v m s Software

Samba for OpenVMS

Version 4.10-16A for OpenVMS I64 and Alpha servers, based on Samba 4.10.16
Release Notes
February 2021

Introduction

VMS Software Inc. (VSI) is pleased to provide you with a new VSI-supported version of Samba for OpenVMS. This release of Samba is based on Samba 4.10.16 and represents a significant update from OpenVMS Samba previous V4.6-5 versions.

For details of new features, enhancements, and known issues please review the release notes found at the following links. Please be aware that some product features may be platform-specific.

*   https://www.samba.org/samba/history/samba-4.10.0.html

*   https://www.samba.org/samba/history/samba-4.10.16.html

Samba Documentation

For the latest information about Samba, see https://www.samba.org/samba/docs/. Please be aware that there may be some differences from this documentation that are specific to Samba on OpenVMS; any such known differences are identified elsewhere in this document.

New and changed features

This release of Samba for VSI OpenVMS is based on Samba 4.10.16. Details of new and changed features may be found at https://www.samba.org/samba/history/samba-4.10.0.html. Additionally, changes in Samba 4.10.16 have necessitated a number of changes in terms of the operation and configuration of Samba for VSI OpenVMS:

*   The WINBIND process is now required to be running on domain member servers (it remains optional on standalone servers). If the WINBIND process is not running, domain users cannot be authenticated. As a consequence of this change, it has been necessary to make several changes to the SAMBA$CONFIG menu and the underlying commands that are performed when configuring the Samba environment.

    If a previous version of Samba is installed and Samba is configured as a domain member server, determine if the WINBIND process is currently being prevented from starting by checking for the existence of the following command in `SAMBA$ROOT:[BIN]SAMBA$SETTINGS.COM`:

    `$ DEFINE/NOLOG WINBINDD_DONT_ENV 1`

If the above command is present in `SAMBA$SETTINGS.COM`, it must be removed or commented out.

Additionally, check for the existence of the `WINBINDD_DONT_ENV` logical name and deassign all definitions (this logical name may be defined in the process and/or system logical name tables) before attempting to start Samba for OpenVMS v4.10-16:

`$ SHOW LOGICAL WINBINDD_DONT_ENV`

If the logical name is defined in the process logical name table, deassign it with the command:

`$ DEASSIGN WINBINDD_DONT_ENV`

If the logical name is defined in the system logical name table:

a) Deassign it with the command:

   `$  DEASSIGN/SYSTEM WINBINDD_DONT_ENV`

b) Locate the command that defines the `WINBINDD_DONT_ENV` logical name in the system startup procedures, such as `SYS$MANAGER:SYLOGICALS.COM`, and remove it.

- Due to limitations of the old LDAP libraries provided by OpenVMS, it has been necessary to build the new version of Samba using the updated LDAP client API provided by OpenLDAP for VSI OpenVMS. The OpenLDAP client libraries are statically linked with Samba, and it is therefore not necessary to install OpenLDAP for VSI OpenVMS before installing Samba.

A list of significant issues from the previous release of Samba for VSI OpenVMS fixed in this release is provided elsewhere in this document.

Requirements

Samba 4.10-16A for VSI OpenVMS servers requires the operating system and layered product software versions listed below.

- VSI OpenVMS Version 8.4-1H1 or higher

- VSI TCP/IP Services for OpenVMS, HPE TCP/IP Services for OpenVMS, or MultiNet TCP/IP

- The software must be installed and used on an ODS-5-enabled file system; the software cannot be installed on an ODS-2 file system and ODS-2 file systems cannot be used for file shares

- The OpenVMS internationalization data kit (VMSI18N) must be installed in order for Samba to be able to correctly support international characters in file names

The reader should be familiar with the installation, configuration, and use of open source products such as Samba in the OpenVMS environment.

Recommended reading

Before using Samba 4.10-16A on OpenVMS, VSI recommends that users read the documentation available at https://www.samba.org/samba/ in order to better understand how to configure and manage the software.

Before installing the kit

1. If necessary, increase the following `SYSGEN` parameters to at least the values specified below, run `AUTOGEN`, and reboot the system in order to pick up the new values:

Parameter	Minimum value required	Add to `MODPARAMS.DAT`
PROCSECTCNT	512	MIN_PROCSECTCNT = 512
CHANNELCNT	2560	MIN_CHANNELCNT = 2560

2. If you are upgrading or migrating from the old CIFS for OpenVMS to Samba for VSI OpenVMS, VSI strongly recommends that you back up the existing CIFS `SAMBA$ROOT` directory tree prior to installing Samba.

Important cluster considerations

Running the old CIFS for OpenVMS and Samba for VSI OpenVMS in the same cluster is not supported. Existing CIFS for OpenVMS configurations will be migrated to Samba for VSI OpenVMS equivalents when `SAMBA$CONFIG.COM` is executed.

Samba supports running multiple instances of Samba in a cluster. An instance is defined by the `SAMBA$ROOT:` directory tree. If there are two separate `SAMBA$ROOT:` directory trees, the cluster contains two instances of Samba. Cluster members that share the same `SAMBA$ROOT:` directory tree form a Samba cluster.

Each Samba instance requires a unique Samba cluster alias name. For example, in a cluster with three nodes:

- All nodes can be members of a single Samba cluster instance
- Two nodes can be members of a Samba cluster instance while the third node runs a separate, standalone instance of Samba
- All three nodes can run separate standalone instances of Samba

WARNING: No two instances of Samba should allow access to the same share directories and files in a cluster because this can lead to data corruption. Separate instances of Samba do not share file locking details with other instances.

Installing the kit

The Samba kit is provided as a compressed OpenVMS PCSI kit (`VSI-I64VMS-SAMBA-V0410-16A-1.PCSI$COMPRESSED` for Integrity or `VSI-AXPVMS-SAMBA-V0410-16A-1.PCSI$COMPRESSED` for Alpha) that can be installed by a suitably privileged user using the following command:

```
$ PRODUCT INSTALL SAMBA
```

Note: If there exists on the system a previous version of Samba for VSI OpenVMS that was installed using the `/DESTINATION` qualifier to specify a non-default location for the installation, the same non-default location must be specified using the `/DESTINATION` qualifier when installing this new version of Samba.

By default, Samba will be installed to `SYS$SYSDEVICE:[VMS$COMMON.SAMBA]`. The PCSI `/DESTINATION` qualifier may be used to install the software into an alternative location.

If multiple cluster members will share the same `SAMBA$ROOT:` directory tree, Samba must be installed to a device that is mounted by all such cluster members.

To allow mixed architecture cluster nodes (Alpha and Integrity servers) to be members of the same Samba instance, install Samba on a node of each architecture type but specify the same installation location using the `/DESTINATION` qualifier.

If you plan to run Samba on multiple cluster members of the same architecture which do not have a common System disk, install Samba on one such cluster member only. Then follow the post-installation instructions in this document to complete installation on other cluster members.

The installation directory of Samba for OpenVMS is associated with the `SAMBA$ROOT` logical name. `SAMBA$ROOT` is a rooted logical name that defines the root location for the Samba configuration files, logs, and other product files. Samba configuration files are stored in `SAMBA$ROOT:[LIB]` and, by default, logs are stored in `SAMBA$ROOT:[VAR]`. The `SAMBA$ROOT` logical name is defined in `SYS$STARTUP:SAMBA$DEFINE_ROOT.COM` (which is executed from Samba startup procedure, `SYS$STARTUP:SAMBA$STARTUP.COM`).

The installation will then proceed as follows (output may differ slightly from that shown depending on platform and other factors):

```
Performing product kit validation of signed kits ...

The following product has been selected:
    VSI I64VMS SAMBA V4.10-16A              Layered Product

Do you want to continue? [YES]

Configuration phase starting ...

You will be asked to choose options, if any, for each selected product and
for any products that may be installed to satisfy software dependency
requirements.

Configuring VSI I64VMS SAMBA V4.10-16A: VSI OpenVMS SAMBA

    © (c) Copyright 2021 VMS Software, Inc.

    OpenVMS SAMBA is released under the terms of GNU Public License.

    This installation procedure requires that all the following
    conditions be satisfied:

    1.  This procedure is running on an Itanium processor.
    2.  The system is running OpenVMS V8.4-1H1 or later.
    3.  All required privileges are currently enabled.
    4.  No CIFS or SAMBA images are running on this node or anywhere in
        the cluster that make use of common samba$root installation
        directory.
    5.  ODS5 filesystem only.
    6.  SYSGEN Parameter values:

        Parameter       Minimum Required
        ----------------  ----------------------
        CHANNELCNT                 2560
```

```
           PROCSECTCNT                512

Do you want to continue? [Please, type N or NO if you do not want
to continue, any other answer means YES]

* This product does not have any configuration options.

Execution phase starting ...

The following product will be installed to destination:
    VSI I64VMS SAMBA V4.10-16A            DISK$I64V842L1SYS:[VMS$COMMON.]

Portion done: 0%...10%...20%...30%...40%...50%...60%...70%...80%.  90%

    User Accounts and User Identification Codes (UICs)
    ------------------------------------------------------------------

    The OpenVMS SAMBA V4.10-16A installation creates five OpenVMS
    accounts: SAMBA$SMBD, SAMBA$NMBD, SAMBA$GUEST, SAMBA$TMPLT
    and SMBADMIN. The default UIC group number for these new
    accounts depends on the following:

    o If you are installing the server for the first time, the
      default is the first unused UIC group number, starting
      with 360.

    o If any of these accounts already exists, then the
      default UIC group number will not be used to change the
      UIC of any existing accounts.

    For more information about UIC group numbers, see the
    OpenVMS System Manager's Manual.

Enter default UIC group number for SAMBA accounts
Group: [360]
Creating OpenVMS accounts required by SAMBA
Created account SAMBA$SMBD
Created account SAMBA$NMBD
Created account SAMBA$GUEST
Created account SAMBA$TMPLT
Created account SMBADMIN
SAMBA$ROOT is defined as "$1$DGA60:[SYS0.SYSCOMMON.SAMBA.]"
Setting file protections...
  File protections are set
Creating Samba for OpenVMS root definition file
SYS$COMMON:[SYS$STARTUP]SAMBA$DEFINE_ROOT.COM...
  File created
Save startup files
Setup SAMBA logical environment
Successfully finished

    In a cluster, on all the nodes that are going to use common
    samba$root installation directory as the current node, copy
    the following files to SYS$STARTUP directory of each node:

        SYS$STARTUP:SAMBA$STARTUP.COM
        SYS$STARTUP:SAMBA$SHUTDOWN.COM
        SYS$STARTUP:SAMBA$DEFINE_ROOT.COM

    To automatically start OpenVMS SAMBA during system startup
    add the following line to the file SYS$MANAGER:SYSTARTUP_VMS.COM
```

```
     after the TCPIP startup command procedure:

          $ @SYS$STARTUP:SAMBA$STARTUP.COM

     To shut down OpenVMS SAMBA during system shutdown
     add the following line to the file SYS$MANAGER:SYSHUTDWN.COM:

          $ @SYS$STARTUP:SAMBA$SHUTDOWN.COM

Press Enter to continue:

     To Configure OpenVMS SAMBA on any of the nodes in OpenVMS
     cluster that will share the common samba$root installation
     directory as the current node, execute:

          $ @SYS$STARTUP:SAMBA$DEFINE_ROOT.COM
          $ @SAMBA$ROOT:[BIN]SAMBA$CONFIG.COM

     Define symbols for all Samba utilities:

...100% $ @SAMBA$ROOT:[BIN]SAMBA$DEFINE_COMMANDS.COM

The following product has been installed:
     VSI I64VMS SAMBA V4.10-16A          Layered Product
```

Post-installation steps

After you have successfully installed Samba, follow these steps to configure it:

1. Verify that the `SAMBA$ROOT` logical name is set:

    `$ SHOW LOGICAL SAMBA$ROOT`

    `"SAMBA$ROOT" = "$1$DGA400:[SYS1.SYSCOMMON.SAMBA.]"`

    If the logical name is not defined, execute the following command:

    `$ @SYS$STARTUP:SAMBA$DEFINE_ROOT`

2. Define symbols for all Samba utilities:

    `$ @SAMBA$ROOT:[BIN]SAMBA$DEFINE_COMMANDS.COM`

3. In a cluster, copy the files in `SAMBA$ROOT:[CLUSTER]` to the `SYS$COMMON:[SYS$STARTUP]` directory on other nodes in the cluster that meet both of the following requirements:

    *   Will use the same `SAMBA$ROOT` directory tree as the installation node

    *   Do not boot from the same system disk as the installation node

4. Configure Samba server:

    Note that for this release of Samba for OpenVMS, the server must be configured as a standalone server or a domain member server.

    Run the Samba configuration utility to generate the Samba configuration files. These files are not created during installation and must be generated as described below. Note that the Samba configuration utility also migrates existing CIFS for OpenVMS configurations to Samba (see the "Migrating CIFS for OpenVMS" section of this document).

    When configuring multiple cluster members that share the same `SAMBA$ROOT:` directory tree:

- The Core and Generic menu options are shared by all members of the same Samba cluster instance; changes to these options affect all cluster members. The System Specific menu options are unique to the cluster member on which they are set.

- On the first cluster member, at the main menu, select option "A" to configure all options.

- On subsequent cluster members, at the main menu, select only option 1 – System Specific options.

```
$ SMBCONF


            SAMBA Configuration Utility

Use this utility to configure the server role, create TCP/IP
services, and configure other options.

The following conditions must be met prior to configuring SAMBA:

   1. Log into OpenVMS using a privileged system account.

   2. No SAMBA images are running on this node.

For more information about Samba Server configuration, please refer to
the Samba for OpenVMS release notes.

If Samba Server is being configured afresh on this node, choose
the following option from OpenVMS Samba Main Configuration
Options Menu:

     A - Configure options 1 - 3

Press Enter to continue:

Checking for existing SAMBA Server configuration...



      OpenVMS Samba Main Configuration Options Menu

Configuration options:

        1  - System specific setup
        2  - Generic options
        3  - Core environment
        A  - Configure options 1 - 3
      [E] - Exit Menu

Enter configuration option:
```

Configure Samba as appropriate using the menu options provided. At a minimum (to generate a minimal basic configuration) you can select the "A" option and accept the defaults as you step through each sub-menu.

The TESTPARM utility should be used to review the final configuration.

For additional information about configuring Samba please refer to the documentation available at https://www.samba.org/samba/.

5. Supported Character Sets

    Samba supports all character sets supported by the OpenVMS internationalization data kit (VMSI18N), including ISO8859: ISO8859-1, ISO8859-2, ISO8859-5, ISO8859-7, ISO8859-8, ISO8859-9, ISO8859-15. The default character set is UTF-8. To configure Samba to use an alternative character set such as ISO8859-1:

    a) Add the following line to the `[global]` section of `SAMBA$ROOT:[LIB]SMB.CONF`:

        ```
        unix charset = iso8859-1
        ```

    b) Add the following line to `SAMBA$ROOT:[BIN]SAMBA$SETTINGS.COM`:

        ```
        $ DEASSIGN/NOLOG DECC$FILENAME_ENCODING_UTF8
        ```

6. Start the Samba server:

    ```
    $ SMBSTART

    Creating NMBD Process...
    %RUN-S-PROC_ID, identification of created process is 000143A2

    Enabling SMBD services...

      Successfully enabled TCPIP SMBD services
    ```

7. Add the following command to the system startup procedure to start Samba when the system is booted. Note that this command must be added after the command that starts TCP/IP.

    ```
    $ @SYS$STARTUP:SAMBA$STARTUP.COM
    ```

8. Add the following command to the system shutdown procedure to stop Samba during system shutdown:

    ```
    $ @SYS$STARTUP:SAMBA$SHUTDOWN.COM
    ```

9. Use the `SMBMANAGE` utility to manage Samba shares, users, groups, and account policies as necessary.

10. You may wish to run the following command to add a version limit on the sub-directories in `samba$root:[var.cores]` to prevent unnoticed process dump files from consuming huge amounts of disk space. You may wish to use an alternative value for the version limit as appropriate to your environment.

    ```
    $ set file/version_limit=10 samba$root:[var.cores]*.dir
    ```

11. By default the `SMBCLIENT` utility, when connected to a share on a Samba for VSI OpenVMS host, does not accept paths using OpenVMS directory syntax. To enable use of OpenVMS directory syntax support for `SMBCLIENT` add the following entry to the `[global]` section of `SAMBA$ROOT:[BIN]SMB.CONF`:

    ```
    vms path allow = yes
    ```

Migrating CIFS for OpenVMS

Run `SAMBA$CONFIG.COM` after installation to migrate an existing CIFS for OpenVMS configuration to Samba for OpenVMS. `SAMBA$CONFIG.COM` runs `SAMBA$MIGRATION.COM`, which renames the appropriate CIFS users/groups and their associated OpenVMS account and identifier names (changing `CIFS` to `SAMBA`) and updates references in `.CONF` and `USERNAME.MAP` files. `SAMBA$CONFIG.COM`

also has routines to migrate obsolete, deprecated, and modified parameters to their equivalents in Samba.

The Samba migration process saves the old CIFS for OpenVMS configuration files to a subdirectory in `SAMBA$ROOT:[BACKUP_MIGRATION]` and produces a log file named according to the format `SAMBA$ROOT:[VAR]SAMBA$MIGRATION_<date>_<time>.LOG`.

To undo or roll back the migration if problems were encountered with the migration, proceed as follows:

- Run the migration procedure (`SAMBA$MIGRATION.COM`) and specify the path where the backup migration files are stored. In the following example, the backed up CIFS configuration files are stored in the directory `SAMBA$ROOT:[BACKUP_MIGRATION.30APR2019_103511]`:

  ```
  $ @SAMBA$ROOT:[BIN]SAMBA$MIGRATION -
  _$ SAMBA$ROOT:[BACKUP_MIGRATION.30APR2019_103511]
  ```

- Reinstall CIFS for OpenVMS

WINBIND support

`WINBIND` is a component of Samba that provides user and group identity mapping. In Samba for OpenVMS, `WINBIND` is a separate daemon process with one to four subprocesses. (Note: in CIFS for OpenVMS there is no separate `WINBIND` process; the `WINBIND` functionality is instead included in each SMBD process.)

When deciding whether to enable or disable `WINBIND`, consider the following points:

- If Samba is configured in a standalone server role, `WINBIND` provides no useful purpose and would typically be disabled.

- If Samba is configured as a member server, `WINBIND` is required and provides three main functions:

  a) `WINBIND` is used when authenticating domain users (including trusted domain users). If `WINBIND` is not running, domain users cannot be authenticated and will fail to connect.

  b) Ability to reference domain user and group accounts when managing Samba. For example, to add or remove domain user and group accounts as members of local Samba groups.

  c) Ability to dynamically create OpenVMS user accounts. Each Samba user requires an OpenVMS account with a unique UIC in order to be distinguished from other users for security purposes. Administrators may create the necessary OpenVMS accounts manually or allow `WINBIND` to create OpenVMS accounts for Samba users dynamically, as they are needed. When properly configured, `WINBIND` creates a new OpenVMS account with a unique UIC for any domain user who successfully authenticates but does not already have an OpenVMS account mapped to their Windows account.

Configuring WINBIND

Before configuring `WINBIND` on a member server, determine what functionality is desired. If configured to dynamically create OpenVMS user accounts `WINBIND` requires a range of values (specified in decimal) to allocate as user identifiers and from which a unique OpenVMS username and UIC are derived.

The range is specified in decimal using the format "low-value - high-value", where low-value must be less than high-value. Due to OpenVMS UIC number restrictions, the low-value should not be less than 256 and the high-value cannot exceed 16382.

If WINBIND will be configured to prevent dynamic creation of OpenVMS accounts for Samba users, the range can be a minimum of 1; for example, 10000 - 10001.

If `WINBIND` will be configured to allow dynamic creation of OpenVMS accounts for Samba users, the range specified should be large enough to accommodate the expected number of user account that could possibly be created. However, the high value of the range can be increased at any time to provide additional user identifiers if necessary.

For example, a UIC number range of 5000 - 6000 allows `WINBIND` to create a total of 1001 OpenVMS user accounts.

When creating an OpenVMS user account, the ID number allocated by `WINBIND` is converted to hexadecimal and appended to the string "`SAMBA$`" to derive the OpenVMS username. The allocated `WINBIND` ID number is also converted to octal and used as the UIC group and member number assigned to the OpenVMS account. For example, if `WINBIND` allocates ID 5200 to a new user, the OpenVMS account name will be `SAMBA$1450` and its UIC will be `[12120,12120]`.

Configure `WINBIND` using the `SAMBA$CONFIG` utility. From the Main Configuration Options menu, option "`3 - Core environment`" includes the following `WINBIND` configuration option:

```
4. Enable WINBIND:
```

- To disable all `WINBIND` functionality, set option 4 to `no`. This will prevent the `WINBIND` process from starting.

- To enable `WINBIND`, set option 4 to `yes`. Note that `WINBIND` **is required** on domain member servers. When `WINBIND` is initially enabled, `SAMBA$CONFIG` will prompt for the UIC number range. After specifying the range, `SAMBA$CONFIG` returns to the core configuration menu and displays the following `WINBIND` options:

```
4. Enable WINBIND: yes
   4A. UIC number range: 10000-10001
   4B. Allow Samba to create OpenVMS accounts on-the-fly: yes
```

- To prevent `WINBIND` from dynamically creating OpenVMS user accounts, select option 4B and specify `no`.

Samba WINBIND configuration parameters

The Samba configuration parameter "`idmap config * : range = `" specifies the UIC number range that WINBIND uses when allocating user identifiers.

Note that the `TESTPARM` utility will display the following messages when no range is specified; these messages may be ignored:

```
idmap range not specified for domain '*'
ERROR: Invalid idmap range for domain *!
```

The Samba configuration parameter "`idmap config * : read only`" controls `WINBIND` ID allocation. If set to `yes`, `WINBIND` is prevented from allocating user IDs. If set to `no` (the default), `WINBIND` may allocate user IDs.

What's missing?

This release of Samba for OpenVMS does not include the following functionality:

- The classic Primary Domain Controller (PDC) and Backup Domain Controller (BDC) roles are not supported.

- The Active Directory Domain Controller role is not supported in this release. Support for this functionality is being considered and may be provided at a later date.

- A copy of the source code for Samba on OpenVMS is not included with the installation kit; however, we will provide a copy of the code on request (email support@vmssoftware.com).

Known issues

The following list identifies currently known problems and restrictions with this release of Samba for VSI OpenVMS.

- If a file and directory have the same name and exist in the same directory, both objects are displayed as directories.

- The `WINBIND` process may become unresponsive or crash on Alpha hosts if the `SYSGEN` `PQL_DPGFLQUO` parameter value is less than 256000. `PQL_DPGFLQUO` is a dynamic `SYSGEN` parameter.

- To use Kerberos authentication (via the `-k` option) with the `SMBCLIENT` utility, perform the following steps:

    a. If necessary, disable elevated process privileges (`NOIMPERSONATE`, `NOSYSPRV`, `NOREADALL`, and `NOBYPASS`):

       ```
       $ SET PROCESS/PRIVILEGE=(NOALL,TMPMBX,NETMBX)
       ```

    b. Obtain a Kerberos ticket using your domain username:

       ```
       $ net ads kerberos kinit --user <domain-username>
       ```

    c. Optionally (and if possible) elevate process privileges (to avoid potential benign errors); for example:

       ```
       $ SET PROCESS/PRIVILEGE=ALL
       ```

    d. Run `SMBCLIENT`, specifying values for server and share names as appropriate:

       ```
       $ SMBCLIENT -k \\server\share
       ```

- The OpenVMS owner of new objects created by users with a local Samba account that is included in the `admin users` list is incorrectly set to `SAMBA$SMBD` instead of the owner of the parent directory. In many environments this issue can be avoided by adding the following line to the applicable share section in `SAMBA$ROOT:[LIB]SMB.CONF`:

    ```
    inherit owner = yes
    ```

- The "`net rpc user rename`" command is not supported by Samba for OpenVMS. This is consistent with previous CIFS for OpenVMS behaviour.

- Use of Samba for OpenVMS is not supported for Microsoft Windows XP and Windows 7 clients.

- If the Samba share parameter "`inherit owner`'" equals "`no`" and a parent directory is owned by a resource identifier, when a user creates a new file or folder Samba sets the file owner to the UIC of the user creating the file rather than the resource identifier. To retain usual OpenVMS behaviour (that is, to set the resource identifier as the file owner), add "`inherit owner = yes`" to the applicable `[share]` sections in `SMB.CONF`.

- For systems running the TCPware TCP/IP stack, Samba is not able to auto-detect the active TCPware IP interface addresses. Therefore, the list of interfaces available for use by Samba must be configured manually (see "Systems Running TCPware TCP/IP" below for additional information).

- The `SMBCLIENT` "`more`" command, which displays the contents of a file, requires the GNV for OpenVMS to be installed. The GNV kit and the associated documentation are available at https://sourceforge.net/projects/gnv/.

  After installing GNV, perform the following tasks to allow the SMBCLIENT "more" command to function correctly:

  1. Define the `GNU` logical name by running the command below (you may wish to add this command to the system startup procedure):

     ```
     $ @SYS$STARTUP:GNV$STARTUP.COM
     ```

  2. Make the GNV utilities available by running the following command, which ensures that the logical name `DCL$PATH` is correctly defined in order for users to access `more` and other GNV commands. You may wish to add this command to your `LOGIN.COM` file or to `SYS$MANAGER:SYLOGIN.COM`.

     ```
     $ @GNU:[LIB]GNV_SETUP.COM
     ```

  3. If the process has a `more` symbol defined, it must be deleted:

     ```
     $ SHOW SYMBOL MORE
     $ DELETE/SYMBOL[/GLOBAL] MORE
     ```

DNS Requirements for Samba Member Servers

A Samba member server requires a proper DNS configuration to avoid long delays or timeouts when attempting to locate domain controllers.

If the OpenVMS host is configured to use the same DNS domain as the domain Samba will join, no changes are required.

However, if Samba will be joined to a domain which is not the same DNS domain configured on the OpenVMS host, the OpenVMS host should be configured as a cache-only BIND server and use a stub zone or conditional forwarding for the domain that Samba will join. Additionally, the DNS name resolver on the OpenVMS host should be configured to use localhost as its DNS server.

Prior to joining the Samba server to a domain, customers should verify the OpenVMS host is capable of resolving DNS SRV record types for the domain. For example, use the following command to verify that Samba will join a domain named example.com:

```
$ dig SRV _kerberos._tcp.example.com +noall +answer +additional
```

The response should be a list of domain controllers.

Systems Running TCPware TCP/IP

For systems running the TCPware TCP/IP stack, Samba is not able to auto-detect the active TCPware IP interface addresses. Therefore, the list of interfaces available for use by Samba must be manually configured. To do this, initially use `SMB$CONFIG` to configure Samba as a standalone server, then edit `SAMBA$ROOT:[LIB]SMB.CONF` and add the appropriate parameters (explained below) to the `[global]` section. If desired, run `SAMBA$CONFIG` to reconfigure as a member server.

To make all active interface addresses available to Samba, list those addresses and their subnet masks in a comma or space delimited list using the "`interfaces`" global parameter. Be sure to always include the localhost address (127.0.0.1) in the list of allowed interfaces.

For example, you may add an entry similar to the following to `SMB.CONF` (setting address details as appropriate to your environment and being sure to include localhost):

```
interfaces = 10.100.10.1/24, 127.0.0.1
```

To make only a subset of active interface addresses available to Samba, include only the addresses (and their subnet) of the allowed interfaces in the interfaces list and also specify "`bind interfaces only = yes`".

For example, if the host has the three interfaces 10.10.1.1/24, 192.168.0.10/16, and 44.4.1.1/10 and Samba is to use only interfaces 10.10.1.1/24 and 192.168.0.10/16 then you would specify the following in `SMB.CONF`:

```
interfaces = 10.10.1.1/24, 192.168.0.10, 127.0.0.1
bind interfaces only = yes
```

Specific issues fixed in this release

- The error message `NT_STATUS_NO_MEMORY` could sometimes be reported when creating multiple `smbclient` connections. This issue has been fixed.

- PID files were not being deleted when `SMBD` processes terminated, resulting in a build-up of these files. These files are now correctly deleted upon `SMBD` process termination.

- `SMBD` processes would sometimes crash or would sometimes become compute-bound while attempting to crash and produce a crash-dump file. This issue has been resolved.

- A non-admin user was unable to change the password of their local Samba account using the `smbpasswd` utility when Samba was configured as a member server. This problem has been resolved.

- The `net` and `wbinfo` commands would fail if the domain controller required LDAP signing. This matter has been addressed by changing the `SAMBA$CONFIG` menu option "`3B Client ldap sasl wrapping`" to "`3B LDAP connection mode: [PLAIN/SIGN/SEAL]`" and including the necessary functionality in order to support this requirement. The default is "`PLAIN`"; to improve LDAP security set this option to either "`SIGN`" or "`SEAL`".

- The `WINBIND` process could sometimes become hung in a `RWMBX` state. This issue has been resolved.

- It was not possible to map a share if the underlying OpenVMS path included a rooted logical name. This problem has been fixed.

- The SMBCLIENT "!" command would incorrectly spawn to a command prompt and wait for input rather than return to the smb: prompt. Now, to spawn to the DCL prompt the user must enter the command "!spawn".

- Samba no longer defines several CRTL feature logical names in the process logical name table of interactive users who execute SAMBA$DEFINE_COMMANDS.COM.

- Problems were observed with the smbclient "put" command when operating on files containing ASCII 0x19 (CTRL-Y) characters. This matter has now been resolved.

- Various problems related to printing have been resolved.

- Files in SAMBA$ROOT:[VAR.LOCK.MSG_LOCK] directory are now correctly cleaned up when the SMBD process terminates. Additionally, when installing new versions of Samba for VSI OpenVMS, if more than 1000 files exist in the MSG_LOCK directory, the number of files is displayed, and the user is asked if they wish to delete these files before continuing with the installation. If the directory contains more than 100,000 files, answering "Yes" (the default) could significantly extend the time to complete the installation.

- When creating new versions of log files, channels to old logs were in some cases not being closed correctly by SMBD processes, resulting in exhaustion of file descriptors, and causing the process to crash. This issue has been fixed.

- The SAMBA$SHUTDOWN procedure displayed warning messages when attempting to deassign several undefined logical names. This issue has been fixed.

- The smb.conf parameter 'map untrusted to domain' is no longer required and has been eliminated. This line should be removed from smb.conf, if present.