "Задаёт виртуальный маршрутизатор, идентифицируемый
        VRID, в адресном пространстве IPv4.";

      uses vrrp-ipv4-attributes;
      uses vrrp-state-attributes;
    }
  }
} // augments ipv4

augment "/if:interfaces/if:interface/ip:ipv6" {
  description
    "Дополняет интерфейс IPv6.";

  container vrrp {
    description
      "Настраивает VRRP версии 3 для IPv6.";

    list vrrp-instance {
      must "derived-from-or-self(version, 'vrrp-v3')" {
        description
          "IPv6 поддерживается только версией 3.";
      }
      key "vrid";
      description
        "Задаёт виртуальный маршрутизатор, идентифицируемый
        VRID, в адресном пространстве IPv6.";

      uses vrrp-ipv6-attributes;
      uses vrrp-state-attributes;
    }
  }
} // augments ipv6

container vrrp {
  config false;
  description
    "Данные VRRP на глобальном уровне.";

  uses vrrp-global-state-attributes;
}

/*
 * Уведомления
 */

notification vrrp-new-master-event {
  description
    "Уведомление для выбора нового VRRP master.";
  leaf master-ip-address {
    type inet:ip-address;
    mandatory true;
    description
      "Адрес IPv4 IPv6 нового ведущего (master).";
  }
  leaf new-master-reason {
    type new-master-reason-type;
    mandatory true;
    description
      "Причина перехода в состояние master.";
  }
}

notification vrrp-protocol-error-event {
  description
    "Уведомление для ошибки протокола VRRP.";
  leaf protocol-error-reason {
    type identityref {
      base vrrp:vrrp-error-global;
    }
    mandatory true;
  }
}

```

```

description
  "Причина протокольной ошибки.";
}
}

notification vrrp-virtual-router-error-event {
description
  "Уведомление для ошибки в виртуальном маршрутизаторе.";
leaf interface {
  type if:interface-ref;
  mandatory true;
  description
    "Интерфейс, на котором произошло событие.";
}

choice ip-version {
  mandatory true;
  description
    "Ошибка может возникнуть на виртуальном маршрутизаторе IPv4
или IPv6. Сведения о версии IP предоставляют указанные ниже
варианты.";
  case ipv4 {
    description
      "IPv4.";
    container ipv4 {
      description
        "Сведения об ошибке для IPv4.";
      leaf vrid {
        type leafref {
          path "/if:interfaces/if:interface"
            + "[if:name = current()../../../../vrrp:interface]/"
            + "ip:ipv4/vrrp:vrrp/vrrp:vrrp-instance/vrrp:vrid";
        }
        mandatory true;
        description
          "Виртуальный маршрутизатор, где произошло событие.";
      }
    }
  }
  case ipv6 {
    description
      "IPv6.";
    container ipv6 {
      description
        "Сведения об ошибке для IPv6.";
      leaf vrid {
        type leafref {
          path "/if:interface"
            + "[if:name = current()../../../../vrrp:interface]/"
            + "ip:ipv6/vrrp:vrrp/vrrp:vrrp-instance/vrrp:vrid";
        }
        mandatory true;
        description
          "Виртуальный маршрутизатор, где произошло событие.";
      }
    }
  }
}
}

leaf virtual-router-error-reason {
  type identityref {
    base vrrp:vrrp-error-virtual-router;
  }
  mandatory true;
  description
    "Причина ошибки виртуального маршрутизатора.";
}
}
}
<CODE ENDS>

```

5. Взаимодействие с IANA

Этот документ регистрирует URI пространства имён в реестре IETF XML Registry [RFC3688].

URI: urn:ietf:params:xml:ns:yang:ietf-vrrp

Registrant Contact: The IESG.

XML: N/A; регистрируемый URI является пространством имён XML.

Документ регистрирует модуль YANG в реестре YANG Module Names [RFC7950].

```

name:      ietf-vrrp
namespace: urn:ietf:params:xml:ns:yang:ietf-vrrp
prefix:    vrrp
reference: RFC 8347

```

6. Вопросы безопасности

Заданные в этом документе модули YANG определяют схемы для данных, которые разработаны для доступа через протоколы управления сетью, такие как NETCONF [RFC6241] и RESTCONF [RFC8040]. Нижним уровнем NETCONF является защищённый транспорт с обязательной реализацией Secure Shell (SSH) [RFC6242]. Нижним уровнем RESTCONF служит HTTPS с обязательной реализацией защищённого транспорта TLS [RFC5246].

Модель управления доступом NETCONF [RFC8341] обеспечивает средства, позволяющие предоставить доступ лишь конкретным пользователям NETCONF и RESTCONF к предопределённому подмножеству доступных в NETCONF или RESTCONF протокольных операций и содержимого.

В этом модуле YANG имеется множество узлов данных, доступных для записи, создания, удаления (т. е. с принятым по умолчанию `config true`). Эти узлы могут быть чувствительными или уязвимыми в некоторых сетевых средах. Операции записи (например, `edit-config`) в такие узлы без подходящей защиты могут оказывать негативное влияние на работу сети. Ниже указаны ветви и узлы данных и их уязвимости.

/if:interfaces/if:interface/ip:ipv4/vrrp:vrrp/vrrp:vrrp-instance

/if:interfaces/if:interface/ip:ipv6/vrrp:vrrp/vrrp:vrrp-instance

Несанкционированный доступ к любому из узлов данных в этих ветвях может негативно влиять на подсистему маршрутизации в локальном устройстве и сети. Это может приводить к некорректной работе сети, доставке пакетов не по назначению и другим проблемам.

Некоторые из доступных для чтения узлов этого модуля YANG могут быть конфиденциальными или уязвимыми в некоторых сетевых средах. Важно контролировать доступ к считыванию таких узлов данных (например, через операции `get`, `get-config`, `notification`). Ниже указаны ветви и узлы данных конфиденциальные или уязвимые в плане их чтения.

/ietf-vrrp:vrrp

/if:interfaces/if:interface/ip:ipv4/vrrp:vrrp/vrrp:vrrp-instance

/if:interfaces/if:interface/ip:ipv6/vrrp:vrrp/vrrp:vrrp-instance

Несанкционированный доступ к любому из узлов данных в этих ветвях может раскрывать данные рабочего состояния VRRP на этом устройстве.

7. Литература

7.1. Нормативные документы

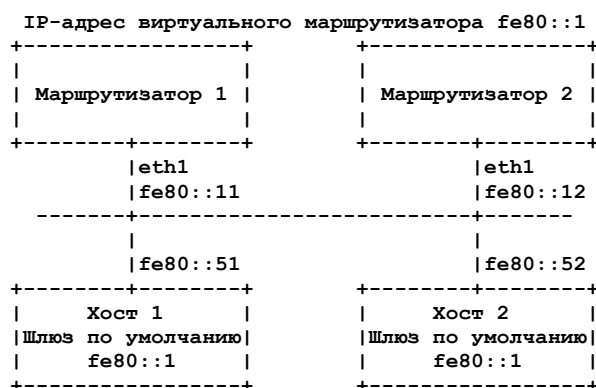
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, [RFC 2119](#), DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC2787] Jewell, B. and D. Chuang, "Definitions of Managed Objects for the Virtual Router Redundancy Protocol", RFC 2787, DOI 10.17487/RFC2787, March 2000, <<https://www.rfc-editor.org/info/rfc2787>>.
- [RFC3688] Mealling, M., "The IETF XML Registry", BCP 81, [RFC 3688](#), DOI 10.17487/RFC3688, January 2004, <<https://www.rfc-editor.org/info/rfc3688>>.
- [RFC5246] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2", [RFC 5246](#), DOI 10.17487/RFC5246, August 2008, <<https://www.rfc-editor.org/info/rfc5246>>.
- [RFC5798] Nadas, S., Ed., "Virtual Router Redundancy Protocol (VRRP) Version 3 for IPv4 and IPv6", [RFC 5798](#), DOI 10.17487/RFC5798, March 2010, <<https://www.rfc-editor.org/info/rfc5798>>.
- [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>>.
- [RFC6242] Wasserman, M., "Using the NETCONF Protocol over Secure Shell (SSH)", [RFC 6242](#), DOI 10.17487/RFC6242, June 2011, <<https://www.rfc-editor.org/info/rfc6242>>.
- [RFC6527] Tata, K., "Definitions of Managed Objects for Virtual Router Redundancy Protocol Version 3 (VRRPv3)", RFC 6527, DOI 10.17487/RFC6527, March 2012, <<https://www.rfc-editor.org/info/rfc6527>>.
- [RFC6991] Schoenwaelder, J., Ed., "Common YANG Data Types", [RFC 6991](#), DOI 10.17487/RFC6991, July 2013, <<https://www.rfc-editor.org/info/rfc6991>>.
- [RFC7950] Bjorklund, M., Ed., "The YANG 1.1 Data Modeling Language", [RFC 7950](#), DOI 10.17487/RFC7950, August 2016, <<https://www.rfc-editor.org/info/rfc7950>>.
- [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>>.
- [RFC8341] Bierman, A. and M. Bjorklund, "Network Configuration Access Control Model", STD 91, [RFC 8341](#), DOI 10.17487/RFC8341, March 2018, <<https://www.rfc-editor.org/info/rfc8341>>.
- [RFC8342] Bjorklund, M., Schoenwaelder, J., Shafer, P., Watsen, K., and R. Wilton, "Network Management Datastore Architecture (NMDA)", [RFC 8342](#), DOI 10.17487/RFC8342, March 2018, <<https://www.rfc-editor.org/info/rfc8342>>.
- [RFC8343] Bjorklund, M., "A YANG Data Model for Interface Management", [RFC 8343](#), DOI 10.17487/RFC8343, March 2018, <<https://www.rfc-editor.org/info/rfc8343>>.
- [RFC8344] Bjorklund, M., "A YANG Data Model for IP Management", [RFC 8344](#), DOI 10.17487/RFC8344, March 2018, <<https://www.rfc-editor.org/info/rfc8344>>.

7.2. Дополнительная литература

- [RFC3768] Hinden, R., Ed., "Virtual Router Redundancy Protocol (VRRP)", [RFC 3768](#), DOI 10.17487/RFC3768, April 2004, <<https://www.rfc-editor.org/info/rfc3768>>.
- [RFC7224] Bjorklund, M., "IANA Interface Type YANG Module", [RFC 7224](#), DOI 10.17487/RFC7224, May 2014, <<https://www.rfc-editor.org/info/rfc7224>>.
- [RFC7951] Lhotka, L., "JSON Encoding of Data Modeled with YANG", [RFC 7951](#), DOI 10.17487/RFC7951, August 2016, <<https://www.rfc-editor.org/info/rfc7951>>.
- [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>>.
- [Subscribed-Notifications] Voit, E., Clemm, A., Gonzalez Prieto, A., Nilsen-Nygaard, E., and A. Tripathy, "Custom Subscription to Event Streams", Work in Progress, draft-ietf-netconf-subscribed-notifications-10¹, February 2018.
- [YANG-Push] Clemm, A., Voit, E., Gonzalez Prieto, A., Tripathy, A., Nilsen-Nygaard, E., Bierman, A., and B. Lengyel, "YANG Datastore Subscription", Work in Progress, draft-ietf-netconf-yang-push-15², February 2018.
- [YANG-Guidelines] Bierman, A., "Guidelines for Authors and Reviewers of YANG Data Model Documents", Work in Progress, draft-ietf-netmod-rfc6087bis-20³, March 2018.

Приложение А. Пример дерева данных

В этом приложении представлен пример дерева данных экземпляра в кодировке JSON [RFC7951] для конфигурации и состояния (пример включает iana-if-type из [RFC7224]).



Данные экземпляра конфигурации Маршрутизатора 1 с приведённого выше рисунка показаны ниже.

```

{
  "ietf-interfaces:interfaces": {
    "interface": [
      {
        "name": "eth1",
        "description": "An interface with VRRP enabled.",
        "type": "iana-if-type:ethernetCsmacd",
        "ietf-ip:ipv6": {
          "address": [
            {
              "ip": "2001:db8:0:1::1",
              "prefix-length": 64
            },
            {
              "ip": "fe80::11",
              "prefix-length": 64
            }
          ]
        },
        "forwarding": true,
        "ietf-vrrp:vrrp": {
          "vrrp-instance": [
            {
              "vrid": 1,
              "version": "vrrp-v3",
              "priority": 200,
              "advertise-interval-centi-sec": 50,
              "virtual-ipv6-addresses": {
                "virtual-ipv6-address": [
                  {
                    "ipv6-address": "fe80::1"
                  }
                ]
              }
            }
          ]
        }
      }
    ]
  }
}

```

¹Опубликовано в [RFC 8639](#). Прим. перев.

²Опубликовано в [RFC 8641](#). Прим. перев.

³Опубликовано в [RFC 8407](#). Прим. перев.

```
]
}
```

Соответствующие данные рабочего состояния Маршрутизатора 1 показаны ниже.

```
{
  "ietf-interfaces:interfaces": {
    "interface": [
      {
        "name": "eth1",
        "description": "An interface with VRRP enabled.",
        "type": "iana-if-type:ethernetCsmacd",
        "phys-address": "00:00:5e:00:53:01",
        "oper-status": "up",
        "statistics": {
          "discontinuity-time": "2016-10-24T17:11:27+02:00"
        },
      },
      "ietf-ip:ipv6": {
        "forwarding": true,
        "mtu": 1500,
        "address": [
          {
            "ip": "2001:db8:0:1::1",
            "prefix-length": 64,
            "origin": "static",
            "status": "preferred"
          },
          {
            "ip": "fe80::11",
            "prefix-length": 64,
            "origin": "static",
            "status": "preferred"
          }
        ]
      }
    ]

    "ietf-vrrp:vrrp": {
      "vrrp-instance": [
        {
          "vrid": 1,
          "version": "vrrp-v3",
          "log-state-change": false,
          "preempt": {
            "enabled": true,
            "hold-time": 0
          }
          "priority": 200,
          "accept-mode": false,
          "advertise-interval-centi-sec": 50,
          "virtual-ipv6-addresses": {
            "virtual-ipv6-address": [
              "ipv6-address": "fe80::1"
            ]
          },
          "state": "master",
          "is-owner": false,
          "last-adv-source": "fe80::11",
          "up-datetime": "2016-10-24T17:11:27+02:00",
          "master-down-interval": 161,
          "skew-time": 11,
          "last-event": "vrrp-event-interface-up",
          "new-master-reason": "priority",
          "statistics": {
            "discontinuity-datETIME":
              "2016-10-24T17:11:27+02:00",
            "master-transitions": 2,
            "advertisement-rcvd": 20,
            "advertisement-sent": 12,
            "interval-errors": 0,
            "priority-zero-pkts-rcvd": 0,
            "priority-zero-pkts-sent": 0,
            "invalid-type-pkts-rcvd": 0,
            "address-list-errors": 0,
            "packet-length-errors": 1
          }
        }
      ]
    }
  }
}

{
  "ietf-vrrp:vrrp": {
    "virtual-routers": 3,
    "interfaces": 2,
```

```
"statistics": {  
  "discontinuity-datetime": "2016-10-24T17:11:27+02:00",  
  "checksum-errors": 2,  
  "version-errors": 0,  
  "vrid-errors": 0,  
  "ip-ttl-errors": 1  
}
```

Адреса авторов

Xufeng Liu (editor)

Jabil
8281 Greensboro Drive, Suite 200
McLean, VA 22102
United States of America
Email: xufeng.liu.ietf@gmail.com

Athanasios Kyparlis

Jabil
8281 Greensboro Drive, Suite 200
McLean, VA 22102
United States of America
Email: Athanasios_Kyparlis@jabil.com

Ravi Parikh

VMware
3425 Hillview Avenue

Palo Alto, CA 94304
United States of America
Email: parikhr@vmware.com

Acee Lindem

Cisco Systems
301 Midenhall Way
Cary, NC 27513
United States of America
Email: acee@cisco.com

Mingui Zhang

Huawei Technologies
No. 156 Beiqing Rd. Haidian District
Beijing 100095
China
Email: zhangmingui@huawei.com

Перевод на русский язык

Николай Малых

nmalykh@protokols.ru