Internet Engineering Task Force (IETF)
Request for Comments: 6242
Obsoletes: 4742
Category: Standards Track
ISSN: 2070-1721

M. Wasserman
Painless Security, LLC

June 2011

Использование протокола NETCONF на базе SSH
Using the NETCONF Protocol over Secure Shell (SSH)

Тезисы

Этот документ описывает использование протокола настройки NETCONF[1] в защищенных сеансах SSH[2], как подсистемы SSH. Документ отменяет действие RFC 4742.

Статус документа

Этот документ не является спецификацией проекта стандарта Internet и публикуется с информационными целями.

Документ является результатом работы IETF[3] и представляет согласованное мнение сообщества IETF. Документ был вынесен на открытое обсуждение и одобрен для публикации IESG[4]. Не все документы, одобренные IESG, претендуют на статус тех или иных стандартов Internet (см. раздел 2 документа RFC 5741).

Информация о статусе этого документа, обнаруженных ошибках и способах обратной связи доступна по ссылке http://www.rfc-editor.org/info/rfc6242.

Оглавление

1. Введение

Протокол NETCONF [RFC6241] на основе формата XML служит для настройки сетевого оборудования. Протокол NETCONF определен как сеансовый и не зависимый от транспорта с возможностью отображения на множество протоколов сеансового или транспортного уровня. В этом документе определяется использование NETCONF в сеансах SSH на основе соединений SSH [RFC4254] с транспортным протоколом SSH [RFC4253]. Это отображение позволяет пользователю или приложению применять NETCONF в защищенных сеансах.

Хотя в документе приводятся конкретные примеры передачи сообщений NETCONF через соединение SSH, применение этого транспорта не ограничивается показанными в примерах сообщениями. Этот транспорт может применяться для любых сообщений NETCONF.

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), **возможно** (MAY), **необязательно** (OPTIONAL) в данном документе интерпретируются в соответствии с RFC 2119 [RFC2119].

3. Начало работы NETCONF на базе SSH

Для запуска NETCONF на основе SSH клиент SSH сначала будет организовывать транспортное соединение SSH с использованием транспортного протокола SSH, а клиент и сервер SSH будут обмениваться ключами для защиты

[1]Network Configuration Protocol.
[2]Secure Shell — защищенная командная оболочка (среда).
[3]Internet Engineering Task Force.
[4]Internet Engineering Steering Group.

целостности и шифрования сообщений. После этого клиент SSH будет вызывать службу ssh-userauth для проверки подлинности пользователя в соответствии с протоколом аутентификации SSH [RFC4252]. После успешной аутентификации клиент SSH будет вызывать службу ssh-connection, называемую также протоколом соединений SSH.

Имя пользователя, представленное реализацией SSH, будет доступно уровню сообщений NETCONF (как NETCONF username) без изменения. Если имя пользователя не соответствует требованиям NETCONF [RFC6241] (т. е. не может быть представлено в XML), сессия SSH **должна** быть разорвана. Любые преобразования аутентифицированного отождествления клиента SSH, выполняемые сервером SSH (например, через службы аутентификации или отображение на учетные записи в системе), выходят за рамки этого документа.

После организации соединения SSH клиент будет создавать канал типа session, что приведет к организации сеанса SSH.

После организации сессии SSH клиент NETCONF будет вызывать NETCONF, как подсистему SSH с именем netconf. Поддержка подсистемы является функцией SSH версии 2 (SSHv2) и отсутствует в SSHv1. Работа NETCONF в качестве подсистемы SSH избавляет от необходимости написания сценариев (script) для распознавания системных приглашений (prompt) и позволяет пропустить дополнительную информацию тпа системных сообщений, выдаваемых при запуске интерпретатора команд (shell).

Для упрощения идентификации и фильтрации межсетевыми экранами и другими устройствами трафика NETCONF серверы NETCONF по умолчанию **должны** предоставлять доступ к подсистеме netconf только в случаях организации сессии SSH через выделенный IANA порт TCP 830. Серверам **следует** обеспечивать возможность доступа к подсистеме netconf SSH через другие порты.

Пользователь (или приложение) может применять для обращения к NETCONF, как подсистеме SSH, через выделенный IANA порт с использованием команды вида:

```
[user@client]$ ssh -s server.example.org -p 830 netconf
```

Отметим, что опция -s задает выполнение команды netconf, как подсистемы SSH.

3.1. Обмен информацией о возможностях

Как сказано в [RFC6241], сервер NETCONF указывает свои возможности путем передачи документа XML с элементом <hello> сразу после организации сессии NETCONF. Клиент NETCONF может проанализировать это сообщения для определения возможностей NETCONF, поддерживаемых сервером NETCONF.

Как указано в [RFC6241], клиент NETCONF также передает документ XML с элементом <hello> для индикации возможностей клиента NETCONF серверу NETCONF. Документ с элементом <hello> является первым документом XML, который клиент NETCONF передает после организации сессии NETCONF.

Приведенный ниже пример показывает обмен информацией о возможностях. Данные от клиента NETCONF отмечены символами C:, а данные от сервера NETCONF — символами S:.

```
S: <?xml version="1.0" encoding="UTF-8"?>
S: <hello xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
S:   <capabilities>
S:     <capability>
S:       urn:ietf:params:netconf:base:1.1
S:     </capability>
S:     <capability>
S:       urn:ietf:params:ns:netconf:capability:startup:1.0
S:     </capability>
S:   </capabilities>
S:   <session-id>4</session-id>
S: </hello>
S: ]]>]]>

C: <?xml version="1.0" encoding="UTF-8"?>
C: <hello xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
C:   <capabilities>
C:     <capability>
C:       urn:ietf:params:netconf:base:1.1
C:     </capability>
C:   </capabilities>
C: </hello>
C: ]]>]]>
```

Хотя в примере показано, что клиент NETCONF передает сообщение <hello> вслед за аналогичным сообщением сервера, на деле обе стороны передают свои сообщения сразу после инициализации подсистемы NETCONF (возможно, одновременно).

4. Использование NETCONF на базе SSH

Сессия NETCONF на базе SSH включает клиента и сервер NETCONF, обменивающихся завершенными документами XML. После того, как сессия организована и произошел обмен информацией о возможностях, клиент NETCONF будет передавать завершенные документы XML с элементами <rpc>, а сервер будет отвечать завершенными документами XML с элементами <rpc-reply>.

4.1. Протокол кадрирования

В предыдущей версии этого документа последовательность символов]]>]]> была определена в качестве разделителя сообщений в предположении, что такая последовательность не будет встречаться в корректных документах XML. Однако это допущение оказалось ошибочным и указанная последовательность символв может встречаться в атрибутах, комментариях и инструкциях по обработке XML. Для решения проблемы и обеспечения совместимости с имеющимися реализациями в этом документе определяется протокол кадрирования.

После сообщения <hello> **должен** следовать разделитель]]>]]>. При получении сообщения <hello> приемная сторона концептуально передает его на уровень сообщений (Messages). Если обе стороны анонсировали возможность :base:1.1, далее в сессии NETCONF используется механизм кадрирования chunked (4.2. Механизм кадрирования Chunked), в противном случае - старый механизм (4.3. Механизм кадрирования End-of-Message) is used.

4.2. Механизм кадрирования Chunked

Этот механизм представляет все сообщения NETCONF с использованием chunked-кадрирования. Сообщения следуют правилу ABNF [RFC5234] Chunked-Message:

```
Chunked-Message = 1*chunk
                  end-of-chunks

chunk           = LF HASH chunk-size LF
                  chunk-data
chunk-size      = 1*DIGIT1 0*DIGIT
chunk-data      = 1*OCTET

end-of-chunks   = LF HASH HASH LF

DIGIT1          = %x31-39
DIGIT           = %x30-39
HASH            = %x23
LF              = %x0A
OCTET           = %x00-FF
```

Поле chunk-size представляет собой строку десятичных цифр, указывающих число октетов в chunk-data. Нули в начале поля размера не допускаются, а максимальное разрешенное значение chunk-size составляет 4294967295.

Например, сообщение

```
<rpc message-id="102"
     xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <close-session/>
</rpc>
```

может быть представлено в виде (\n указывает символ перевода строки LF)

```
C:  \n#4\n
C:  <rpc
C:  \n#18\n
C:   message-id="102"\n
C:  \n#79\n
C:       xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">\n
C:    <close-session/>\n
C:  </rpc>
C:  \n##\n
```

Концептуально транспортный уровень SSH кодирует сообщения, переданные уровнем Messages и декодирует сообщения из канала SSH перед их отправкой уровню Messages.

В примерах chunked-кадрирования показаны все переводы строк (LF), даже если они не являются частью механизма кадрирования. Отметим, что транспорт SSH не интерпретирует содержимого XML, т. е. не заботится о каких-либо необязательных символах LF, относящихся к XML.

Во втором и третьем блоках (chunk), показанных выше, каждая строка завершается символом LF. Для всех строк XML (кроме последней) в этом примере символы LF свитаются частью chunk-data и учитываются в chunk-size.

Отметим отсутствие LF после завершающего <rpc> тега в приведенном выше сообщении. Символ LF, требуемый в начале блока end-of-chunks, следует непосредственно за последним символом > в сообщении.

Если поле или значение chunk-size не пригодно или возникает ошибка при декодировании, партнер **должен** разорвать сессию NETCONF путем закрытия соответствующего канала SSH. Реализации **должны** гарантировать отсутствие уязвимостей, связанных с переполнением буферов.

4.3. Механизм кадрирования End-of-Message

Этот механизм поддерживается для обеспечения совместимости с реализациями предыдущих версий документа. Он применяется лишь в тех случаях, когда удаленный партнер не анонсировал базовую версию протокола с поддержкой chunked-кодирования, т. е. реализация NETCONF поддерживает лишь :base:1.0.

При использовании этого механизма клиентом и сервером **должна** передаваться специальная последовательность символов]]>]]> после каждого сообщения (документ XML) в обмене NETCONF. Концептуально транспортный уровень SSH передает все данные между парой последовательностей]]>]]> уровню Messages.

Сессия NETCONF на базе SSH с использованием совместимого с прежними версиями кадрирования end-of-message для получения набора конфигурационных данных может иметь вид:

```
C: <?xml version="1.0" encoding="UTF-8"?>
C: <rpc message-id="105"
C: xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
C:   <get-config>
C:     <source><running/></source>
C:     <config xmlns="http://example.com/schema/1.2/config">
C:      <users/>
C:     </config>
C:   </get-config>
C: </rpc>
C: ]]>]]>
```

```
S: <?xml version="1.0" encoding="UTF-8"?>
S: <rpc-reply message-id="105"
S: xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
S:   <config xmlns="http://example.com/schema/1.2/config">
S:     <users>
S:       <user><name>root</name><type>superuser</type></user>
S:       <user><name>fred</name><type>admin</type></user>
S:       <user><name>barney</name><type>admin</type></user>
S:     </users>
S:   </config>
S: </rpc-reply>
S: ]]>]]>
```

5. Завершение работы подсистемы NETCONF

Завершение работы NETCONF обеспечивается операцией <close-session>. Сервер NETCONF обрабатывает сообщения NETCONF от клиента в порядке их поступления. Когда сервер NETCONF получает операцию <close-session>, ему ответить завершением сессии в канале SSH. Серверу NETCONF **недопустимо** обрабатывать какие-либо сообщения NETCONF после получения операции <close-session>.

В качестве продолжения примера из параграфа 4.2 действующая сессия подсистемы NETCONF может быть завершена, как показано в примере ниже.

```
C: \n#140\n
C: <?xml version="1.0" encoding="UTF-8"?>\n
C: <rpc message-id="106"\n
C:      xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">\n
C:   <close-session/>\n
C: </rpc>
C: \n##\n

S: \n#139\n
S: <?xml version="1.0" encoding="UTF-8"?>\n
S: <rpc-reply id="106"\n
S:           xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">\n
S:   <ok/>\n
S: </rpc-reply>
S: \n##\n
```

6. Вопросы безопасности

Протокол NETCONF применяется для доступа к конфигурационным параметрам и данным состояния для изменения параметров конфигурации, поэтому возможности такого доступа должны предоставляться лишь пользователям или системам, уполномоченным видеть такие данные или менять конфигурацию сервера NETCONF (сетевого устройства).

Для идентификации сервер SSH **должен** быть проверен и аутентифицирован клиентом SSH в соответствии с локальной политикой до того, как от сервера будут переданы или приняты какие-либо данные парольной аутентификации, конфигурационные параметры или данные состояния. Для идентификации клиента SSH он также **должен** быть проверен и аутентифицирован сервером SSH в соответствии с локальной политикой для подтверждения легитимности входящего запроса клиента до начала обмена с ним конфигурационными параметрами или данными состояния. Ни одной из сторон не следует организовывать соединения NETCONF на базе SSH с неизвестным, неожиданным или некорректным отождествлением противоположной стороны.

Данные конфигурации или состояния могут включать конфиденциальную информацию типа имен пользователей или ключей защиты. Поэтому для работы NETCONF требуются коммуникационные каналы, обеспечивающие надежное шифрование для защиты конфиденциальности данных. Этот документ определяет работу протокола NETCONF с использованием отображения SSH, которое поддерживает стойкое шифрования и проверку подлинности.

Данный документ требует, чтобы серверы SSH по умолчанию предоставляли доступ к подсистеме SSH netconf лишь через специальный порт TCP, выделенный IANA для этой цели. Это позволяет идентифицировать и фильтровать трафик NETCONF на базе SSH в межсетевых экранах и других устройствах. Однако это дает возможность атакующему легко идентифицировать трафик NETCONF на базе SSH.

Документ также рекомендует серверам SSH обеспечивать возможность настройки доступа к подсистеме netconf через другие порты. Использование такой конфигурации без соответствующей настройки межсетевых экранов и других устройств может приводить к получения доступа не уполномоченных узлов к подсистеме netconf через межсетевой экран или иную административную границу.

В RFC 4742 принято допущение о том, что последовательность EOM[1]]]>]]> не может появляться в корректно сформированных документах XML, однако это оказалось ошибкой. Последовательность EOM может вызывать проблемы в работе и открывает возможность для атак с использованием специально подготовленных сообщений RPC. Однако эта угроза не представляет опасной. Этот документ продолжает использовать последовательность EOM в начальном сообщении <hello> для предотвращения несовместимости с имеющимися реализациями. Когда обесстороны соединения поддерживают возможность :base:1.1, в сессии NETCONF после приветствия применяется специальное кадрирование (4.2. Механизм кадрирования Chunked), позволяющее предотвратить атаки со вставкой фиктивных сообщений.

7. Взаимодействие с IANA

На основе предыдущей версии этого документа (RFC 4742) агентство IANA выделило порт TCP с номером 830 для принятого по умолчанию порта в сессиях NETCONF на базе SSH.

[1] End-of-message — конец сообщения.

Агентство IANA также выделило идентификатор netconf в качестве имени подсистемы SSH (SSH Subsystem Name), как указано в [RFC4250]

Имя подсистемы	*Документ*
netconf	RFC 4742

Агентство IANA в своих реестрах обновило информацию со ссылкой на этот документ.

8. Благодарности

Ted Goddard был соавтором предыдущих версий этого документа.

Документ был создан с помощью инструмента xml2rfc, описанного в RFC 2629 [RFC2629].

Вного полезных предложений было получено от других членов группы разработчиков NETCONF, включая Andy Bierman, Weijing Chen, Rob Enns, Wes Hardaker, David Harrington, Eliot Lear, Simon Leinen, Phil Shafer, Juergen Schoenwaelder и Steve Waldbusser. Значительную помощь оказали также рецензии Olafur Gudmundsson, Sam Hartman, Scott Hollenbeck, Bill Sommerfeld, Balazs Lengyel, Bert Wijnen, Mehmet Ersue, Martin Bjorklund, Lada Lothka, Kent Watsen и Tom Petch.

9. Литература

9.1. Нормативные документы

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC4250] Lehtinen, S. and C. Lonvick, "The Secure Shell (SSH) Protocol Assigned Numbers", RFC 4250, January 2006.

[RFC4252] Ylonen, T. and C. Lonvick, "The Secure Shell (SSH) Authentication Protocol", RFC 4252, January 2006.

[RFC4253] Ylonen, T. and C. Lonvick, "The Secure Shell (SSH) Transport Layer Protocol", RFC 4253, January 2006.

[RFC4254] Ylonen, T. and C. Lonvick, "The Secure Shell (SSH) Connection Protocol", RFC 4254, January 2006.

[RFC5234] Crocker, D. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", STD 68, RFC 5234, January 2008.

[RFC6241] Enns, R., Ed., Bjorklund, M., Ed., Schoenwaelder, J., Ed., and A. Bierman, Ed., "Network Configuration Protocol (NETCONF)", RFC 6241, June 2011.

9.2. Дополнительная литература

[RFC2629] Rose, M., "Writing I-Ds and RFCs using XML", RFC 2629, June 1999.

Приложение A. Отличия от RFC 4742

В этом приложении перечислены основные отличия этого документа от RFC 4742.

- Добавлен механизм кадрирования chunked для решения проблемы безопасности, связанной с EOM.
- Расширен раздел «Вопросы безопасности» с включением текста, посвященного проблеме EOM.
- Добавлены примеры, иллюстрирующие новое chunked-кодирование, уточнено применение символов перевода строки.
- Добавлен текст об обработке имен пользователей NETCONF в соответствии с требованиями [RFC6241].
- Термины client/server и manager/agent заменены на SSH client/server и NETCONF client/server.
- Используется термин operation (операция) вместо command или message.
- Учтены ошибки, обнаруженные в RFC 4742 с моменту публикации этого документа (см ошибки RFC 4742 на сайте http://www.rfc-editor.org).

Адрес автора

Margaret Wasserman
Painless Security, LLC
356 Abbott Street
North Andover, MA 01845
USA
Phone: +1 781 405-7464
EMail: mrw@painless-security.com
URI: http://www.painless-security.com

Перевод на русский язык
Николай Малых
nmalykh@gmail.com