

What's New in LISTSERV® Version 16.5

Copyright © 2018 L-Soft international, Inc.
7 May 2018

LISTSERV 16.5

LISTSERV 16.5 includes all known fixes and patches and between-release enhancements up to 7 May 2018. There are a number of bug fixes and enhancements in LISTSERV itself, and there are also changes and fixes to the WA CGI for the web interface and to the default web templates (default.wwwtpl).

**IMPORTANT: LISTSERV 16.5 requires a valid version 16.5 LAK!
This Release also requires valid Maintenance expiring on 7 May 2018 or later!**

You must obtain and install a LISTSERV version 16.5 product LAK and (for sites with perpetual licensing) an appropriate maintenance LAK, or LISTSERV will not start after the upgrade.

[More information](#)

[Current Supported Operating Systems](#)

Table of Contents

LISTSERV 16.5 Changes for LISTSERV.....	3
Security: SHA-2 password hashing, strong RNG	3
Security: One-time passwords for web interface	3
General Improvement: SYSTEM.WWWTPL	4
Usability: [HPO] SHOW DOWNTIME command	5
Performance, Usability: [HPO] Dynamic Query Cache	5
Performance, Usability: [HPO] SEARCH_PRELOAD	6
Performance, Usability: Faster archive searching	6
Usability: Archive search precedence improvement.....	8
Security, Anti-Spam: TRAPIN now applied to X-CONFIRM command.....	8
GDPR, Usability: GET (COLUMNS command enhancement	9
GDPR, Usability: SYSTEM and NOLIST-* changelogs exposed in LISTSERV FILELIST	10
Security: PW REP expiration message exposed replacement password	10
Usability: Apply sub-list REPRO setting to "Sender:" address	10
Usability: Time specification now allowed for weekly/monthly digests	10
LISTSERV 16.5 Changes for WA	11
Performance: [HPO] Introducing HPO mode for WA.....	11
Usability: [Windows] WA.SYM debugging symbols for WA.EXE	11
Security, Anti-Spam: Reject Subscription Requests Containing URLs	12
Security: New configuration variable to set HSTS maximum age	12
LISTSERV 16.5 Changes for Web Templates.....	12
GDPR: New privacy policy web templates available	12
Installing LISTSERV 16.5.....	13
Current Supported Operating Systems	15
SPECIAL NOTES.....	15
Upgrade Instructions.....	15
Supported Operating Systems.....	15
Support for GDPR Compliance.....	15

LISTSERV 16.5 Changes for LISTSERV

Since the release of LISTSERV 16.0-2017a in February 2017, the following improvements and bug fixes have been made.

Security: SHA-2 password hashing, strong RNG

In LISTSERV 16.5, all new LISTSERV password records are generated with 256-bit SHA-2 hashing and strong random-number generation (cryptographic RNG) by default. Login tickets are also now generated using cryptographic RNG.



Note: Existing passwords hashed with SHA-1 will not be rehashed with SHA-2 unless and until the user changes his or her password. This is because LISTSERV does not have the password available to rehash – it stores only the hash itself.

The use of SHA-2 family hash functions is recommended for US federal agencies by NIST (see *NIST Special Publication 800-131A Revision 1*, which is the current version of NIST's guidance as of the release of LISTSERV 16.5).

Once implemented, SHA-2 hashing cannot be reversed, but all LISTSERV versions since 16.0-2014a understand both SHA-1 and SHA-2 hashes.

Security: One-time passwords for web interface

In previous versions of LISTSERV, WA login cookies contained a masked but unencrypted password (which, should the cookie be opened by accident, would appear to be garbage, but still was not encrypted). LISTSERV 16.5 introduces a new secure WA login cookie feature. Going from an insecure to a secure WA login cookie is completely transparent to users and is enabled by default.

When WA logs in with the user's password, LISTSERV generates a random one-time password and returns it to WA along with the login ticket. WA stores this one-time password in its login cookie, overwriting any prior unencrypted password that might have existed in the cookie.

```
17 Apr 2018 12:59:34 From [ANONYMOUS]: X-LOGIN joe@example.com 10.123.2.30 PW=[redacted]
17 Apr 2018 12:59:34 To [ANONYMOUS]: ***OK*** PFA2F8FA92A1E2A4810 [redacted]
```

WA will use the one-time password the next time it needs to log in. LISTSERV will “burn” the password upon use, and generate a new one for WA to update its login cookie with:

```
17 Apr 2018 14:17:57 From [ANONYMOUS]: X-LOGIN joe@example.com ONETIME 10.123.2.30 PW=[redacted]
17 Apr 2018 14:17:57 To [ANONYMOUS]: ***OK*** OPF36ABF742466929D97 [redacted]
```

The ONETIME option simply states that the supplied password is a one-time password and should not be attempted as a real password (so as not to lock people out if using LDAP for password validation). Note that the ticket now starts with O, indicating that it was obtained from a one-time password.

Users are permitted up to 5 contemporaneous one-time passwords, allowing the concurrent use of up to 5 different devices or browsers.

By default, LISTSERV creates roaming one-time passwords that can be used from any IP address. This is in practice necessary to support home users, who may get a different IP address every few minutes, or indeed with every access. In an enterprise environment, it may make more sense to disable roaming and bind one-time passwords to the IP address they were issued to, by setting:

```
WWW_ONETIME_PW_ROAMING=0
```

Non-roaming passwords are only accepted from the IP address they were issued to; any attempts from a different IP address are rejected, and the password is immediately burned. This provides additional protection from XSS attacks because, even if the browser can somehow be fooled into disclosing the one-time password to a malicious third party, the password will not work from that third party's IP address. And, because tickets expire after 15 minutes (by default), the browser will use the one-time password within that time frame to request a new ticket, forever burning the password.

One-time passwords are stored as a user configuration variable called `*ONETIME*`. You can query its existence, but not see or alter its contents.

```
17 Apr 2018 14:34:11 From LISTSERV@LISTSERV.EXAMPLE.COM: x-logck x getcfg: *
* ***OK***
* *ONETIME*=[redacted]
```

If necessary, one-time passwords can be disabled by setting the `WA_USE_INSECURE_COOKIE` configuration variable as follows:

```
WA_USE_INSECURE_COOKIE=1.
```

Finally, LISTSERV login tickets are now generated cryptographically, using the operating system's internal cryptographic functions.

General Improvement: SYSTEM.WWWTPL

LISTSERV 16.5 combines `DEFAULT.WWWTPL` and `SITE.WWWTPL` into a single file, `SYSTEM.WWWTPL`, which is present only in the web archive directory. This file is refreshed every hour, as well as after a restart and whenever `SITE.WWWTPL` is modified using one of the supported methods, either via the web interface or a `PUT` command via e-mail.



IMPORTANT: If you have a list on the server named "SYSTEM", you **MUST** change the name of that list to something else before installing LISTSERV 16.5, as "SYSTEM" is now a reserved word for list names.

When present, WA will load only this file instead of searching first `SITE` and then `DEFAULT`. More importantly, there is a special `INIT` template at the top of the file, containing all sorts of information that LISTSERV wishes to publish to WA. As an example,

```
>>> INIT Initialization data
+SEL RELEASE_LSV_VERSION 16.5
+SEL RELEASE_RUNNING_UNDER Windows Server 2016 Datacenter x64 Edition
+SEL RELEASE_BUILD_DATE 18 Apr 2018
+SEL RELEASE_BUILD_DATEB 736801
+SEL RELEASE_LSV_TYPE LISTSERV(R) High Performance
+SEL RELEASE_RUNMODE Networked
+SEL RELEASE_INTERNET_PEERS_FILE_VERSION 2018-02-27 13:12:05
+SEL RELEASE_ALIAS_FILE_VERSION 2017-02-16 10:36:58
+SEL RELEASE_VIRUS_DATABASE_VERSION 2018-04-26 10:08:07 (F-Secure Anti-Virus 9.52)
+SEL LITE_NO_SEARCH 0
+SEL LICENSE_LITE 0
+SEL LICENSE_HPO 1
+SEL LICENSE_SCOPE ISP,AVSTATION,MAESTRO(10)
+SEL LICENSE_SERIAL_NUMBER LSOFT-DEV-WINNT-INTEL
+SEL LICENSE_LICENSE_TYPE Permanent
+SEL LICENSE_EXPIRATION_DATE None - perpetual license
+SEL LICENSE_MAINTENANCE_UNTIL 31 Dec 2999, serial number LSOFT-DEV-WINNT-INTEL-MNT
+SEL LICENSE_CAPACITY Unlimited
+SEL LICENSE_MAESTRO_LIMITS 10/day (0 used)
+SEL LICENSE_VERSION 19.0
+SEL LICENSE_PRODUCT_OPTIONS HPO
```

```
+SEL SIGNUP_REQUIRE_SSL 0
+SEL WWW_AUTHINFO_DISABLE 1
+SEL WA_USE_INSECURE_COOKIE 0
+SEL WWW_ARCHIVE_PREFIX /archives/
+SEL HSTS_MAX_AGE 31536005
+SEL WWW_ALLOW_LEGACY_INTERFACE 1
```

If WA finds SYSTEM.WWWTPL, it will load this information from the INIT template instead of issuing queries to LISTSERV.

Usability: [HPO] SHOW DOWNTIME command

LISTSERV 16.5 includes a new, built-in downtime monitor. LISTSERV basically keeps track of when it is up, and LISTSERV maintainers can use the privileged SHOW DOWNTIME command to see when it was down and how good its uptime has been a particular month (or over a period of several months). The command syntax is:

```
SHOW DOWNtime [yyyymm1[-[yyyymm2]]]
```

Given a hypothetical command like

```
SHOW DOWNTIME 201801-201804
```

the output looks like this:

```
Downtime started   Downtime ended      Duration
-----
2018-01-17 20:03   2018-01-17 20:06      4 min
2018-01-17 20:13   2018-01-17 20:13      1 min
2018-03-15 03:06   2018-03-15 03:06      1 min
2018-03-15 03:08   2018-03-15 03:09      2 min
2018-04-12 03:06   2018-04-12 03:06      1 min
2018-04-16 20:01   2018-04-16 20:06      6 min
2018-04-16 20:32   2018-04-16 20:33      2 min
2018-04-16 21:07   2018-04-16 21:08      2 min
2018-04-16 22:18   2018-04-16 22:28     11 min
-----
                        30 min (99.982% uptime)
```

If there is no downtime in the specified period, LISTSERV simply responds,

```
There was no downtime in the specified range.
```

The first time LISTSERV 16.5 is started, it assumes that there has been no downtime to date in the current month, simply because it is a binary state with no “don’t know” option. From that point on, it assumes that every future time period is a downtime period unless marked otherwise as time progresses. DST will cause a false positive when jumping forward by one hour, and will only see 1h of downtime if you are down for 2h during the backward jump.

Performance, Usability: [HPO] Dynamic Query Cache

Dynamic query performance under HPO has been improved by optionally caching queries run for authentication purposes (queries run for mail delivery purposes bypass the cache). The intent is to speed up response time for queries run from WA against lists which use DQL queries for the purpose of defining owners, moderators, etc. LISTSERV will now cache the information after the first run against the data source and use the cache for subsequent queries (e.g., "LISTS MODERATED BY" and "LISTS OWNED BY") for the same list. The cache is available for any DQL data source, including exits.

This improvement includes two new site configuration variables for tuning the DQL cache:

`DYN_QUERY_CACHE_LIMIT`: Sets the maximum query size that will be cached, in kilobytes. Default value of 0 disables the cache.

`DYN_QUERY_CACHE_TTL`: Sets the Time To Live (TTL), in seconds, of cache entries. Defaults to 15 minutes.

The dynamic query cache is disabled by default and must be enabled to take advantage of the performance improvement it offers.

Performance, Usability: [HPO] SEARCH_PRELOAD

LISTSERV 16.5 includes a new `SEARCH_PRELOAD` feature. This is a space-separated list of list names for which you want to preload all the files that might be necessary for the fastest possible search experience. It is set in the site configuration file in the usual way:

```
SEARCH_PRELOAD=BIGLIST HUGELIST ANOTHERLIST
```

Alternately, it can be set in the site configuration section of the web interface.

The LISTSERV 16.5 enhanced search engine is very fast when the data is already in memory. Without `SEARCH_PRELOAD`, if your search encompasses 1,000 text files and a 100MB compressed index, there will be a very noticeable startup time, at least on conventional drives, every time the search is run. Naturally, running LISTSERV on expensive server-class SSD drives would speed the process up, but by enabling `SEARCH_PRELOAD`, the average customer gets a \$0.00 solution to the problem, and can put all their “free” RAM to good use instead of leaving it unused.

Certain customers have one main list for which they may want to provide the best possible search performance, faster than even search engines. With `SEARCH_PRELOAD`, LISTSERV will load all the relevant files at startup (in the background) and keep them loaded for the fastest possible access.

Performance, Usability: Faster archive searching



Note: *This improvement does not apply to LISTSERV Lite, which does not have the archive search interface.*

z/VM availability: *The ORDER BY feature described below is **not** available for LISTSERV servers running on z/VM. The performance improvements and the other two features described **are** available on z/VM.*

In addition to the new `SEARCH_PRELOAD` feature, LISTSERV 16.5 includes significant performance improvements for standard (non-preloaded) archive searches, which are a potential bottleneck for the web interface. This is particularly true on servers with large numbers of lists that in turn have many years' worth of large archive notebooks. LISTSERV HPO servers will see the greatest performance increases, as the changes address long-standing issues with how HPO and WA interact during a web archive search.

The changes have been thoroughly tested, and performance has significantly improved for both Classic and HPO – indeed, **HPO is now 30 times faster than Classic for large list searches.**

There are three new features for the `SEARCH` command which constitute the customer-facing portion of the changes. The first new feature is `BRIEF`, and is specified prior to any other search options or search terms:

SEARCH [BRIEF] ...

This feature suppresses the long report showing where the search string was found in context. Note that this has no effect unless the search string contains actual search words (not just a wildcard followed by a date or author in a "where" clause). For instance,

```
s * in mylist-1 where sender is joe@example.com
```

and

```
s brief * in mylist-1 where sender is joe@example.com
```

will both show a brief output without the more-verbose contextual report, whereas specifying at least one search word, e.g.,

```
s multifaceted in mylist-1 where sender is joe@example.com
```

and

```
s brief multifaceted in mylist-1 where sender is joe@example.com
```

will result in the first response containing the contextual report showing a few lines where the search term(s) appear in each posting followed by the list of postings that march the search criteria, and the second response containing only the list of matching postings.

For the record, the contextual reports look like this:

```
>>> Item #6622 (17 Dec 1994 10:50) - Re: Too Much Data?
The problem you have is that DOS is slowly but surely going to disappear.
While it will be possible to install NT and/or WIN95 over an existing DOS
      ^^^^^^^
installation, and dual boot them, you're going to find that DOS apps are
^^^^^^^
going to be increasingly rare. Thus the move to things like FoxPro for
```

and are printed before the index list of matching postings.

The second new feature is an ORDER BY clause on the SEARCH command:

```
SEARCH ... ORDER [BY] field1 [ASCending|DESCending] [field2
[ASCending|DESCending]...]
```

The supported universal fields are # (item number) and DATE; these fields are used as-is for sorting, without any pre-processing. The supported list archive fields are SENDER and SUBJECT; these fields have any escape sequences decoded before sorting, and are sorted in a case-insensitive manner. The 'Re:' prefix is ignored when sorting the subject.

Note that sorting a non-Unicode index will not always group like subjects together if they were encoded with different character sets. L-Soft recommends reindexing any lists with legacy indexes to enable Unicode support.



IMPORTANT: *If you order by a single non-unique field, the order of items with identical sort fields is unpredictable. It is suggested that such searches should end with a sort on #, which is guaranteed to be unique. This also ensures that you don't miss any messages when going to the next page (i.e., in a web archive search).*

Finally, to make the ORDER BY feature useful to WA or to any other application that might send LISTSERV a search command, there needed to be a way to tell LISTSERV to skip the first X matches. The record selector syntax, 'SEARCH ... IN LISTNAME.xxx-yyy', refers to item numbers, not search results; WA can safely use it to go to the next page when using the default search order, but not otherwise. Therefore a SKIP option was added to tell LISTSERV to skip the first nnnn number of results:

```
SEARCH ... SKIP nnnn
```

For instance,

```
s antidisestablishmentarianism in mylist-1 order by subject date skip 100
```

will skip the first 100 matches and start output at the 101st match.

Usability: Archive search precedence improvement

LISTSERV 16.5 include a search improvement which increases the precedence of NEAR over AND in archive searches. This solves a problem where a search like "pecan pie and whipped cream" was not always evaluated as "(pecan pie) and (whipped cream)."

Security, Anti-Spam: TRAPIN now applied to X-CONFIRM command

In early April 2018, a pesky bot began to exploit the list subscription form in the web interface and creating backscatter spam by placing a phishing URL in the "Name:" field and a random email address in the "Email:" field. Because the request was considered "anonymous" by LISTSERV at this point, it caused LISTSERV to generate an X-CONFIRM message to the targeted email address requesting an "OK" response, resulting in the X-CONFIRM message containing the phishing URL. For instance:

```
SUBSCRIBE TEST http://example.com/badscrip.php
```

L-Soft has responded to this by having the X-CONFIRM code check the TRAPIN site configuration variable prior to sending out the OK confirmation request, to see if the address entered in the "Email:" field matches any address or address pattern found in that setting. Because the bot was observed to generate many subscription requests for users at the same domain, setting a TRAPIN pattern such as "*@MAIL.RU" (which was one of the actual domains being targeted) would result in the following if an address such as "test@mail.ru" were used:

```
7 May 2018 15:25:00 From [ANONYMOUS]: QUERY ***GUI*** ***DEFSUB*** TEST
7 May 2018 15:25:00 From [ANONYMOUS]: LIST AVAIL
7 May 2018 15:25:37 From [ANONYMOUS]: X-LOGCK - ORGINFO(fe80::c837:79e7:7c69:4f8e%3)
WM: X-CONFIRM ***GUI*** test@mail (...)
7 May 2018 15:25:37 To [ANONYMOUS]: Rejected - access disabled by administrator
(TRAPIN).
7 May 2018 15:25:37 From [ANONYMOUS]: QUERY ***GUI*** ***DEFSUB*** TEST
7 May 2018 15:25:37 From [ANONYMOUS]: LIST AVAIL
```

It should be noted that backscatter messages can end up reported in the list's daily error monitoring report (DEMR). One such example we noted was as follows (with some obfuscation for obvious reasons):

```
1 04/09 04/09 "У вас не израсходованный баланс 6834p. http://baddomain.tld/Vasha-
vyplata-oformlena-04-08-19" <USER83@INBOX.RU>
Last error: 5.0.0 550 spam message rejected. Please visit
http://help.mail.ru/notspam-support/id?c=wWWm_ixxGu
0bL505NZY_R7AMvUmUtFOe2J2kPCS69ikQAAAAA9QBAPPErSY~
or report details to abuse@corp.mail.ru. Error
```



```
code:
FEA665C1ED1A712C399D2F1B473F963549BD0CB09E53B4943CA
49DD829F6BA24. ID: 000000100001D40326ADC4F3.
```

This does not mean that the address was successfully subscribed to the list, only that the confirmation request bounced and was handled normally by LISTSERV's error handling suite.

This change is related to but distinct from the change to WA's handling of anonymous subscription requests documented below.

GDPR, Usability: GET (COLUMNS command enhancement

In order to make it easier to search changelog files for specific email addresses (primarily for the purpose of EU General Data Protection Regulation (GDPR) reporting), an enhancement to the existing GET (COLUMNS command switch was added. The original syntax could be used to search for text in specified columns of the changelog and report back, e.g.,

```
GET MYLIST-L CHANGELOG (MSG COLUMNS(W1 20170601-)
```

which would show all records in MYLIST-L CHANGELOG from June 1, 2017, to the present. The MSG switch tells LISTSERV to send the results back "on the wire" rather than in an email message.

LISTSERV 16.5 expands the COLUMNS functionality by adding a comparison sub-option (in addition to all of the existing comparison sub-options) which allows searches using an optional data type followed by a pattern, as follows:

```
[^]=typepattern
```

Where:

- '^' is an optional negation
- Type is a mandatory single character indicating the type of comparison
- Pattern is the text to match the requested column to; as usual, a match will select the line for inclusion, and no match will try the next column/pattern pair, if any
- No spaces are allowed in the operand

Type can be:

- '=': column must equal pattern (this is not a substring match)
- 'W': pattern must appear in column as a space-delimited word
- '*': wildcard matching of column to pattern

The '=' option is arguably redundant since COLUMNS(16- ==ADD) does the same thing as COLUMNS(16- ADD), but it can come in handy if you want to match to a pattern containing a hyphen, or starting with an equal sign.

In a GDPR context, you might do:

```
GET listname.CHANGELOG (MSG COL(16- =Wemailaddr)
```

to pull all records out of the listname.CHANGELOG file that contain the space-delimited email address (represented by *emailaddr* in the example) anywhere from column 16 of the record to the end of the record. For an email address such as joe@example.com, this might include records such as

```
20170329114508 POST joe@EXAMPLE.COM This is the subject of my email
```



For GDPR requests, it is important to note that this type of search might return records that you do not want to show to a GDPR requestor because they contain someone else's private information, or other confidential information.

GDPR, Usability: SYSTEM and NOLIST-* changelogs exposed in LISTSERV FILELIST

As part of the general GDPR readiness revamp, LISTSERV's SYSTEM and NOLIST-* changelog files are now exposed in LISTSERV FILELIST (the contents of which can be returned with an INDEX command without further qualification, or with a GET LISTSERV FILELIST command). This makes it possible for an automated process to determine which system-level or DISTRIBUTE changelogs (generally, the latter will be Maestro-related) are available for GDPR processing.



Note that, while the *existence* of the files is exposed in the filelist output, the *content* of the files themselves is not available to the public. Their GET and PUT FACs are set either to CTL or DST automatically. Only LISTSERV maintainers (those email addresses listed in the POSTMASTER= site configuration setting) have the authority to GET or PUT a file set to CTL, whereas DST is restricted to DIST_ALLOWED_USERS.

Security: PW REP expiration message exposed replacement password

When a user sends a PW REP command to LISTSERV to change their password, LISTSERV requests an OK confirmation, which eventually times out unless approved. When the request times out, LISTSERV sends a notice to the user indicating that the request has expired. As with all such requests, LISTSERV includes the command that was sent for reference, but in the case of PW REP, did not redact the replacement password from the command included with the expiration notice. This has been fixed, and the replacement password is now redacted from the message.

Usability: Apply sub-list REPRO setting to "Sender:" address

A rare case arose as follows:

- a superlist with a sub-list
- an email subscribed to that sub-list and set to REPRO
- a message sent to the superlist with
 - the Sender: header set to the email that was subscribed to the sub-list, and
 - the From: header set to an email address that was not subscribed to any of the lists involved.

The issue was that the mail went through, but the email that was subscribed to the sub-list with REPRO did not receive a copy of the post because it was not the address LISTSERV found in the From: header.

While we believe this will continue to be a rare case, LISTSERV was changed to apply the sub-list REPRO setting to any address used in a Sender: header that is subscribed to a sub-list, even if that address is not the address found in the From: header.

Usability: Time specification now allowed for weekly/monthly digests

LISTSERV 16.5 now allows a weekly or monthly digest to be cut at a specified hour. The previous behavior allowed such a setting only for daily digests. Weekly digests could be set to go out on a particular day of the week, and monthly digests could be set to go out on a particular day of the month, but it was not possible to control what time of day the weekly or monthly digests were cut. Such digests were simply cut at midnight (server time).

The new behavior is enabled by setting the hour as the final parameter of the Digest= keyword setting. For instance,

```
Digest= Yes, /home/listserv/lists/mylist-1, Weekly, Tuesday, 10
```

```
Digest= Yes, c:\listserv\lists\mylist-1, Monthly, 15, 19
```

In the first case, the weekly digest will be cut on Tuesday at 10:00 AM. In the second case, the monthly digest will be cut on the 15th of the month at 7:00 PM.

As always, you may at your option specify the minutes (e.g., "10:00", "19:00") but also as always, digests are cut on the hour, regardless of whether you have specified ":mm" in your setting.

LISTSERV 16.5 Changes for WA

The current WA version at the version 16.5 release is "16.5 (Build Date: 25 Apr 2018)".

WA continues to undergo improvements and optimizations. L-Soft also continues to quash XSS and similar vulnerabilities as they are reported. Because these issues remain exploitable in older versions of WA, L-Soft will not discuss the specific changes made in this regard, other than to state that all those vulnerabilities that have been found during the course of a comprehensive code review have been fixed in the current release of WA.

Performance: [HPO] Introducing HPO mode for WA

Starting with version 16.5, customers with a LISTSERV HPO license (High Performance Option) will experience performance improvements in WA as well as in LISTSERV itself, most notably when accessing lists with large archives or supporting a large number of concurrent web users. Like LISTSERV, WA will use more RAM when running in HPO mode, which can add up if there are hundreds of concurrent WA processes. You may want to review RAM allocations if you run LISTSERV in a virtual machine.

Note that there is no separate license for WA's HPO mode. It is included at no extra charge with every LISTSERV HPO license, and is enabled automatically.

Usability: [Windows] WA.SYM debugging symbols for WA.EXE

This file was added in the LISTSERV for Windows 16.0-2017a level set release in February, 2017. However, unless the option was chosen during the upgrade to also upgrade the web interface, the WA.SYM file may not have been installed.



Note: It is important that this file *always* be installed along with WA.EXE, because it is required to be present in order for WA to produce meaningful error reports under Microsoft Windows.

The installer will place the file in the CGI/scripts directory with WA.EXE if the option to install or upgrade the web interface components is chosen.

If you have been given an out-of-band WA.EXE file to fix a problem, it should come with a WA.SYM file specific to that WA.EXE build, and both files should be installed in that case.

If WA.EXE crashes, files with names similar to WA-20170209-155115-007.CRASH will be written into the web interface's UPLOAD directory. These files can be then used by L-Soft Support to assist in diagnosing the crash.

Security, Anti-Spam: Reject Subscription Requests Containing URLs

Beginning with this release of WA, any anonymous subscription request (that is, a subscription request from a non-logged-in user) which contains the string "://" is rejected. This effectively prevents a spammer or 'bot from using the subscription interface to generate an "OK" confirmation request by inserting a random email address in the email field and a URL pointing to a malicious website in the "real name" field.

This improvement is related to but distinct from the change to X-CONFIRM documented above.

Security: New configuration variable to set HSTS maximum age

A new LISTSERV site configuration variable, `WWW_HSTS_MAX_AGE`, when set to a non-zero value, causes WA to output the header "Strict-Transport-Security: max-age=x", where "x" is the non-zero variable setting.

This is intended primarily to address the US DHS directive BOD 18-01, but will enhance security for any LISTSERV site which uses the HTTPS protocol in the LISTSERV web interface.



Note: WA functions named `DEBUG-*` (e.g., `DEBUG-SHOW-VERSION`) may not load the configuration file, and therefore might not output the Strict-Transport-Security: header.

Also note that if you already have the webserver configured for HSTS, the webserver HSTS value may override the setting provided by WA.

With HSTS enabled site-wide in IIS, you may find that:

- The "archives" directory no longer works correctly if it is implemented as an IIS virtual directory
- Some images served by WA.EXE may be broken

`WWW_HSTS_MAX_AGE` can be used to solve this problem rather than enabling site-wide HSTS in IIS.

LISTSERV 16.5 Changes for Web Templates

Other than minor editorial changes, the only significant change to the web templates since 16.0-2017a is:

GDPR: New privacy policy web templates available

Three new templates have been added to the default web templates to provide the capability of adding privacy policy information to LISTSERV. The three templates include

PRIVACY	The calling template
PRIVACY-POLICY	The HTML text of your localized privacy policy
LAYOUT-GLOBAL-PRIVACY	The general layout of all privacy policy pages on the server

PRIVACY-POLICY is the template most list owners and administrators will be concerned with, as it is the place where the privacy policy HTML text is stored. It contains only comments by default:

+* Enter the privacy policy for your list in this template in HTML format.
+* A link to this policy will automatically be placed on the list home page.

As noted, this is an HTML template, quickly-accessible via the Web Template Editor's "Frequently Modified Templates" category, which can be provided at either the site level or at the individual list level.



Note: If a policy is provided at both the site level and the list level, the list level policy will override the site level policy.

By default, the home page of each list will contain a link to the privacy policy if one has been provided. List owners may, at their option, modify the \$SIGNUP mail template to include a link to the privacy policy page. The basic URL pointing to the privacy policy is of the form

Windows:	<code>http://server-fqdn/scripts/wa.exe?SHOWTPL=PRIVACY&L=listname</code>
Unix:	<code>http://server-fqdn/scripts/wa?SHOWTPL=PRIVACY&L=listname</code>

where *server-fqdn* is the fully-qualified LISTSERV hostname and *listname* is the name of the list for which the privacy policy is to be viewed.

If there is no explicit privacy policy template defined at either the site or list level, the page still displays, but other than the header, it will be blank.

Installing LISTSERV 16.5

IMPORTANT: Install (or ensure that it is installed) your LISTSERV 16.5 product LAK before upgrading! A valid product LAK (License Activation Key) with "REL=16.5" must be installed before upgrading or LISTSERV will not start after the upgrade. Also, **a valid maintenance LAK expiring no earlier than 7 May 2018 is required** in order to apply this release.

If you have not received a LISTSERV 16.5 product LAK, please contact your sales representative or SALES@LSOFT.COM before upgrading!

To find out if you can upgrade to LISTSERV 16.5 with your current license key, please issue a SHOW LICENSE command to LISTSERV and examine the response. It will be similar to this:

```
License type:      Permanent
Expiration date:  None - perpetual license
Maintenance until: 26 Nov 2018, serial number MNT-XYZ-1
Capacity:         Unlimited
Version:          16.5
Serial number:    XYZ-1
Build date:       7 May 2018
```

Your license key will be valid for the 16.5 upgrade if your current product LAK is for version 16.0 or higher, and your maintenance LAK is valid until at least 7 May 2018.

Sites running LISTSERV 14.4 or later may use the LAK input tool in the Web Administration interface to apply and check their new LAK before upgrading.

The installation kits found on L-Soft's web site can be used either to install a new copy of LISTSERV or to upgrade an existing installation. To download LISTSERV 16.5, simply go to L-Soft's Web site and download an evaluation copy of LISTSERV or LISTSERV Lite, then follow the installation instructions for your operating system. The kits can be found at:

<http://www.Isoft.com/download/listserv.asp>

<http://www.Isoft.com/download/listservlite.asp>

Current Supported Operating Systems

LISTSERV for Microsoft Windows	Windows 7 (64-bit only)
	Windows 8.x (64-bit only)
	Windows 10 (64-bit only)
	Windows Server 2008 R2
	Windows Server 2012
	Windows Server 2012 R2
LISTSERV for Unix	Windows Server 2016
	AIX 4.3 (PowerPC) and later
	Linux 2.6 and later (64-bit)
	Linux 2.6 and later (RH6 64-bit)
	Linux 2.4.2 or later (S/390)
LISTSERV for MacOS	Solaris 10 and later (SPARC)
	Solaris 10 and later (x64)
LISTSERV for MacOS	MacOS X (x86)
LISTSERV for VM/ESA, z/VM	

Support for 32-bit Windows and Linux was withdrawn as of the release of LISTSERV 16.0-2017a.

Support was withdrawn for OpenVMS as of the release of LISTSERV 16.0-2017a.

SPECIAL NOTES

Upgrade Instructions

This document does not include upgrade instructions. Please see the installation guide specific to your OS platform for upgrade instructions. Installation guides are available at <http://www.lsoft.com/resources/manuals.asp#2> .

Supported Operating Systems

LISTSERV version 16.5 is available only for operating systems currently supported by L-Soft. L-Soft no longer has development systems for unsupported operating systems and is not in a position to compile LISTSERV 16.5 for those systems.

Support for GDPR Compliance

L-Soft has made available, at no charge, a comprehensive EU General Data Protection Regulation (GDPR) reporting script written in the Microsoft PowerShell scripting language, which can be used in conjunction with L-Soft's LCMD or LCMDX utilities on Microsoft Windows, Linux, and MacOS systems. For more information, please see our *GDPRSCAN Installation and Operating Guide for Windows, Linux, and MacOS* at http://www.lsoft.com/manuals/16.5/GDPRSCAN_Installation_and_Operating_Guide.html.

end of file