

Internet Engineering Task Force (IETF)  
Request for Comments: 8335  
Updates: 4884  
Category: Standards Track  
ISSN: 2070-1721

R. Bonica  
R. Thomas  
Juniper Networks  
J. Linkova  
Google  
C. Lenart  
Verizon  
M. Boucadair  
Orange  
February 2018

## PROBE - утилита для зондирования интерфейсов

### PROBE: A Utility for Probing Interfaces

#### Тезисы

Этот документ описывает инструмент сетевой диагностики, названный PROBE. Программа PROBE похожа на PING в том смысле, что может использоваться для запроса состояния проверяемого интерфейса, но отличается тем, что ей не требуется двухсторонняя связность между проверяющим и проверяемым интерфейсом. Вместо этого программе PROBE нужна двухсторонняя связность между проверяющим интерфейсом и интерфейсом-посредником (проху). Этот интерфейс может размещаться на одном узле с проверяемым интерфейсом или на узле, с которым тот непосредственно соединен. Этот документ обновляет RFC 4884.

#### Статус документа

Документ относится к категории Internet Standards Track.

Документ является результатом работы IETF<sup>1</sup> и представляет согласованный взгляд сообщества IETF. Документ прошел открытое обсуждение и был одобрен для публикации IESG<sup>2</sup>. 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/rfc8335>.

#### Авторские права

Авторские права (Copyright (c) 2018) 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.е документа Trust Legal Provisions, без каких-либо гарантий (как указано в Simplified BSD License).

## Оглавление

1. Введение.....	2
1.1. Терминология.....	2
1.2. Уровни требований.....	2
2. Запрос ICMP Extended Echo.....	2
2.1. Объект идентификации интерфейса.....	3
3. Отклик ICMP Extended Echo.....	4
4. Обработка сообщений ICMP.....	4
4.1. Обработка поля Code.....	5
5. Примеры использования.....	5
6. Обновление RFC 4884.....	6
7. Взаимодействие с IANA.....	6
8. Вопросы безопасности.....	6
9. Литература.....	7
9.1. Нормативные документы.....	7
9.2. Дополнительная литература.....	7
Приложение А. Программа PROBE.....	7
Благодарности.....	8
Адреса авторов.....	8

<sup>1</sup>Internet Engineering Task Force.

<sup>2</sup>Internet Engineering Steering Group.

## 1. Введение

Сетевые операторы используют PING [RFC2151] для проверки двухсторонней связности между интерфейсами. В этом документе такие интерфейсы называются проверяющим и проверяемым. PING передает сообщения ICMP [RFC792] [RFC4443] Echo Request с проверяющего интерфейса проверяемому. Проверяющий интерфейс размещается на проверяющем узле, а проверяемый - на проверяемом.

Если проверяемый интерфейс получает сообщение ICMP Echo Request, он возвращает ICMP Echo Reply. Получив ICMP Echo Reply, проверяющий узел убеждается в наличии двухсторонней связности между проверяющим и проверяемым интерфейсом. В частности, подтверждается:

- возможность доступа проверяющего узла к проверяемому интерфейсу;
- активность проверяемого интерфейса;
- возможность доступа проверяемого узла к проверяющему интерфейсу;
- активность проверяющего интерфейса.

Этот документ описывает инструмент сетевой диагностики, названный PROBE. Программа PROBE похожа на PING в том смысле, что может использоваться для запроса состояния проверяемого интерфейса, но отличается тем, что ей не требуется двухсторонняя связность между проверяющим и проверяемым интерфейсом. Вместо этого программе PROBE нужна двухсторонняя связность между проверяющим интерфейсом и интерфейсом-посредником (проху). Этот интерфейс может размещаться на одном узле с проверяемым интерфейсом или на узле, с которым тот непосредственно соединен. В разделе 5 рассмотрены примеры использования программы.

Подобно PING, программа PROBE выполняется на проверяющем узле, который передает сообщение ICMP Extended Echo Request со своего локального интерфейса, называемого проверяющим, на интерфейс посредника, размещаемый на узле-посреднике.

ICMP Extended Echo Request содержит структуру ICMP Extension Structure, а та содержит объект Interface Identification Object, указывающий проверяемый интерфейс, который находится на узле-посреднике или соединен с ним.

Когда интерфейс-посредник получает ICMP Extended Echo Request, узел-посредник выполняет процедуры контроля доступа. Если доступ разрешен, этот узел определяет статус проверяемого интерфейса и возвращает ICMP Extended Echo Reply. Сообщение ICMP Extended Echo Reply указывает статус проверяемого интерфейса.

Если проверяемый интерфейс размещен на узле-посреднике, PROBE определяет его статус как oper-status [RFC7223]. Если oper-status имеет значение up (1), PROBE сообщает об активности интерфейса. В противном случае PROBE сообщает, что интерфейс не активен.

Если проверяемый интерфейс расположен на узле, напрямую соединенном с посредником, и присутствует в таблице IPv4 ARP<sup>1</sup> [RFC826] или IPv6 Neighbor Cache [RFC4861], PROBE сообщает о доступности интерфейса. В противном случае PROBE сообщает, что в таблице нет записи для интерфейса.

### 1.1. Терминология

Ниже приведены определения используемых в документе терминов.

- Probing interface (проверяющий интерфейс) - интерфейс, передающий ICMP Extended Echo Request.
- Probing node (проверяющий узел) - узел, на котором размещается проверяющий интерфейс.
- Proxy interface (интерфейс-посредник) - интерфейс, которому передается ICMP Extended Echo Request.
- Proxy node (узел-посредник) - узел, на котором размещается проху-интерфейс.
- Probed interface (проверяемый интерфейс) - интерфейс, чей статус запрашивается.
- Probed node (проверяемый узел) - узел, на котором размещается проверяемый интерфейс. Если проверяемый и посредничающий интерфейсы расположены на одном узле, узел посредник является и проверяемым узлом. В остальных случаях узел-посредник непосредственно соединен с проверяемым узлом.

### 1.2. Уровни требований

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

## 2. Запрос ICMP Extended Echo

Сообщение ICMP Extended Echo Request определено для ICMPv4 и ICMPv6. Подобно другим сообщениям ICMP, оно инкапсулируется с использованием заголовков IP (версия ICMPv4 инкапсулируется в IPv4, версия ICMPv6 в IPv6).

На рисунке 1 показано сообщение ICMP Extended Echo Request.

<sup>1</sup>Address Resolution Protocol - протокол преобразования адресов.

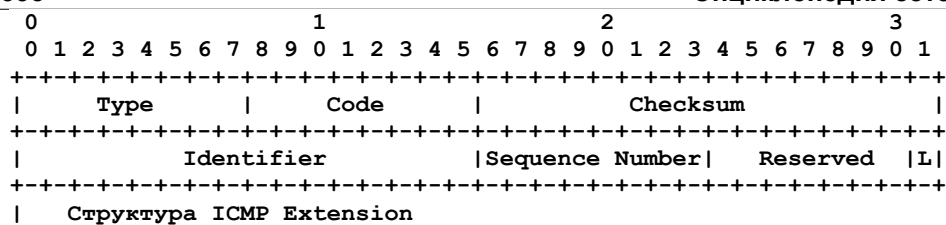


Рисунок 1. ICMP Extended Echo Request.

Поля заголовка IP

- Source Address - адрес отправителя для проверяющего интерфейса. Поле **должно** содержать действительный индивидуальный адрес IPv4 или IPv6.
- Destination Address - адрес получателя на интерфейсе-посреднике. Адрес **должен** быть индивидуальным.

Поля ICMP

- Type - Extended Echo Request. Для ICMPv4 имеет значение 42, для ICMPv6 - 160.
- Code - **должен** устанавливаться в 0 при передаче, на приемной стороне **должен** игнорироваться.
- Checksum - для ICMPv4 см. RFC 792, для ICMPv6 - RFC 4443.
- Identifier - идентификатор, служащий для сопоставления отклика с запросом. Может иметь значение 0.
- Sequence Number - порядковый номер, служащий для сопоставления отклика с запросом. Может иметь значение 0.
- Reserved - резервное поле, **должно** устанавливаться при передаче и игнорироваться при получении.
- L (локальный) - бит L устанавливается в тех случаях, когда проверяемый интерфейс размещается на узле-посреднике и сбрасывается, если проверяемый интерфейс расположен на узле, напрямую соединенном с посредником.
- ICMP Extension Structure - структура, указывающая проверяемый интерфейс.

Определение ICMP Extension Structure дано в разделе 7 [RFC4884]. В соответствии с RFC 4884 эта структура включает один заголовок Extension Header, за которым следует один или множество объектов. При использовании в сообщениях ICMP Extended Echo Request структура **должна** включать ровно один экземпляр Interface Identification Object (см. параграф 2.1).

Если бит L установлен, Interface Identification Object может указывать проверяемый интерфейс по имени, индексу или адресу. Если бит L сброшен, Interface Identification Object **должен** указывать адрес проверяемого интерфейса.

Если Interface Identification Object указывает адрес проверяемого интерфейса, этот адрес может относиться к любому семейству. Например, сообщение ICMPv4 Extended Echo Request может включать Interface Identification Object, указывающий проверяемый интерфейс адресом IPv4, IPv6 или IEEE 802. То же самое относится и к сообщениям ICMPv6 Extended Echo Request.

## 2.1. Объект идентификации интерфейса

Interface Identification Object указывает зондируемый интерфейс его именем, индексом или адресом. Подобно другим объектам ICMP Extension, этот идентификатор включает Object Header и Object Payload. Заголовок Object Header содержит следующие поля:

- Class-Num - Interface Identification Object (3);
- C-Type - значение 1 задает идентификацию интерфейса по имени, 2 - по индексу и 3 - по адресу;
- Length - указывает размер объекта в октетах с учетом Object Header и Object Payload.

Если Interface Identification Object указывает интерфейс по имени, поле Object Payload **должно** содержать имя интерфейса в соответствии с [RFC7223]. Если Object Payload не завершается на 32-битовой границе, поле **должно** быть дополнено символами ASCII NULL.

Если Interface Identification Object указывает интерфейс индексом, поле length имеет значение 8, а поле данных содержит if-index [RFC7223].

Если Interface Identification Object указывает зондируемый интерфейс адресом, поле данных имеет вид, показанный на рисунке 2.

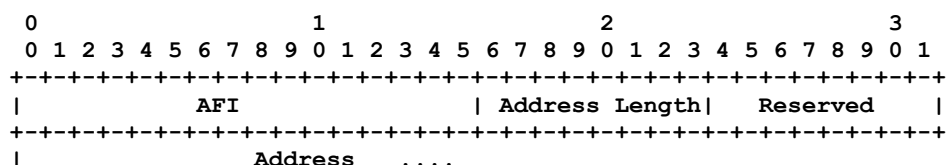


Рисунок 2. Interface Identification Object - C-Type 3.

Поля данных показаны ниже.

### AFI<sup>1</sup>

Это 16-битовое поле указывает тип адреса в поле Address. Доступные значения приведены в реестре IANA [Address Family Numbers](http://www.iana.org/assignments/address-family-numbers).

<sup>1</sup>Address Family Identifier - идентификатор семейства адресов.

**Address Length**

Число значимых байтов в поле Address (это поле может включать байты заполнения).

**Reserved**

Это поле **должно** устанавливаться в 0 при передаче и игнорироваться на приемной стороне.

**Address**

Это поле переменного размера представляет адрес, связанный с проверяемым интерфейсом. Если поле адреса не завершается на 32-битовой границе, оно **должно** быть дополнено нулями.

### 3. Отклик ICMP Extended Echo

Сообщения ICMP Extended Echo Reply определены для ICMPv4 и ICMPv6. Подобно остальным сообщениям ICMP, эти отклики инкапсулируются с использованием заголовка IP (ICMPv4 инкапсулируется в IPv4, ICMPv6 - в IPv6).

Сообщение ICMP Extended Echo Reply показано на рисунке 3.

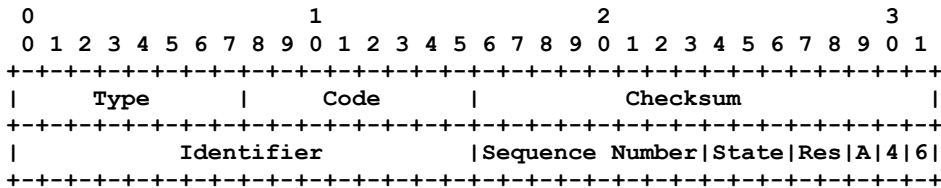


Рисунок 3. Сообщение ICMP Extended Echo Reply.

Поля заголовка IP

- Source Address - копируется из поля Destination Address в заголовке Extended Echo Request.
- Destination Address - копируется из поля Source Address в заголовке Extended Echo Request.

ICMP fields:

- Type - Extended Echo Reply (для ICMPv4 имеет значение 43, для ICMPv6 - 161).
- Code - может принимать значение 0 (No Error - нет ошибок), 1 (Malformed Query - непригодная форма запроса), 2 (No Such Interface - нет такого интерфейса), 3 (No Such Table Entry - нет такой записи в таблице) и 4 (Multiple Interfaces Satisfy Query - запросу соответствует множество интерфейсов).
- Checksum - для ICMPv4 см. RFC 792, для ICMPv6 - RFC 4443.
- Identifier - копируется из поля Identifier соответствующего запроса Extended Echo.
- Sequence Number - копируется из поля Sequence Number соответствующего запроса Extended Echo.
- State - если поле Code отлично от 0, это поле **должно** иметь значение 0 и игнорироваться при получении. Если проверяемый интерфейс находится на узле-посреднике, это поле **должно** иметь значение 0 и игнорироваться при получении. В остальных случаях это поле отражает состояние таблицы ARP или записи Neighbor Cache, связанной с проверяемым интерфейсом. Состояние может указываться значением 0 (Reserved - резерв), 1 (Incomplete - неполная запись), 2 (Reachable - доступен), 3 (Stale - устарел), 4 (Delay - задержка), 5 (Probe - проба) или 6 (Failed - отказ).
- Res - это поле **должно** устанавливаться в 0 и игнорироваться приемной стороной.
- A (Active) - бит A устанавливается, если Code = 0, проверяемый интерфейс размещен на узле-посреднике и активен. В остальных случаях флаг A сбрасывается.
- 4 (IPv4) - бит 4 устанавливается при установленном флаге A, если проверяемый интерфейс относится к IPv4.
- 6 (IPv6) - бит 6 устанавливается при установленном флаге A, если проверяемый интерфейс относится к IPv6.

### 4. Обработка сообщений ICMP

Когда узел получает сообщение ICMP Extended Echo Request и выполняется любое из приведенных ниже условий, узел **должен** отбросить это сообщение, не уведомляя об этом отправителя.

- Узел не понимает сообщений ICMP Extended Echo Request.
- На узле явно не включена функциональность ICMP Extended Echo.
- Входящий запрос ICMP Extend Echo содержит адрес отправителя (Source Address), который не уполномочен явно устанавливать флаг L в сообщениях ICMP Extended Echo Request.
- Входящий запрос ICMP Extend Echo содержит адрес отправителя (Source Address), который не уполномочен явно для указанного типа ICMP Extended Echo Request (т. е. ifName, IfIndex или Address).
- Поле Source Address во входящем сообщении содержит адрес, не являющийся индивидуальным.
- Поле Destination Address во входящем сообщении содержит групповой адрес.

В остальных случаях получивший ICMPv4 Extended Echo Request узел **должен** сформировать сообщение ICMP Extended Echo Reply как показано ниже:

- установлен флаг DF<sup>1</sup> (1);
- флаг More Fragments сброшен (0);

<sup>1</sup>Don't Fragment - не фрагментировать.

- Fragment Offset = 0;
- TTL = 255;
- Protocol = ICMP.

Получивший сообщение ICMPv6 Extended Echo Request узел **должен** создать отклик ICMPv6 Extended Echo Reply :

- Hop Limit = 255
- Next Header - ICMPv6

В любом случае отвечающий узел **должен** выполнить перечисленные ниже действия.

- Скопировать поле Source Address из сообщения Extended Echo Request в поле Destination Address отклика Extended Echo Reply.
- Скопировать поле Destination Address из сообщения Extended Echo Request в поле Source Address отклика Extended Echo Reply.
- Установить для кода DiffServ значение CS0 [RFC4594].
- Установить для ICMP Type значение Extended Echo Reply.
- Скопировать поле Identifier из сообщения Extended Echo Request в сообщение Extended Echo Reply.
- Скопировать поле Sequence Number из сообщения Extended Echo Request в сообщение Extended Echo Reply.
- Установить поле Code в соответствии с параграфом 4.1.
- Установить в поле State значение 0.
- Сбросить флаги A, 4 и 6.
- Если (1) поле Code имеет значение 0 (нет ошибок), (2) флаг L установлен и (3) проверяемый интерфейс активен, установить бит A и один из битов 4 и 6.
- Если (1) поле Code имеет значение 0 (нет ошибок), а бит L сброшен, значение поля State установить в соответствии с состоянием таблицы ARP или записи Neighbor Cache для проверяемого интерфейса.
- Установить подходящее значение поля Checksum.
- Отправить сообщение ICMP Extended Echo Reply получателю.

#### 4.1. Обработка поля Code

В поле Code **должно** устанавливаться значение 1 (Malformed Query) при выполнении любого из условий:

- запрос ICMP Extended Echo Request не включает ICMP Extension Structure;
- структура ICMP Extension Structure не включает ровно один Interface Identification Object;
- бит L сброшен, а Interface Identification Object указывает проверяемый интерфейс по ifName или ifIndex;
- иные ошибки в форме запроса.

В поле Code **должно** устанавливаться значение 2 (No Such Interface), если флаг L и ICMP Extension Structure не указывают интерфейс, размещенный на узле-посреднике.

В поле Code **должно** устанавливаться значение 3 (No Such Table Entry), если флаг L сброшен, а адрес, указанный в Interface Identification Object, не присутствует в таблице IPv4 ARP или IPv6 Neighbor Cache.

В поле Code **должно** устанавливаться значение 4 (Multiple Interfaces Satisfy Query) при выполнении любого из условий:

- флаг L установлен, а ICMP Extension Structure указывает более одного интерфейса на узле-посреднике;
- флаг L сброшен, а адрес, указанный в Interface Identification Object, отображается на несколько записей IPv4 ARP или IPv6 Neighbor Cache.

В остальных случаях для поля Code **должно** устанавливаться значение 0 (No Error).

### 5. Примеры использования

В перечисленных ниже сценариях операторы могут использовать PROBE для определения состояния проверяемого интерфейса, но не могут воспользоваться для этого же утилитой PING. Во всех случаях предполагается наличие между проверяющим интерфейсом и посредником двухсторонней связи. Однако такой связи нет между проверяющим и проверяемым интерфейсом.

- Проверяемый интерфейс не имеет адреса (unnumbered).
- Проверяющий и проверяемый интерфейс не соединены между собой напрямую. Проверяемый интерфейс имеет адрес IPv6 link-local, но не имеет адрес с глобальной значимостью.
- Проверяющий интерфейс поддерживает только IPv4, а проверяемый - только IPv6.
- Проверяющий интерфейс поддерживает только IPv6, а проверяемый - только IPv4.
- По причине отсутствия маршрута проверяющий интерфейс не может достигнуть проверяемого.

## 6. Обновление RFC 4884

В параграфе 4.6 [RFC4884] приведен список расширяемых сообщений ICMP (т. е. сообщений, которые могут включать структуру ICMP Extension). Этот документ добавляет в список сообщения ICMP Extended Echo Request и ICMP Extended Echo Reply.

## 7. Взаимодействие с IANA

Агентство IANA выполнило перечисленные ниже действия.

- Добавлена запись в реестр ICMP Type Numbers

42 Extended Echo Request

Добавлена запись в субреестр Type 42 - Extended Echo Request

(0) No Error

- Добавлена запись в реестр ICMPv6 'type' Numbers

160 Extended Echo Request

В ICMPv6 различаются информационные сообщения и сообщения об ошибках. Данное сообщение является информационным и его значение должно относиться к диапазону 128-255.

Добавлена запись в субреестр Type 160 - Extended Echo Request

(0) No Error

- Добавлена запись в реестр ICMP Type Numbers

43 Extended Echo Reply

Добавлена запись в субреестр Type 43 - Extended Echo Reply

(0) No Error

(1) Malformed Query

(2) No Such Interface

(3) No Such Table Entry

(4) Multiple Interfaces Satisfy Query

- Добавлена запись в реестр ICMPv6 'type' Numbers

161 Extended Echo Reply

В ICMPv6 различаются информационные сообщения и сообщения об ошибках. Данное сообщение является информационным и его значение должно относиться к диапазону 128-255.

Добавлена запись в субреестр Type 161 - Extended Echo Reply

(0) No Error

(1) Malformed Query

(2) No Such Interface

(3) No Such Table Entry

(4) Multiple Interfaces Satisfy Query

- Добавлена запись в реестр ICMP Extension Object Classes and Class Sub-types

(3) Interface Identification Object

Добавлены перечисленные ниже C-типы в субреестр Sub-types - Class 3 - Interface Identification Object

(0) Reserved

(1) Identifies Interface by Name

(2) Identifies Interface by Index

(3) Identifies Interface by Address

Значения C-Типе выделяются по процедуре FCFS<sup>1</sup> из диапазона 0-255.

Все указанные выше значения кодов выделены по процедуре FCFS из диапазона 0-255.

## 8. Вопросы безопасности

Утилиту PROBE легитимно можно применять для определения

- рабочего состояния интерфейса;
- активных на интерфейсе протоколов (например, IPv4 или IPv6).

<sup>1</sup>First Come First Serve - обслуживание в порядке очередности запросов.



Однако злоумышленники могут использовать PROBE для получения дополнительной информации. Например, с помощью PROBE можно попытаться определить имена интерфейсов. Раскрыв эти имена, злоумышленник может вывести и другую информацию. Например,

- пропускную способность интерфейса;
- тип устройства, поддерживающего интерфейс (например, его производителя);
- версию операционной системы, используемой устройством.

С учетом этих рисков операторам следует создать правила, ограничивающие доступ к функциональности ICMP Extended Echo. Для выполнения этих правил поддерживающие ICMP Extended Echo узлы **должны** иметь перечисленные ниже конфигурационные опции.

- Включение и отключение функциональности ICMP Extended Echo (по умолчанию отключена).
- Определение возможностей установки бита L (по умолчанию установка разрешена, сброс запрещен).
- Определение разрешенных типов запросов - имя, индекс, адрес (по умолчанию все отключено).
- Для каждого разрешенного типа запросов определение префиксов, с которых принимаются запросы ICMP Extended Echo Request.
- Для каждого интерфейса управление восприятием сообщений ICMP Echo Request.

При получении узлом запроса ICMP Extended Echo, поддержка которого не разрешена, сообщение **должно** отбрасываться без уведомления отправителя (см. раздел 4).

При использовании PROBE недопустима утечка информации о VPN<sup>1</sup> в другие сети VPN. Поэтому при получении узлом запроса ICMP Extended Echo для проверяемого интерфейса, относящегося к другой VPN, нежели интерфейс-посредник, узел **должен** возвращать отклик ICMP Extended Echo с кодом 2 (No Such Interface).

Для защиты локальных ресурсов реализациям **следует** ограничивать скорость входящих сообщений ICMP Extended Echo Request.

## 9. Литература

### 9.1. Нормативные документы

- [RFC792] Postel, J., "Internet Control Message Protocol", STD 5, [RFC 792](#), DOI 10.17487/RFC0792, September 1981, <<https://www.rfc-editor.org/info/rfc792>>.
- [RFC826] Plummer, D., "Ethernet Address Resolution Protocol: Or Converting Network Protocol Addresses to 48.bit Ethernet Address for Transmission on Ethernet Hardware", STD 37, [RFC 826](#), DOI 10.17487/RFC0826, November 1982, <<https://www.rfc-editor.org/info/rfc826>>.
- [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>>.
- [RFC4443] Conta, A., Deering, S., and M. Gupta, Ed., "Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification", STD 89, RFC 4443, DOI 10.17487/RFC4443, March 2006, <<https://www.rfc-editor.org/info/rfc4443>>.
- [RFC4861] Narten, T., Nordmark, E., Simpson, W., and H. Soliman, "Neighbor Discovery for IP version 6 (IPv6)", RFC 4861, DOI 10.17487/RFC4861, September 2007, <<https://www.rfc-editor.org/info/rfc4861>>.
- [RFC4884] Bonica, R., Gan, D., Tappan, D., and C. Pignataro, "Extended ICMP to Support Multi-Part Messages", [RFC 4884](#), DOI 10.17487/RFC4884, April 2007, <<https://www.rfc-editor.org/info/rfc4884>>.
- [RFC7223] Bjorklund, M., "A YANG Data Model for Interface Management", RFC 7223<sup>2</sup>, DOI 10.17487/RFC7223, May 2014, <<https://www.rfc-editor.org/info/rfc7223>>.
- [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>>.

### 9.2. Дополнительная литература

- [RFC2151] Kessler, G. and S. Shepard, "A Primer On Internet and TCP/IP Tools and Utilities", FYI 30, RFC 2151, DOI 10.17487/RFC2151, June 1997, <<https://www.rfc-editor.org/info/rfc2151>>.
- [RFC4594] Babiarz, J., Chan, K., and F. Baker, "Configuration Guidelines for DiffServ Service Classes", RFC 4594, DOI 10.17487/RFC4594, August 2006, <<https://www.rfc-editor.org/info/rfc4594>>.

## Приложение А. Программа PROBE

Приложение PROBE принимает входные параметры, устанавливает счетчик и запускает цикл с выходом по нулевому значению счетчика. В каждом шаге цикла PROBE передает одно сообщение ICMP Extended Echo Request, декрементирует счетчик, запускает таймер и ждет. Запрос ICMP Extended Echo Request включает идентификатор и порядковый номер.

Если полученное в ответ сообщение ICMP Extended Echo Reply содержит такие же значения Identifier и Sequence Number, программа PROBE возвращает полученную информацию пользователю. Однако в каждом шаге цикла PROBE ждет завершения отсчета таймера, независимо от получения отклика Extended Echo.

<sup>1</sup>Virtual Private Network - виртуальная частная сеть.

<sup>2</sup>Заменен RFC 8343. *Прим. перев.*

Программа PROBE принимает следующий параметры:

- счетчик;
- время ожидания;
- адрес проверяющего интерфейса;
- счетчик интервалов маршрутизации (hop);
- адрес интерфейса-посредника;
- значение бита L;
- идентификатор проверяемого интерфейса.

Счетчик указывается положительным целым числом (по умолчанию 3) и определяет число итераций упомянутого выше цикла PROBE.

Время ожидания задается положительным целым числом (по умолчанию 1) и определяет продолжительность отсчета упомянутого выше таймера в секундах.

Адрес проверяющего интерфейса задает поле Source Address в пакете ICMP Extended Echo Request. Это **должен** быть индивидуальный (unicast) адрес интерфейса проверяющего узла.

Адрес интерфейса-посредника указывает интерфейс, которому передается сообщение ICMP Extended Echo Request. Это должен быть индивидуальный адрес IPv4 или IPv6. Для адреса IPv4 программа PROBE будет передавать сообщение ICMPv4, для IPv6 - сообщение ICMPv6.

Бит L указывается логическим значением (TRUE для случая размещения проверяемого интерфейса и посредника на одном узле, FALSE в противном случае).

Идентификатор проверяемого интерфейса указывает этот интерфейс с помощью

- имени;
- адреса из любого семейства (например, IPv4, IPv6, IEEE 802, 48-битовый MAC, 64-битовый MAC);
- if-index.

Если идентификатором проверяемого интерфейса является адрес, он не обязан относиться к тому же семейству адресов, что и адрес интерфейса-посредника. Например, PROBE принимает адрес IPv4 для посредника и IPv6 для проверяемого узла.

## **Благодарности**

Спасибо Sowmini Varadhan, Jeff Haas, Carlos Pignataro, Jonathan Looney, Dave Thaler, Mikio Hara, Joel Halpern, Yaron Sheffer, Stefan Winter, Jean-Michel Combes, Amanda Barber и Joe Touch за рецензирование документа.

## **Адреса авторов**

### **Ron Bonica**

Juniper Networks  
2251 Corporate Park Drive  
Herndon, Virginia 20171  
United States of America  
Email: [rbonica@juniper.net](mailto:rbonica@juniper.net)

### **Reji Thomas**

Juniper Networks  
Elnath-Exora Business Park Survey  
Bangalore, Karnataka 560103  
India  
Email: [rejithomas@juniper.net](mailto:rejithomas@juniper.net)

### **Jen Linkova**

Google  
1600 Amphitheatre Parkway  
Mountain View, California 94043  
United States of America  
Email: [furry@google.com](mailto:furry@google.com)



**Chris Lenart**

Verizon

22001 Loudoun County Parkway

Ashburn, Virginia 20147

United States of America

Email: [chris.lenart@verizon.com](mailto:chris.lenart@verizon.com)

**Mohamed Boucadair**

Orange

Rennes 35000

France

Email: [mohamed.boucadair@orange.com](mailto:mohamed.boucadair@orange.com)

**Перевод на русский язык**

Николай Малых

[nmalykh@gmail.com](mailto:nmalykh@gmail.com)