Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: December 18, 2010

A. Vesely

June 16, 2010

Verified Hello SMTP extension
draft-vesely-vhlo-06

Abstract

Verified Hello (VHLO) is an SMTP extension for managing authorization by policy, as done for whitelisting messages. The VHLO command verb provides for weak authentication of SMTP clients and policy negotiation.

Policies and reputation are being increasingly used to identify messages worthiness. However, they are currently enforced by rejecting SMTP transactions, or discarding messages. Feedback is scarce, also because reply codes are difficult to interpret automatically. Negotiation is not provided for. VHLO is designed so that servers can provide feedback to their clients about which vouching services or authentication methods they require. Credentials can also be negotiated on the fly, in order to allow clients to learn whether messages will be whitelisted by the receiving server before actually transmitting them. Negotiation and feedback are intended to ease rapid diffusion of popular reputation systems and authentication methods. A IANA register is defined for extending the set of available methods.

The VHLO command is similar to EHLO, but accepts a series of parameters. The sender communicates the mail domain name of the organization on whose behalf it operates, along with any vouching services (VBR) for its reputation. On the other hand, the sending host's affiliation with that mail domain is checked by DNS lookups (MX, PTR, or SPF) or using DKIM. DNSBLs and Greylisting are also considered.

Weakly authenticated clients enjoy an intermediate level of trust: they have no relying privileges, but may attempt to deliver mail to local users, are whitelisted from some filters, and may receive DSNs and feedback-loop abuse reports as needed. However, failing to succesfully negotiate VHLO authentication does not preclude a client's ability to relay mail: It may relay as usual; that is to say, without knowing whether the credentials it tries to provide have any meaning for the receiving server.

Status of this Memo

Copyright Notice

Table of Contents

1. Introduction

The SMTP extension defined by this memo provides a VHLO command verb that takes a registered domain name, instead of the client identity taken by EHLO. The declared Domain identifies the organization that is responsible for sending the messages. Two kinds of verifications are required to validate the VHLO command:

the Domain is trustworthy, and

the SMTP client is affiliated with the Domain.

Not all mail messages are amenable to be transmitted in the framework of a VHLO command, only those transmitted on behalf of the weakly authenticated Domain. If weak authentication succeeds, the client can transmit messages that will enjoy prime delivery (Section 1.1). If it fails, the client is told what requirements it misses, so that its administrators know exactly what to do in order to gain acceptance. Such feedback is deemed necessary and sufficient for triggering widespread deployment of domain whitelists, a.k.a. RHSWL (right hand side whitelists), as discussed below (Section 1.2).

1.1. Prime delivery

The term "prime delivery" is used to indicate that a message is not tagged as spam, quarantined, silently dropped, or delivered to junk folders. Here, a junk folder is one from where unread messages are normally deleted, or moved to another junk folder, without human intervention. In addition, prime delivery implies that messages are not altered in such a way as to make them less visible or discourage users from displaying their content.

In case the message has to be forwarded to another internal or external server, its transmission SHOULD attempt to preserve the trust and reputation that was granted on acceptance, as detailed in Section 4, always reporting failure to relay.

End users may operate their own content filtering. They can do so within their clients, or setting up their own filtering recipes within per-user sections of the Mail Delivery Agent configuration. Prime delivery only concerns stock filters that operate for all users. In case users can configure their mailboxes by making on/off decisions about specific content filters, implementing prime delivery involves dynamically turning off the relevant filters. For the sake of reliability, the delivery agent SHOULD ensure that prime delivery is consistently flagged by Authentication Status [RFC5451] headers, and known IMAP keywords. Administrators should educate their users on how to appropriately whitelist messages flagged that way.

1.2. Domains as branding

DNS domain names can be used as a brand, and reputation records based on them last longer than those based on IP addresses. While a domain is not formally required for sending email messages, Verified Hello provides for a framework where only messages sent on behalf of an authenticated domain are accepted. In this respect, this extension is only useful for relaying messages across domain boundaries, which typically happens after Message Submission [RFC4409].

1.3. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

2. Definition and Registration of the VHLO Extension

According to [RFC5321] provisions, the definition for this extension goes as follows:

- the textual name of this extension is "Verified Hello";

- the EHLO keyword associated with the extension is "VHLO";

- the parameter associated with the EHLO keyword is a random value up to 16 octets long (see Section 3.3.2.1);

- this extension defines one additional verb, VHLO, whose only mandatory parameter is the Domain name of the sender, possibly followed by one parameter for each reputation tag (see Section 3.1);

- VHLO is also defined as one additional parameter to the MAIL verb (see Section 3.3.2.1), no parameters are defined for the RCPT verb;

- supporting the extension affects the behavior of a server and client SMTP as described in Section 3; and

- the maximum length of the MAIL command is increased by 22 octets, while the RCPT command is not affected.

Finally, as required by [RFC4409], this extension is NOT RECOMMENDED on the Submission port.

3. Behavior of SMTP client and server

The VHLO command is used by a client to request prime delivery (Section 1.1) of messages. If the server accepts the command by giving a positive response (see Section 3.3.2), the messages transmitted thereafter are said to be in the framework of that VHLO command.

The framework terminates on any of the following, whichever comes first:

the end of the SMTP session,

a further successful VHLO command, or

a further successful EHLO command.

An SMTP session contains zero or more VHLO frameworks, and each VHLO framework contains zero or more transactions.

An SMTP client MAY issue the VHLO command as part of a session initiation, before initiating a mail transaction. That is to say, right after the EHLO command, or instead of it. (In the latter case, of course, the client has to infer that the server supports this extension by some other means.) Clients MAY attempt the VHLO command various times with different parameters, as long as the receiving server allows further retries (see Section 3.3.4).

If no EHLO command has been issued by the client, the server assumes an EHLO command with an address literal matching the remote address. However, if the client specified the PTR parameter, the server MAY assume an EHLO command with the resolved host name.

Clients failing to issue a successful VHLO command MAY transmit messages using regular transactions, outside of any VHLO framework. If the server supports VHLO, and issued reply codes 550 or 553 to indicate that relaying from the given Domain is not wanted in the current state, then if the client's configuration includes a list of alternative MSAs that it may use as smart hosts in such cases, then the client SHOULD relay through an alternative MSA instead (see Section 5).

After successfully transmitting one or more messages in the framework of a successful VHLO command, a client MAY issue another VHLO command to transmit more messages. At any time in a VHLO framework, except during a transaction, a client MAY issue an EHLO command to transmit messages outside of any VHLO framework. Changing framework is required when new messages are transmitted on behalf of a different

Domain, or with different VHLO parameters.

Messages transmitted in a VHLO framework are subject to the MAIL FROM restriction, and, possibly, to the DKIM-Signature headers existence and verification, the VBR restriction, and any Greylisting restrictions (see Section 3.4). A message satisfying all those restrictions is said to be compatible with the VHLO parameters.

By giving a positive reply to VHLO, a server commits itself to accept messages compatible with the given VHLO parameters, and grants them prime delivery. Prime delivery overrides any other policy that might otherwise encourage the server to discard messages, such as ADSP [RFC5617]. While this section indicates circumstances for the failure of each single check, it is up to the local policy to establish what combinations of successful checks yield positive responses. Missing requirements are communicated to the client as described below (Section 3.3.5).

3.1. Syntax of the VHLO command

The only mandatory argument to VHLO is the Domain. The syntax is as follows:

```
vhlo            = "VHLO" SP Domain *( SP auth-rept-claim) CRLF

auth-rept-claim = auth-rept-tag [ ":" tag-spec-param ]

auth-rept-tag   = "GID" / "MX" / "PTR" / "VBR" / "DKIM" / further-tag

tag-spec-param  = gid-param / vbr-param / dkim-param / further-param
```

where the Domain is the fully-qualified DNS domain name delegated to the entity or organization that is responsible for sending the message(s) that will be transmitted in the framework of this command. Note that, unlike the EHLO command, the Domain is not necessarily the host name of the SMTP client.

The maximum line length of the VHLO command is 1000 octets, including the terminating CRLF.

When the authentication method corresponding to a VBR or DKIM auth-rept-tag fails, it may be recovered automatically as described in Section 3.3.5. The remaining methods defined in this document don't provide for negotiation.

The GID auth-rept-tag and its associated gid-param SHOULD be supplied in the special cases described in sections Greylisting check, and Greylisting restrictions.

The remaining arguments MAY be supplied to authenticate the domain name or provide hints for its reputation. These arguments are supplied spontaneously by the client, up to the maximum line length.

3.2. Server side checks on the Domain

The receiving server SHOULD check that the supplied domain is valid and reckon its reputation. The server is not limited by the checking methods indicated in the parameters. Checks that are normally carried out anyway don't even have a corresponding auth-rept-tag, but are mentioned below.

Some circumstances may require to terminate a VHLO framework and start a new one, with varied Domain or parameters. Typically, only a part of the checks need to be carried out again.

3.2.1. Greylisting check

The GID auth-rept-tag provides the value of a VHLO framework that had been given by this same server or a related MX during a previous SMTP session:

```
gid-param       = original-vhlo-string
```

The receiving server SHOULD check that the original-vhlo-string corresponds to the value that it or a related MX has given as random-string in response to a successful VHLO command. Use of the GID auth-rept-tag is reserved for retrying the transmission of messages that suffered a transient failure in the framework of the corresponding VHLO command, as described in Section 3.3.2.1.

If the server applies Greylisting[greylisting], it MAY use the provided gid-param, if supplied, as an additional key to a group of messages, besides other data items used to implement Greylisting. If using this parameter, the server MUST still check that the other data items correspond, and that the sender accomplishes the directives described in Greylisting restrictions.

The server SHOULD NOT issue a negative response for improper usage of this parameter. However, if bad faith can be ascertained, the server MAY add that knowledge to the sending Domain's reputation. On the other hand, using this parameter eases the task of verifying that a Domain's servers adopt a regular retrying behavior. Such knowledge MAY also be added to the Domain's reputation. It is RECOMMENDED that Domains with enough reputation are whitelisted from Greylisting.

3.2.2. DNSBL check

The server SHOULD check any relevant DNSBL, and, if a DNSBL that the server, according to its policy, considers trustworthy for either rejecting messages or degrading their worthiness, gives a positive match, then the server SHOULD issue a negative 550 response to VHLO. See [RFC5782] for details on this check.

3.2.3. SPF check

Checking SPF SHOULD be omitted when the MX or DKIM parameters are specified by the client. Otherwise, if the server carries out SPF checks, it SHOULD check the supplied Domain using the method described in [RFC4408], and, if that results in a "fail" or "permerror", the server SHOULD issue a negative 550 response. For "temperror" see Section 3.3.3. According to its policy, the server MAY issue a negative response when the result is anything but "pass". However, if the client specified the PTR parameter, then the "none", "neutral", and "softfail" SPF results SHOULD also be accepted.

Administrators of a Domain who do not take responsibility for messages transmitted by specific hosts, even if those hosts would be related to the Domain according to the PTR and 'iprev' checks, SHOULD use SPF to exclude those addresses, so that the SPF check results in "fail".

Note that the so-called "helo check" often gets a result of "none" because [RFC4408] does not provide for SPF (or TXT) RRs to be valid for a whole zone, and many hostmasters omit to define an SPF policy for each host. Unlike EHLO, the Domain argument taken by VHLO points to the sending domain, not the host. Because of the MAIL FROM restriction (Section 3.4.1), no further SPF checks are required for transactions in the framework of this VHLO command.

3.2.4. MX check

The MX auth-rept-tag suggests that the client is connecting from an IP address that belongs to one of the Domain's MX servers. The receiving server SHOULD lookup the MX records of the given Domain and successively lookup the addresses (A or AAAA depending on the connection) of each of the hosts listed therein, until it finds a matching address or the list is exhausted. If no match was found, the server SHOULD issue a negative 550 response.

3.2.5. PTR and 'iprev' checks

The PTR auth-rept-tag suggests that the client is connecting from an IP address that can be resolved backward to an host name under the

given Domain's hierarchy.

The receiving server SHOULD lookup the PTR records for the connecting address and verify that at least one of the returned RRs, after resolving any CNAME, results in a host name whose rightmost part matches the Domain. If no match was found, the server SHOULD issue a negative 550 response.

The server SHOULD also check that the name found thereby resolves forward, possibly through a CNAME, to the connecting address, as indicated by the 'iprev' Authentication Method described in [RFC5451]. In case the 'iprev' check fails, the server SHOULD issue a negative 550 response.

3.2.6. VBR check

The VBR auth-rept-tag provides a list of vouching services:

```
vbr-param       = [ "mc=" type-string ";" "mv=" ] certifier-list

certifier-list  = domain-name *( ":" domain-name )
```

If the receiving server has a list of trusted vouching services, it SHOULD carry out the VBR validation process as it would be done for a VBR-Info header containing the corresponding elements, see [RFC5518]. In particular, the type-string defaults to "all", and the domain to certify is the given Domain. The server SHALL remove from the certifier-list provided by the client any certifier not mentioned in its list of trusted vouching services. If the resulting list is empty, the server SHOULD issue a negative 555 response, passing its full list of trusted vouching services as indicated in Section 3.3.5. Otherwise, the server SHOULD proceed with querying one or more services in the resulting list. If any of those queries fails for non-transient reasons, the server SHOULD issue a 550 response. If all the services in the resulting list fail for a transient reason, the server SHOULD issue either a 455 response (formatted as if the failed services were not trusted) or a 450 or 451 response.

The meaning of the list of trusted vouching services configured within the server is that any single vouch suffices. In case a more complicated logic is needed, e.g. service A and either B or C or else service D, it has to be implemented as an ad-hoc mashup of vouching services to be presented as a single service.

Domains within a closed set may enjoy mutual whitelisting by setting up their own ad-hoc vouching server.

3.2.7. DKIM check

The DKIM auth-rept-tag asserts that all messages transmitted in the given VHLO framework have a valid DKIM-Signature header field whose domain (d) tag matches the Domain in the VHLO command.

The parameter contains additional properties of such signatures:

See [RFC4871] for imported ABNF

```
dkim-param      = sig-s-tag *( ";" sig-tag )
```

where the sig-s-tag is the s=selector string, while the optional sig-tag's are selected parts of the DKIM-Signature header field. Note that the parameter MUST NOT contain any whitespace, although it is allowed in the signature header. At least the sig-s-tag for the selector (and the sig-q-tag if a query method different than "dns/txt" is used) MUST be provided. In addition, in case multiple signatures are present whose domain (d) tag equals the Domain, then the client MUST include a sig-b-tag, that is a b=base64data string containing the signature data, or its first bytes, so that the receiving server can unambiguously identify which signature determines a message's compatibility with the VHLO framework. Any other tag from the DKIM-Signature MAY be present in the parameter.

The receiving server MAY fetch the public key required to verify the DKIM signatures. If the key does not exist, the server SHOULD issue a negative 550 response.

The receiving server may be picky about DKIM signatures; that is to say, it may reject or drop messages based on there being too few signed fields, too weak algorithm used for signing, too old signature timestamp, and similar policy requirements. In such case, it SHOULD verify that the tags accompanying the parameter are sufficient to make a decision, and issue a negative 555 response otherwise, passing the full list of the tags it needs, as indicated in Section 3.3.5. The machine readable part of such response SHALL contain any required sig-tag, with or without a value. Values given for header list, signature timestamp, and expiration date are not meant to be exact, but to specify minimal requirements, and the client may retry with compatible values.

> Header fields (h) are compatible if the dkim-param contains more fields than required by the 555 reply. The dkim-param may contain fields required by the reply even if they are not present in the actual DKIM-Signature, provided that they are not present in the message's header.

Timestamp (t) is compatible if it is actually more recent in the dkim-param than required in the 555 reply.

Expiration time (x) is compatible if it is actually more permissive than required in the 555 reply.

In case the server can determine from the content of the tags present in the parameter that the DKIM-Signature is not adequate for its policy, it SHOULD issue a negative response 550, 553, or 555.

3.3. Responses to the VHLO command

An organization's servers accept incoming mail messages according to some policies. The requisites for according a positive reply to a VHLO command SHOULD NOT be less strict than those for accepting an incoming message. In particular, if a policy states that certain conditions imply that a message would be accepted with some reserves, it should likely state that VHLO is denied under the same conditions.

When processing the optional auth-rept-claim's parameters, the server MUST ignore any parameter whose tag it does not support or understand.

In case of unsuccessful response, the server retains its previous state.

3.3.1. Overview of possible responses

250 Domain OK, greetings and extension list

450 VHLO temporarily unavailable

451 VHLO aborted: error in processing

455 Parameter temporarily unverifiable

500 Syntax error, command unrecognized

501 Syntax error in parameters or arguments

502 Command not implemented

503 Bad sequence of commands

550 Missing required qualification

553 Domain rejected by policy

555 Failed for recoverable reason

3.3.2. Positive response

If the checks carried out on the Domain and the connection indicate that the server will wholeheartedly accept messages from the client, the server returns a 250 reply code. The response is a multi-line response with the same format as the EHLO response (ehlo-ok-rsp in [RFC5321]), with the keywords for all the SMTP extensions available as a consequence of entering this VHLO framework.

Upon a positive response, the client MUST reset any flags and variables associated to SMTP extensions that it may have since previous EHLO or VHLO commands in the same session.

3.3.2.1. VHLO parameter and MAIL FROM command

The server response to the VHLO and EHLO commands includes the VHLO keyword along with a randomly generated token of up to 16 octets. The format of the relevant line is as follows:

```
ehlo-line       = "VHLO" SP random-string

random-string   = 1*16( %d33-60 / %d62-126 )
                ; any CHAR excluding "=", SP, and control
                ; characters.
```

The random string supplied by the server MUST be repeated by the client as the value of the VHLO parameter to the MAIL command, for each transaction in the framework of this VHLO command. This is meant to guard against blind attacks and to ease Greylisting checks.

3.3.3. Transient error responses

If the the server is temporarily unable to carry out any required check on the Domain, it SHOULD return the 451 reply code. Then, the client SHOULD quit the session and retry at a later time.

The server MAY return the 450 reply code to indicate that it is not able or willing to reckon the client's reputation during this session, irrespectively of any parameter supplied. In this case, the client MAY try an EHLO command instead, to transmit messages outside of any VHLO framework.

The server MAY return the 455 reply code to indicate that it is temporarily unable to carry out the checks implied by one or more specific parameters. It is possible that a positive response is given if the client repeats the command using different auth-rept-

claim's or different tag-spec-param's. The text of the response SHOULD indicate the parameters that are still available as described in Section 3.3.5.

3.3.4. Negative responses

If the the server cannot grant prime delivery (Section 1.1) because of a missing parameter or parameter's value in the VHLO command, it SHOULD return the 550 or 555 reply codes indicating the missing parameters and arguments as described in Section 3.3.5.

The server MAY return the 553 reply code to indicate that it will never grant prime delivery for the given Domain to the current client, whatever auth-rept-claim's the client may supply.

The server MUST return the 503 reply code (bad sequence of commands) if a VHLO command is issued while a transaction is active.

Servers that don't support this extension MAY return the 500 or 502 reply codes.

After a 555 reply code, the client MAY retry a VHLO command with the parameters modified accordingly. Otherwise, if it is unable to satisfy the server requirements, the client SHOULD proceed as if it obtained a 500 reply code. It is RECOMMENDED that the client application logs the missing requirements, so that administrators know how to gain access to the given server.

After reply codes 500, 502, 550, and 553, the client MUST NOT attempt more VHLO commands during the current session. In addition, after reply codes 550 and 553, the client SHOULD NOT ever attempt any further VHLO command to an MX server of the current target for messages originating from the given Domain; this implies caching the domains pair in a buffer that will be cleared by either configuration updates or overrun (in theory, VHLO should not be retried until the relevant datum changes in any of the involved servers, including third parties).

After reply codes 500, 502, 550, 553, and 555, the client MAY quit the session and send the message through an alternative relay as described in Section 5. Alternatively, the client MAY try an EHLO command, if it hasn't issued one already, and transmit messages outside of any VHLO framework.

3.3.5. Diagnosis of failed VHLO commands

Normally, a client supplies all the claims that can possibly result in increased reputation, except for line length limitations. VBR's

certifier-list's, for example, might grow quite long and clients may be unable to store them on a single line. However, servers can issue multi-line responses containing the complete list, so that a client can select the correct certifiers to include in the next attempt. As some failures can be worked around automatically, failure responses SHALL contain both human readable text and machine readable text. Formally, reply codes 455, 550, and 555 to the VHLO verb have the following syntax:

See [RFC5234] for imported ABNF

```
Failure-resp    = *( Failure-code "-" [ diag-text ] CRLF )
                  Failure-code [ SP diag-text ] CRLF

Failure-code    = %x34-35 %x35 %x35

diag-text       = hread-text / (":" mread-text)

hread-text      = 1*( %d09 / %d32-57 / %d59-126 ) *VCHAR
                  ; the first character must not be ":"

mread-text      = auth-rept-claim / check-failed

check-failed    = check-keyword ":" check-spec-info

check-keyword   = auth-rept-tag / "SPF"

check-spec-info = hread-text
                  ; a column separated domain name list for VBR,
                  ; a domain name or URL for DNSBL,
                  ; required result or failure reason for SPF,
                  ; required tags or failure reason for DKIM,
                  ; n/a for MX, GID, PTR
```

A server SHOULD NOT vary its requirements during a given session.

If a client manages to issue a successful VHLO command for a given Domain after a previous attempt failed, it MAY store the parameters for future reuse. However, the server requirements MAY be changed in future sessions.

3.4. Restrictions and further server side checks

Messages transmitted in the framework of a successful VHLO command are subject to the restrictions detailed in this section. Clients MUST NOT attempt to break these restrictions. Servers SHOULD check that clients comply.

3.4.1. MAIL FROM restriction

Non-empty arguments of the MAIL FROM commands are restricted to addresses whose domain part consists of the authenticated Domain.

In addition, the server MUST check that the VHLO parameter is included and that the corresponding value matches the random string that the server generated on giving the positive response to the VHLO command.

3.4.2. VBR restriction

If the VHLO command in whose framework the message is received contained a VBR tag, the message MAY have a VBR-Info header. If that header is present, it MUST be compatible with the given vbr-param. Compatible here means that it mentions at least the certifier that the server trusts and verified before accepting the relevant VHLO command.

If a VBR-Info header is not present, the receiving server MAY add one based on the Domain given, the certifiers it trusts and verified, and its guess of the type of content among those mentioned in the RR(s) obtained during the verification query.

3.4.3. DKIM-Signature headers existence and verification

If the VHLO command in whose framework the message is received contained a DKIM tag, the message MUST have a valid DKIM-Signature header field, compatible with the given dkim-param. Compatible here means that the domain (d) of the DKIM-Signature is the same, the selector (s) is the same one given in the parameter, and any sig-tag on the signature that was also present in the VHLO parameter has a value compatible with what has been given in the parameter. Again, compatible means different things for different tags:

Header fields (h) are compatible if the actual signature contains more fields than dkim-param. However, the signature may not contain some fields, present in the dkim-param, but not actually present in the message header, and still be compatible.

Timestamp (t) is compatible if it is actually more recent than advertised in dkim-param.

Expiration time (x) is compatible if it is actually more permissive than advertised in dkim-param.

Signature data (b) is compatible if the actual signature begins with the same sequence of bytes contained in the dkim-param, and

if it this unambiguously identifies it among signatures by the given Domain.

If the server verifies signatures on the fly, the verification fails, and such failure would prevent the message from having a prime delivery (Section 1.1), the server SHOULD reject the message instead.

Note that the server does not need to verify more than one signature.

3.4.4. Greylisting restrictions

If transmission of a message in the framework of a VHLO command fails due to transient conditions (4xx reply codes), and the transmission was not itself a retry, the sending server SHOULD annotate the current VHLO parameter in the message's meta data while it queues the message for further retries. We refer to this piece of data as original-vhlo-string. Typically, a message's meta data includes the envelope and possibly the failure reason, and is used by a server to devise a sending strategy as described in section 4.5.4.1 of [RFC5321]. (Note that we are talking about transient failures in the transmission of a message, i.e. after MAIL, RCPT, DATA, or data completion by <CRLF>.<CRLF>; not the VHLO command.)

The current VHLO parameter should be added to meta data only after the very first failure; in particular, not if a previous attempt to transmit the message has happened before, whether in the framework of a VHLO command or not. This implies that use of VHLO is restricted to hosts who are able to discern new messages from retried attempts.

When attempting to retransmit a queued message that has this original-vhlo-string in its meta data, the sending client SHOULD transmit such string using the GID auth-rept-tag with

```
gid-param       = original-vhlo-string
```

Only messages that share the same original-vhlo-string may be transmitted in the framework of a VHLO command that used the GID auth-rept-tag with that value. This implies that the sending client MUST terminate the current VHLO framework in case the next message's original-vhlo-string differs from the gid-param used to establish it (where no gid-param matches an empty original-vhlo-string.)

4. Forwarding of messages accepted under VHLO

A message accepted in the framework of a VHLO command deserves prime delivery (Section 1.1). However, the receiving server possibly does not host the mailboxes of the relevant recipients directly. For

example, it may be a boundary or secondary exchanger, a vanity address server, or it may be following user-specific forwarding instructions. For this specification, we just distinguish if the message is forwarded within the same organization or to an external domain.

If the message is forwarded internally, all hosts MUST be configured so as to honor the promise of prime delivery that border or secondary exchangers grant on their behalf. If, for whatever reason, prime delivery is not possible, a failure notification MUST be sent to the Return-Path address, if any. Even if sending notifications is expected to be fairly safe at this point, it is RECOMMENDED that any organization-wide policy that can be applied on acceptance produces an on-line rejection rather than a delayed failure notification.

If the message is forwarded to an external domain, the SMTP client SHOULD attempt to transmit it in the framework of a VHLO command, unless either it can determine that the target host does not implement this SMTP extension, or it has some other arrangement with the target host that grants prime delivery (e.g. using strong authentication as provided by [ff]).

VHLO may be used for forwarding in two different ways:

> If the forwarder is affiliated with the original Domain or if the message contains a valid DKIM signature from it, then the message can be sent using the original envelope's originator address. The Domain declared as VHLO parameter is the original one. (This is as "Alias expansion".)
>
> Otherwise, the Domain in the VHLO parameter is the forwarder's domain and the originator address MUST be changed (e.g. using [srs]). (This is as "List expansion".)

Failure to relay MUST always be reported as indicated by [RFC5321]. In particular, any reason that may be considered valid for not issuing a failure notification SHOULD be ruled out before giving a positive reply to the VHLO command.

5. Submission strategy

In order to avoid hassles, several smaller MTAs are configured to use external Mail Submission Agents (MSAs) as smart hosts. One collateral advantage of using Verified Hello is that falling back to smart hosts can be confined to specific cases, depending on the outcome of the weak authentication process. The postmasters of a sending domain can resort to smart hosts while they collect feedback.

Then, for the increased privacy and efficiency that direct delivery yields, they'll have the ability to select what combination of mechanisms and brands will satisfy the majority of their targets, and decide to implement those requirements.

The VHLO command, by allowing to check deliverability in advance, enables clients to use smart hosts optionally. Rather than configuring a fixed mail-out path for certain target domains, relays can dynamically adjust their strategy according to the target host's response to the VHLO command. The list of preferred VBR certifiers provided by a 555 negative response may be used as keys to build a corresponding list of smart hosts that can be used as Mail Submission Agents, provided that the certifiers of each smart host are known.

To implement this strategy, a relay's configuration needs a list of alternative MSAs, consisting in one or more entries containing a host name, a username/password pair, and an optional list of VBR certifiers of that MSA. The latter field should be updated dynamically whenever it does not correspond to the list returned with a 555 negative reply from the smart host; it is RECOMMENDED to log such updates as appropriate. Other means to dynamically select an MSA, and how to determine the default one MAY also be provided for.

6. IANA Considerations

6.1. IANA Mail Parameters

This extension will have to be inserted in the mail-parameters assignments IANA registry. The keyword VHLO should appear

- as a Registry Keyword, along with the "Verified Hello" description, this document's reference, and a "+" for SMTP only, and

- as an SMTP extension keyword that has a parameter, after the "Verified Hello" description column, before the "Random ID" parameter description and this document's reference that terminate its row.

Formally, VHLO is not a service type, as it requires or assumes EHLO.

6.2. IANA VHLO methods

A registry is needed for tracking the auth-rept-tag / check-keyword that must be unique in the diagnostic text. New methods may be defined publishing their own RFCs where semantic and syntactic details are explained, including error response and diagnosis. This

document defines

Keyword	Parameter/Description	Reference
DKIM	Key selector, query method, etc.	[this]
DNSBL	None. Diagnostic only.	[this]
GID	Greylisting ID.	[this]
MX	None. DNS lookup.	[this]
PTR	None. rDNS lookup.	[this]
SPF	None. Diagnostic only.	[this]
VBR	Certifier list.	[this]

Initial registry values

7. Security Considerations

Global communications require that SMTP servers accept mail coming from unknown hosts. This requirement rules out strong authentication schemes, because, by definition, it is not possible to authenticate unknown entities. Historically, Internet protocols granted some trust to any host, since sporting a global IP address was deemed a sufficient credential. When more restrictive criteria became required, a number of mechanisms have emerged for identifying the sender. DNS and rDNS are used to check the relationship between the sender's IP address and its domain. However, using EHLO, the sender's domain can only be guessed at. Some mechanisms, e.g. rDNS, are not universally available, and, although good senders try and facilitate the identification of themselves by setting up DNS as well as they can, receivers provide no feedback on their effort. Since senders don't know which mechanism, if any, would satisfy the requirements of a target server, they can only follow generic guidelines, outdated static policy pages, and rare support team's hints whose validity is not imperishable.

This document proposes an intermediate level of trust. An SMTP client is being authenticated based on weak evidence, originating from the DNS and the TCP layer:

- The IP address of the remote client is known from the TCP layer. Verification of the random string implies it is fairly difficult to forge it.

- Any of the MX, PTR, or SPF checks confirms that the IP address is somehow authorized by the organization who owns the Domain.

- The DNSBL check implies that the IP address is not that of a known attacker.

The two remaining checks, DKIM and VBR, may provide two additional characterizations of the messages being transmitted. DKIM ensures that messages have passed through the domain's signing process, which presumably implies that any sender's local policy has been enforced. In this respect, DKIM can be regarded as an open authorization to impersonate the original Domain for the purpose of forwarding a signed message. See [RFC5617], [RFC5672], and [RFC5863] for further insight on DKIM semantics.

VBR, depending on the certifier's policy, may generically ensure that the sending domain is well behaved. A vouching service may scrutinize the DNS settings of a given domain, verify its whois record, check their spam rate using honeypots, investigate the domain's users, receive and process copies of the abuse reports issued against messages emitted by that domain, verify that reported spammers get blocked according to some policy, or otherwise establish the domain reputation. The possibility to communicate the preferred vouching services may work as an incentive for the advertised service providers.

The authentication provided by this extension is weaker than SMTP Authentication [RFC4954]. Therefore, it SHOULD NOT be used instead of it.

Diagnostic messages provided with negative responses to the VHLO command may disclose acceptance policies of the target domain. This is not considered harmful, since such policies are usually public. Letting a sender know which mechanism failed is a risk only in case of security through obscurity. Mechanisms that are secure by design don't have to be kept secret. The mechanisms considered in this memo only involve DNSBL, SPF, MX, PTR, VBR, and DKIM. However, Verified Hello provides for extensibility of this authentication/reputation (auth-rept) mechanisms base. Giving feedback is important for mechanism management, as it allows popular mechanisms to gain momentum. In addition, some mechanisms reference a different domain that makes explicit assertions about the reputation of the sender's domain. This is where the branding practice comes into play. As the number of domains that give reputation indications may grow much more quickly than the number of mechanisms, feedback is specially important for spreading their popularity. In this respect, Verified Hello is not yet another authentication mechanism. It is a framework for managing those mechanisms.

However, in case a Domain's security structure depends on keeping that information secret, the server should carefully consider what

diagnostic messages it provides to what clients. It is possible to provide VHLO services to selected domains only, and discarding the rest with the reply code 553.

8. References

8.1. Normative References

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

[RFC4871] Allman, E., Callas, J., Delany, M., Libbey, M., Fenton, J., and M. Thomas, "DomainKeys Identified Mail (DKIM) Signatures", RFC 4871, May 2007.

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

[RFC5321] Klensin, J., "Simple Mail Transfer Protocol", RFC 5321, October 2008.

[RFC5518] Hoffman, P., Levine, J., and A. Hathcock, "Vouch By Reference", RFC 5518, April 2009.

8.2. Informative References

[RFC4408] Wong, M. and W. Schlitt, "Sender Policy Framework (SPF) for Authorizing Use of Domains in E-Mail, Version 1", RFC 4408, April 2006.

[RFC4409] Gellens, R. and J. Klensin, "Message Submission for Mail", RFC 4409, April 2006.

[RFC4954] Siemborski, R. and A. Melnikov, "SMTP Service Extension for Authentication", RFC 4954, July 2007.

[RFC5451] Kucherawy, M., "Message Header Field for Indicating Message Authentication Status", RFC 5451, April 2009.

[RFC5617] Allman, E., Fenton, J., Delany, M., and J. Levine, "DomainKeys Identified Mail (DKIM) Author Domain Signing Practices (ADSP)", RFC 5617, August 2009.

[RFC5672] Crocker, D., "RFC 4871 DomainKeys Identified Mail (DKIM) Signatures -- Update", RFC 5672, August 2009.

[RFC5782] Levine, J., "DNS Blacklists and Whitelists", RFC 5782,

February 2010.

[RFC5863] Hansen, T., Siegel, E., Hallam-Baker, P., and D. Crocker, "DomainKeys Identified Mail (DKIM) Development, Deployment, and Operations", RFC 5863, May 2010.

[ff] FixForwarding.org, "solution proposed", 2009, <http://FixForwarding.org/wiki/solution_proposed>.

[greylisting] Greylisting.org, "Greylisting.org - a great weapon against spammers", 2009, <http://www.greylisting.org/>.

[srs] Libsrs2.org, "libsrs2 - Home", 2004, <http://www.libsrs2.org/>.

Appendix A. Examples

Some examples showing the relevant snippet of client-server dialog.

A.1. Prime delivery message transfer

Complete example where the client successfully transfers a message

```
S: 220 example.com SMTP server ready
C: VHLO example.net
S: 250-example.com greetings example.net
   250 VHLO 0123456789ABCDEF
C: MAIL FROM:<author@example.net> VHLO=0123456789ABCDEF
S: 250 Ok
C: RCPT TO:<dest@example.com>
S: 250 Ok
C: DATA
S: 354 Go ahead
S: From: author@example.net
   To: dest@example.com
   Subject: test

   This is transmitted with prime delivery!
   .
S: 250 Ok
C: QUIT
S: 221 Bye
```

A.2. Failure after DNSBL check

Colons have been replaced in the automatic message to formally preserve machine readability

```
C: VHLO example.net
S: 555-You are blacklisted
   555 :DNSBL:see http_//www.dnsbl.example/query/bl?ip=192.0.2.3
C: QUIT
S: 221 Bye
```

Alternatively, the failure can be signaled as usual. Since feedback plays a minor role for negative (black) vouching, the following is likely to get an equivalent effect.

```
C: VHLO example.net
S: 550-You are blacklisted
   550 see http://www.dnsbl.example/query/bl?ip=192.0.2.3
C: QUIT
S: 221 Bye
```

A.3. Failure on the MAIL FROM restriction check

In this snippet, the domain names are mismatched

```
C: VHLO example.net
S: 250-example.com greetings example.net
   250 VHLO 0123456789ABCDEF
C: MAIL FROM:<user@example.org> VHLO=0123456789ABCDEF
S: 550 Domain origin mismatch
C: QUIT
S: 221 Bye
```

A.4. Automatically finding a common vouching service

In this snippet, the client finds a valid VBR name

```
C: VHLO example.net MX VBR:vouch1.example:vouch2.example
S: 555-we only accept these :VBR:vouch97.example:vouch98.example
   555-:VBR:vouch99.example:vouch100.example:vouch101:example
   555 :VBR:vouch102:example:vouch103:example:vouch104:example
C: VHLO example.net MX VBR:vouch100.example:vouch101.example
S: 250-example.com greetings example.net
   250 VHLO 0123456789ABCDEF
```

A.5. Reattempting Greylisted transmission

On a first attempt the client got greylisted

```
   S: 220 example.com SMTP server ready
   C: VHLO example.net
   S: 250-example.com greetings example.net
      250 VHLO FirstTime
   C: MAIL FROM:<author@example.net> VHLO=FirstTime
   S: 250 Ok
   C: RCPT TO:<dest@example.com>
   S: 450 You are greylisted, retry after 5 mins.
   C: QUIT
   S: 221 Bye

   ... 5 minutes later ...

   S: 220 example.com SMTP server ready
   C: VHLO example.net GID:FirstTime
   S: 250-example.com greetings example.net
      250 VHLO SecondTime
   C: MAIL FROM:<author@example.net> VHLO=SecondTime
   S: 250 Ok
   C: RCPT TO:<dest@example.com>
   S: 250 Ok
   C: DATA
   S: 354 Go ahead
   S: From: author@example.net
      To: dest@example.com
      Subject: test

      This is transmitted after greylisting delay!
      .
   S: 250 Ok
   C: QUIT
   S: 221 Bye
```

A.6. Mandating DKIM usage

In this snippet, the server requires DKIM signature for specific headers. The client might have turned for alternative delivery, EHLO or alternative MSA, if it could not comply. In addition, a common vouching service is automatically found as in the example above (Appendix A.4).

```
C: VHLO example.net VBR:v1.example:v2.example
S: 555-we only accept these :VBR:v97.example:v98.example
   555-:VBR:v99.example:v100.example:v101:example
   555-:VBR:v102:example:v103:example:v104:example
   555 :DKIM:h=to:from:cc:date
C: VHLO example.net VBR:v100.example DKIM:s=mail;h=to:from:cc:date
S: 250-example.com greetings example.net
   250 VHLO 0123456789ABCDEF
```

A.7. Requiring extra DKIM tags

In this snippet, the server wants to know the timestamp and expiration of the signature. Note that this almost certainly will require a new VHLO framework in case further message for that Domain have to be relayed.

```
C: VHLO example.net DKIM:s=mail
S: 555-we want to check signature timestamp and expiration time
   555 :DKIM:t=;x=
C: VHLO example.net DKIM:s=mail;t=1117574938;x=1118006938
S: 250-example.com greetings example.net
   250 VHLO 0123456789ABCDEF
```

Author's Address

Alessandro Vesely
v. L. Anelli 13
Milano, MI 20122
IT

Email: vesely@tana.it