RFC 2554 - Расширение SMTP сервиса для аутентификации

Оригинальный документ: RFC 2554- SMTP Service Extension for Authentication
Перевод: Цыгырлаш Игорь (15.12.2004)

1. Введение

В данном документе дается определение расширения SMTP сервиса [ESMTP] в соответствии с которым, SMTP клиенты могут указать серверу механизм аутентификации, выполнить аутентификационный протокольный обмен, и опционально согласовать уровень безопасности для последующего взаимодействия. Это расширение является профилем простого протокола защиты и аутентификации (Simple Authentication and Security Layer) [SASL].

2. Соглашения используемые в документе

В примерах, "C:" и "S:" обозначают строки отправляемые клиентом и сервером соответственно.

Ключевые слова "ДОЛЖЕН" ("MUST"), "НЕ ДОЛЖЕН" ("MUST NOT"), "СЛЕДУЕТ" ("SHOULD"), "НЕ СЛЕДУЕТ" ("SHOULD NOT"), и "МОЖЕТ" "MAY" в данном документе толкуются исходя из их определения в RFC 2119 "Key words for use in RFCs to Indicate Requirement Levels" [KEYWORDS].

3. Аутентификационное расширение службы

1. название расширения SMTP службы "Authentication" ("Аутентификация")
2. значение ключевого слова EHLO ассоциированное с данным расширением - "AUTH"
3. ключевое слово AUTH команды EHLO содержит в качестве параметра список поддерживаемых SASL механизмов разделенных пробелами.
4. определена новая SMTP команда - "AUTH"
5. добавлен опциональный параметр, использующий ключевое слово "AUTH", в команде MAIL FROM, и увеличена максимальная длина строки команды MAIL FROM до 500 символов.
6. (6) данное расширение соответствует протоколу передачи (протокол передачи) [SUBMIT].

4. Команда AUTH

Синтаксис:

AUTH mechanism [initial-response]

Аргументы:

- *mechanism* - строка идентифицирующая SASL-механизм аутентификации;
- *initial-response* - опциональный base64-кодированный ответ.

Ограничения:

После успешного выполнения команды AUTH, выполнить её в данном сеансе повторно уже нельзя. После успешного завершения команды AUTH, сервер ДОЛЖЕН (MUST) отклонять любые дальнейшие команды AUTH с кодом ответа 503.

Команда AUTH недопустима в процессе выполнения mail-транзакции.

Подробное обсуждение:

Команда AUTH сообщает серверу механизм аутентификации. Если сервер поддерживает запрашиваемый механизм аутентификации, то он выполняет аутентификационный протокольный обмен, для того чтобы установить подлинность и идентифицировать пользователя. Опционально сервер также договаривается об уровне безопасности. В случае если запрашиваемый механизм аутентификации не поддерживается, сервер отклоняет команду с кодом ответа 504.

Аутентификационный протокольный обмен состоит из серии запросов сервера и ответов клиента зависящих от механизма аутентификации. При получении команды аутентификации от клиента, сервер отправляет клиенту ответ с кодом 334 (ответ о готовности) и текстовой частью содержащей BASE64-закодированную строку. Ответ клиента состоит из BASE64-закодированной строки. Если клиент желает отменить аутентификационный обмен, он должен послать строку с единственным символом "*". Если сервер получает такой ответ, он ДОЛЖЕН (MUST) отменить команду аутентификации AUTH и выдать ответ с кодом 501.

Опциональный аргумент команды AUTH - initial-response используется, для исключения полного цикла обмена сообщениями, в случае, когда используется аутентификационный механизм, не использующий аутентификацию с запросом и подтверждением (аутентификация с запросом (оклик) и подтверждением (отзыв) - это что-то вроде "Оклик: Стой, кто идет! Пароль! Отзыв: Я от Василия Макаровича. - Проходи!", а при использовании AUTH initial-response ситуация примет вид "- Я от Василия Макаровича. - Проходи!" *[Примечание переводчика]*).

Когда initial-response аргумент используется таким механизмом аутентификации, то пустой начальный запрос не посылается клиенту, и сервер использует данные из аргумента initial-response, так же как если бы он был послан в ответ на пустой запрос. В отличие от ответа клиента нулевой длины на 334 ответ сервера, аргумент initial response нулевой длины отправляется как одиночный знак равенства ("=").

Если клиент использует initial-response аргумент команды AUTH с механизмом который шлет данные на оклик сервера, сервер отклоняет AUTH команду с кодом ответа 535.

В случае если сервер не может декодировать BASE64-аргумент, он отклоняет команду AUTH с кодом ответа 501. Если сервер отклоняет утентификационные данные, то ему СЛЕДУЕТ (SHOULD) отклонить команду AUTH с кодом ответа 535, пока более точный код ошибки, такой как один из представленных в разделе 6, не назначен. При успешном завершении клиентом аутентификационного обмена, SMTP сервер отвечает кодом ответа 235.

Название сервиса определенного посредством данного протокольного профиля SASL - "smtp".

Если об уровне безопасности договорено в процессе SASL аутентификационного обмена, то он вступит в силу непосредственно за последовательностью CRLF, завершающей аутентификационный обмен для клиента, и соответственно для сервера это CRLF строки его успешного ответа. После принятия уровня безопасности, SMTP протокол сбрасывается в начальное состояние (состояние SMTP после ответа сервера с кодом 220 приветствия готовности сервиса). Сервер ДОЛЖЕН (MUST) сбросить любые полученные от клиента данные, такие как аргумент команды EHLO, которые не были получены при SASL согласовании. Клиент ДОЛЖЕН (MUST) сбросить любые полученные от сервера данные, такие как список расширений SMTP сервиса, которые не были получены через SASL согласование (за исключением того случая, когда клиент МОЖЕТ (MAY) сравнить список заявленных SASL механизмов перед, и после аутентификации, для того чтобы определить активную down-negotiation атаку). Клиенту СЛЕДУЕТ (SHOULD) послать EHLO команду первой после успешного SASL согласования устанавливающего уровень безопасности.

Сервер не обязан поддерживать какие-либо специфические механизмы аутентификации, и не обязан поддерживать какие-то аутентификационные механизмы вообще. В случае провала команды AUTH, клиент может попробовать другой механизм аутентификации, послав другую команду AUTH.

В случае неудачного выполнения команды AUTH, сервер ДОЛЖЕН (MUST) вести себя так же как если бы клиент и не посылал команду AUTH.

BASE64-кодированная строка в общем может быть произвольной длины. Клиенты и сервера ДОЛЖЕНЫ (MUST) быть способны обрабатывать запросы и ответы, которые могут быть настолько длинными, насколько позволяет поддерживаемый аутентификационный механизм, независимо от ограничений на длину строки у клиента и сервера, которые могут быть в других частях их протоколов реализации.

Примеры:

```
S: 220 smtp.example.com ESMTP server ready
C: EHLO jgm.example.com
S: 250-smtp.example.com
S: 250 AUTH CRAM-MD5 DIGEST-MD5
C: AUTH FOOBAR
S: 504 Unrecognized authentication type.
C: AUTH CRAM-MD5
S: 334
PENCeUxFREJoU0NnbmhNWitOMjNGNndAZWx3b29kLmlubm9zb2Z0LmNvbT4=
C: ZnJlZCA5ZTk1YWVlMDljNDBhZjJiODRhMGMyYjNiYmFlNzg2ZQ==
S: 235 Authentication successful.
```

5. AUTH параметр команды MAIL FROM

AUTH=addr-spec

Аргументы:

addr-spec содержит личность, передавшую сообщение системе доставки, или двухсимвольную последовательность "<>", указывающую на то, что личность неизвестна или недостаточно подтверждена. Подчиняясь ограничениям, наложенным на ESMTP, параметры addr-spec кодируются внутри как xtext. Синтаксис xtext описан в секции 5 [ESMTP-DSN].

Подробное обсуждение:

Опциональный *AUTH* параметр *MAIL FROM* команды позволяет взаимодействующим агентам в доверительном окружении передавать аутентификацию индивидуальных сообщений.

Если сервер доверяет подтвержденной личности клиента и может заявить, что сообщение было первоначально отправлено личностью указанной в аргументе addr-spec, тогда ему (серверу) СЛЕДУЕТ (SHOULD) использовать тот же аргумент addr-spec в качестве параметра команды AUTH при передаче сообщения любому серверу поддерживающему аутентификационное расширение.

MAIL FROM параметр в случае когда AUTH=<> говорит о том, что автор сообщения неизвестен. При этом, сервер НЕ ДОЛЖЕН (MUST NOT) рассматривать сообщение как первоначально отправленное клиентом.

Если AUTH параметр команды MAIL FROM отсутствует, но была установлена подлинность клиента, и сервер полагает, что сообщение было первоначально отправлено клиентом, то сервер МОЖЕТ (MAY) использовать личность клиента как значение аргумента addr-spec в AUTH параметре при передаче сообщения любому серверу, поддерживающему AUTH расширение.

Если сервер недостаточно доверяет аутентифицированной сущности клиента, или клиент не был аутентифицирован, тогда сервер ДОЛЖЕН (MUST) вести себя так, будто был передан параметр AUTH=<>. Тем не менее, сервер МОЖЕТ (MAY) записать значение параметра AUTH в лог-файл.

Если параметр AUTH=<> был передан явно или получен в результате требований описанных в предыдущем параграфе, тогда сервер ДОЛЖЕН (MUST) использовать параметр AUTH=<> при пересылке сообщения любому серверу поддерживающему AUTH расширение.

Сервер МОЖЕТ (MAY) рассматривать расширение списка рассылки как новую пересылку, используя в качестве AUTH параметра адрес списка рассылки или адрес администрации списка рассылки при передаче сообщения списку подписчиков.

Реализация жестко определяет отношение к клиентам, к которым нет достаточного доверия. В таком случае, реализация определяет только разбор и отбрасывание синтаксически правильного AUTH параметра MAIL FROM команды и Передача в качестве параметра AUTH=<> любым серверам поддерживающим AUTH расширение.

Примеры:

C: MAIL FROM: AUTH=e+3Dmc2@example.com

S: 250 OK

6. Коды ошибок

Следующие коды ошибок используются для обозначения различных ситуаций.

432 требуется передача пароля

Данный ответ на AUTH команду говорит о том, что пользователю необходимо перейти к выбранному механизму аутентификации. Обычно используется механизм аутентификации PLAIN.

534 Механизм аутентификации слишком слаб

Данный ответ на AUTH команду сообщает, что выбранный механизм аутентификации слабее чем допустимый политикой сервера для пользователя.

538 Требуется шифрование для запрошенного механизма аутентификации

Данный ответ на AUTH команду сообщает, что выбранный механизм аутентификации может быть использован, когда основное SMTP соединение является зашифрованным.

454 Временная отказ аутентификации

Данный ответ на AUTH команду говорит о том, что аутентификация не удалась в результате временный отказ сервера.

530 Требуется аутентификация

Данный ответ может быть получен на любую команду кроме AUTH, EHLO, HELO, NOOP, RSET, или QUIT. Он означает что политика сервера требует аутентификации для выполнения запрошенной операции.

7. Официальный синтаксис

Нижеследующий синтаксическая спецификация расширенный Backus-Naur Form (BNF) нотацию как определено в [ABNF].

Все алфавитные символы являются не чувствительными к регистру, когда не сказано другое. Использование символов в верхнем или нижнем регистре определенное для символьных строк употребляется только редакционной ясности. Реализация ДОЛЖНА (MUST) принимать эти строки в не чувствительной к регистру форме.

UPALPHA = %x41-5A ;; верхний регистр: A-Z

```
   LOALPHA = %x61-7A ;; нижний регистр: a-z

   ALPHA = UPALPHA / LOALPHA ;; без учета регистра

   DIGIT = %x30-39 ;; цифры 0-9

   HEXDIGIT = %x41-46 / DIGIT ;; шестнадцатиричные цифры (верхний
регистр)

   hexchar = "+" HEXDIGIT HEXDIGIT

   xchar = %x21-2A / %x2C-3C / %x3E-7E
;; US-ASCII кроме "+", "=", SPACE и CTL

   xtext = *(xchar / hexchar)

   AUTH_CHAR = ALPHA / DIGIT / "-" / "_"

   auth_type = 1*20AUTH_CHAR

   auth_command = "AUTH" SPACE auth_type [SPACE (base64 / "=")]
*(CRLF [base64]) CRLF

   auth_param = "AUTH=" xtext
;; декодированная форма строки типа xtext ДОЛЖНА
(MUST) быть либо addr-spec, либо двумя символами "<>"

   base64 = base64_terminal /
( 1*(4base64_CHAR) [base64_terminal] )

   base64_char = UPALPHA / LOALPHA / DIGIT / "+" / "/"
;; с учетом регистра

   base64_terminal = (2base64_char "==") / (3base64_char "=")

   continue_req = "334" SPACE [base64] CRLF

   CR = %x0C ;; ASCII CR, возврат каретки

   CRLF = CR LF

   CTL = %x00-1F / %x7F ;; любой управляющий ASCII символ и DEL

   LF = %x0A ;; ASCII LF, перевод строки
SPACE = %x20 ;; ASCII SP, пробел
```

8. Ссылки

- [ABNF] Crocker, D. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", RFC 2234, November 1997.

- [CRAM-MD5] Klensin, J., Catoe, R. and P. Krumviede, "IMAP/POP AUTHorize Extension for Simple Challenge/Response", RFC 2195, September 1997.
- [ESMTP] Klensin, J., Freed, N., Rose, M., Stefferud, E. and D. Crocker, "SMTP Service Extensions", RFC 1869, November 1995.
- [ESMTP-DSN] Moore, K, "SMTP Service Extension for Delivery Status Notifications", RFC 1891, January 1996.
- [KEYWORDS] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [SASL] Myers, J., "Simple Authentication and Security Layer (SASL)", RFC 2222, October 1997.
- [SUBMIT] Gellens, R. and J. Klensin, "Message Submission", RFC 2476, December 1998.
- [RFC821] Postel, J., "Simple Mail Transfer Protocol", STD 10, RFC 821, August 1982.
- [RFC822] Crocker, D., "Standard for the Format of ARPA Internet Text Messages", STD 11, RFC 822, August 1982.

9. Размышления о безопасности

В данном разделе обсуждаются вопросы безопасности.

Если клиент использует данное расширение для получения зашифрованного тоннеля с взаимодействующим сервером в небезопасной сети, он должен быть сконфигурирован так, чтобы никогда не отправлять почтовые сообщения серверу, если соединение не аутентифицировано и не зашифровано обоюдно. В противном случае, злоумышленник может украсть сообщение клиента, совершив атаку на SMTP соединение, и притворяться, что сервер не поддерживает расширение аутентификации (Authentication extension), либо не давать выполнить команды AUTH.

Перед началом SASL-согласования, любые протокольные взаимодействия выполняются в незашифрованном виде и могут быть модифицированы злоумышленником. В этой связи, клиенты и сервера ДОЛЖНЫ (MUST) сбрасывать любые данные, полученные до начала SASL согласования завершающегося выбором уровня безопасности.

Данный способ не защищает TCP порт, поэтому атакующий может перенаправить релейное соединение на передающий порт [SUBMIT]. AUTH=<> параметр предотвращает такую атаку переданного сообщения без конвертной аутентификации направленную на получение аутентификационных данных передающего клиента.

Клиент, передающий сообщение может требовать от пользователя прохождения аутентификации всякий раз, когда объявляется подходящий SASL механизм. Поэтому, может быть нежелательным передающему серверу [SUBMIT] объявлять SASL механизм, если его использование не дает никаких выгод клиенту по сравнению с анонимной передачей.

Это расширение не заменяет и не может использоваться вместо сквозных подписей сообщений и зашифрованных систем, таких как S/MIME или PGP. Это расширение направлено на решение проблем отличных от проблем решаемых сквозными системами и имеет следующие ключевые отличия:

(1) данное расширение, как правило, полезно только в доверительном окружении;

(2) защищает целый конверт сообщения, а не только тело сообщения;

(3) подтверждает передачу сообщения, но не авторство содержания сообщения;

(4) данное расширение может дать отправителю некоторые гарантии того, что сообщение было доставлено следующему транзитному участку, в случае, где отправитель обоюдно аутентифицируется со следующим транзитным участком и договаривается о подходящем уровне безопасности.

Дополнительные рассуждения о безопасности рассматриваются в спецификации SASL [SASL].

10. Адрес автора оригинального документа

John Gardiner Myers
Netscape Communications
501 East Middlefield Road
Mail Stop MV-029
Mountain View, CA 94043

EMail: jgmyers@netscape.com

11. Полное изложение авторских прав