OpenVMS Technical Journal

Setting Up a High-Available E-mail Server Using OpenVMS

Andreas Fassl, Senior Consultant

Abstract

The OpenVMS Cluster technology provides unique features in the standard distribution of the operating system. One can very easily create highly available, highly scalable, and highly secure configurations.

This article shows how to set up an electronic mail system to meet the multifarious needs of companies, starting with small business and ending in typical enterprise solutions.

Prerequisite

While knowledge of OpenVMS is a plus, this article's intention points to a broad audience.

The reading of Keith Parris's article, *Using OpenVMS Clusters for Disaster Tolerance* (http://h71000.www7.hp.com/openvms/journal/v1/disastertol.htm), is a must for a deeper understanding of the capabilities of OpenVMS.

If you're not familiar with the IP world, you'll find a very comprehensive introduction in *TCP/IP Services for OpenVMS Concepts and Planning* (http://h71000.www7.hp.com/doc/73FINAL/6523/6523pro.html).

A good, hands-on practice book, *Linux and OpenVMS Interoperability,* by John Robert Wisniewski, is available from Digital Press.

Importance of Email

Electronic Mail has changed to one of the most important forms of communication. Everyone who had to turn down using email half a day did experience this the hard way.

Most smaller email server installations do meet the need of their operators, but with ascending demands even the smallest risk has to be avoided.

Failures and Counter Measures

To analyze your needs you can use a matrix of the possible failures to your setup and the available counter measures. Table 1 shows possible failures and counter measures.

Failure of...	Single Node Counter Measure	Cluster Counter Measure
System disk	Shadow system disk	Same or cluster shared system disk
Storage controller	Multipath storage	Multipath storage
Network interconnect	Second production LAN	Same
Node	-	Cluster IP
Power	UPS	UPS
Internet connection	Second/third ISP	Second/Third ISP

Table 1: Possible Failures and Counter Measures

Many more failures are possible; the mileage of your experience shows it.

Many vendors do provide counter measures mentioned above, but many of them are very costly and often you have to do a complete redesign combined with a major migration project to scale up your environment.

IP Services

Email is nowadays mostly based on TCPIP. Some of the other protocols are more reliable, aren't that vulnerable against misuse, but the decision has been made in favor of an easy to configure protocol. Another disadvantage of the more secure protocols has been their proprietary design and more restricted configuration rules.

The IP protocol suite has a couple of services. The services needed for an email server are:

SMTP (Simple Mail Transfer Protocol)

Usually tied to the port 25 of a given IP host this protocol is the core for most of all mail transfers. Being created in the "dark ages" of electronic communication, people suffer under the design flaws made in the past. The most "popular" suffering is called SPAM.

Only a few commands are needed to create an email (to illustrate the use of SMTP):

```
VMAL06 $ telnet /port=25 vmcl02.progis.net
%TELNET-I-TRYING, Trying ... 194.49.54.3
%TELNET-I-SESSION, Session 01, host vmcl02.progis.de, port 25
220 vmal06.progis.de V5.3-18E, OpenVMS V7.3 Alpha ready at Sun,
11 May 2003 18:03:09 +0200 (MET DST)
hello
250 vmal06.progis.de Hello vmcl02, pleased to meet you, friend
mail from:afassl@progis.net
250 <afassl@progis.net>... Sender OK
rcpt to:afassl@progis.net
250 <afassl@progis.net>... Recipient OK
data
354 Start mail input; end with <CRLF>.<CRLF>
```

```
Short Message
.
250 OK
quit
221 vmal06.progis.de Service closing transmission channel
%TELNET-S-REMCLOSED, Remote connection closed
-TELNET-I-SESSION, Session 01, host vmcl02.progis.de, port 25
```

Due to this easiness, SMTP had to be extended many times to fight misuse.

Be very careful while setting up an SMTP service attached to the internet – an improper configuration will make your (or your company's) mail server to another SPAM provider. There are many automatic robots scanning around to find active SMTP-ports and probing them to determine their usability for spammers' needs.

Spammers are very creative in their work. A new way observed recently uses an improper proxy configuration of a web server to use a local web server as a relay to a local mail server. The local mail server will accept mail this way because it has been identified the local web server as a trusted source.

The "guest" will leave marks like this:

```
64.70.45.253 - - [13/Apr/2003:07:47:43 +0200] "CONNECT
64.12.138.89:25 HTTP/1.0" 200 8959 "-" "-"
```

This is just an example to remind you - NEVER believe your system (even if it's OpenVMS) is 100% secure.

Keep in mind – OpenVMS engineering uses a very secure-wise approach while designing a new piece of software. When using IP based protocols it always implies inheriting security problems.

A more detailed discussion about this topic you'll find in the previous mentioned book "Linux & OpenVMS Interoperability".

Read more about SMTP on OpenVMS, see *HP TCP/IP Services for OpenVMS Management*(http://h71000.www7.hp.com/doc/73final/6526/6526pro_030.html#smtp_chap)

If you need more features and a very extensive solution, I'll recommend Madgoat Software's MX package. It isn't very expensive; more information can be found at:

http://www.madgoat.com/mx053.html

POP3 (Post Office Protocol Version 3)

Via POP3, Email clients connect to a server supporting this protocol on port 110. The Emails are downloaded to the client and can optionally be deleted on the server after download. Usually POP3 is used for clients who are not online all the time.

Read more about POP3 on OpenVMS, see the *HP TCP/IP Services for OpenVMS Management* manual:

http://h71000.www7.hp.com/doc/732final/6526/6526pro_041.html#pop_chap

IMAP4 (Internet Message Access Protocol Version 4)

IMAP is the most sophisticated non-proprietary Email protocol. IMAP4 (using port 143) is a client/server application offering lots of features to govern Email. The protocol has been implemented in TCP/IP-Services for OpenVMS.

Read more about IMAP4 on OpenVMS:

http://h71000.www7.hp.com/doc/732final/6526/6526pro_043.html

BIND (Berkeley Internet Name Domain)

The Domain Name System (DNS) is a system that maintains and distributes information about Internet hosts. DNS consists of several databases that store host names and host IP addresses. With DNS, there is no central storage of data --- no one server knows everything about all the Internet domains.

TCP/IP-Services Version 5.4 supports Version 9.2.1 of the BIND protocol.

DNS Cluster Load broker

Unique to OpenVMS – the load broker. Other system vendors have to refer to costly hardware based load balancing (CISCO arrowhead, Nortel alteon, etc.) with the disadvantage of another system in the chain between client and server. The load broker gets a key role when scaling up system.

Read more about DNS Load Balancing on OpenVMS, see "Using DNS to Balance Work Load" in the *HP TCP/IP Services for OpenVMS Management* manual:

http://h71000.www7.hp.com/doc/732final/6526/6526pro_016.html

Some Thoughts About…Firewalls

Whilst planning an internet access structure, one of the first claims is a firewall. In several discussions I got a different view to the benefit of firewalls.

- Most attacks are done by insiders.
- Most successful attacks are using (unknown) flaws within the application (buffer overflow e.g.)

In other words – it is verisimilar a firewall is defenseless. You never are discharged from security liabilities just by setting up a firewall. Actually a good secure application server without a firewall is even superior to an unsecured application server protected by a perfect firewall.

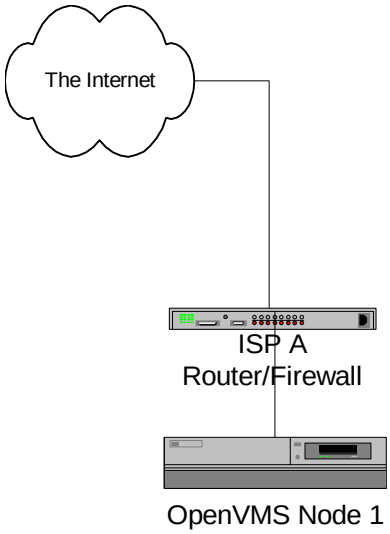
With a simple accounting/summary you'll see, for example, the trails of the attempts to misuse an ftp server (users like: admin, ftproot, informix, linux, lizdy, oracle, pwrchute, rethat, suese, sybase, win2000,etc.)

Hardware Configuration

Single node

We'll start with the smallest configuration, the router will be maintained by your ISP (Internet Service Provider), probably you want to add your own router to separate more networks, but this is the minimum.

Figure 1: Single Node Configuration



To increase availability, one can configure Volume Shadowing for the system and the data disks. This will minimize the risk of failing disks. For more information, see HP *OpenVMS Volume Shadowing*,

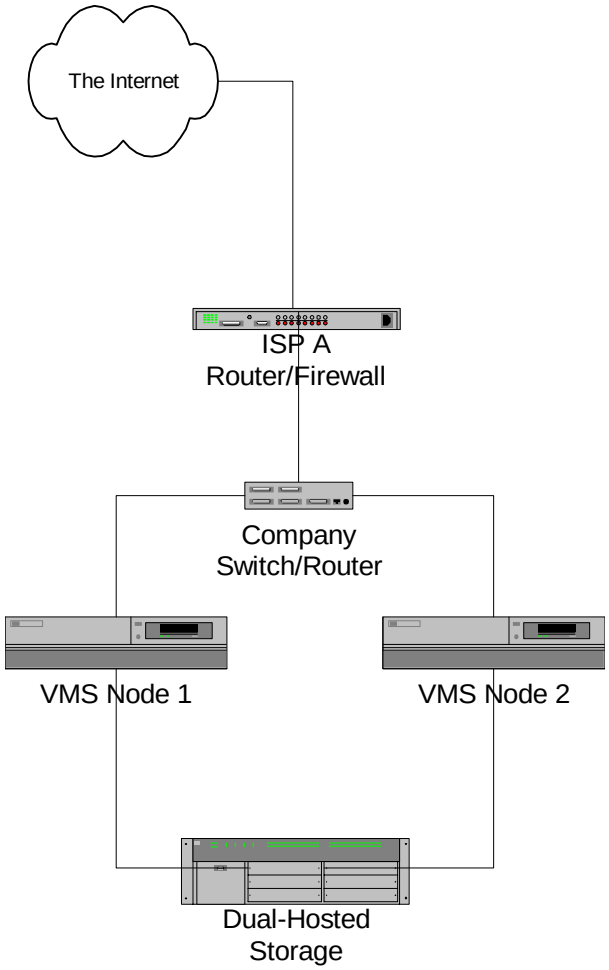
http://h71000.www7.hp.com/doc/732FINAL/aa-pvxmj-te/aa-pvxmj-te.HTML

The configuration consists of a simple OpenVMS installation using a plain SMTP/IMAP4/POP3 setup.

Dual Node

A dual node configuration will reduce the risk of a failing system.

Figure 2: Dual Node Configuration



The configuration uses OpenVMS Cluster Technology, but you have to consider some additional requirements.

Cluster Shared Environment

Most resources can be shared in an OpenVMS Cluster environment. For more information, see "Preparing a Shared Environment" in *HP OpenVMS Cluster Systems,*

http://h71000.www7.hp.com/doc/731FINAL/4477/4477pro_006.html#startup_ch

This must be done for all user data.

Quorum Disk

To ensure the clusters integrity, especially in a small cluster setup, you have to configure a quorum disk. For more information, see "OpenVMS Cluster Concepts" in *HP OpenVMS Cluster Systems,* (http://h71000.www7.hp.com/doc/731FINAL/4477/4477pro_002.html#connection_management)

The quorum disk must be accessible for both members directly. Reliable solutions are based on HSx-controller technology with the corresponding interface cards in the nodes. Please **never** use unsupported configurations. The components have a higher price with a reason. If you're lucky, it will work. If not, no one will (and can) help you. Don't think about standards when talking about SCSI or Fibre Channel, there are so many proprietary enhancements. This is unavoidable to implement the features customers want like improved monitoring, fault detection, etc. The standards don't include those requirements, so every company has to develop and verify their own supported configuration for a comparable small market. This means lower trade volumes, higher prices.

Configuration of the IP Services

All the TCP/IP services do support OpenVMS Clustering. For more information, see "Configuring and Managing BIND Version 9 in *HP TCP/IP Services for OpenVMS Management,*

http://h71000.www7.hp.com/doc/73final/6526/6526pro_008.html#config_cluster

In this small configuration a simple cluster IP-address should be sufficient.

Both nodes have a node-specific IP-address and a cluster-wide IP-address:

```
VMAL06 $ mc sysman
SYSMAN> set env/clus
SYSMAN> do tcpip show inter/clus
%SYSMAN-I-OUTPUT, command execution on node VMAL05
                                                          Packets
Interface   IP_Addr         Network mask          Receive          Send     MTU

 LO0        127.0.0.1       255.0.0.0                 107           107    4096
 WE0        194.49.54.1     255.255.255.0          324377        203105    1500
   Cluster  194.49.54.3     255.255.255.0

%SYSMAN-I-OUTPUT, command execution on node VMAL06
                                                          Packets
Interface   IP_Addr         Network mask          Receive          Send     MTU

 LO0        127.0.0.1       255.0.0.0                2003          2003    4096
 WE2        194.49.54.2     255.255.255.0         1210418        988863    1500
      Cluster  194.49.54.3     255.255.255.0   Impersonator
```

If a client program connects to a cluster address, it will get typically (due to certain limitations in the implementation) only one node.

At the latest from this stage on OpenVMS will show it's power. You can take over the complete configuration of the single node solution as it is and you only need to move it in a cluster shared directory.

A single line in your SMTP Startup (for the TCPIP Services SMTP)

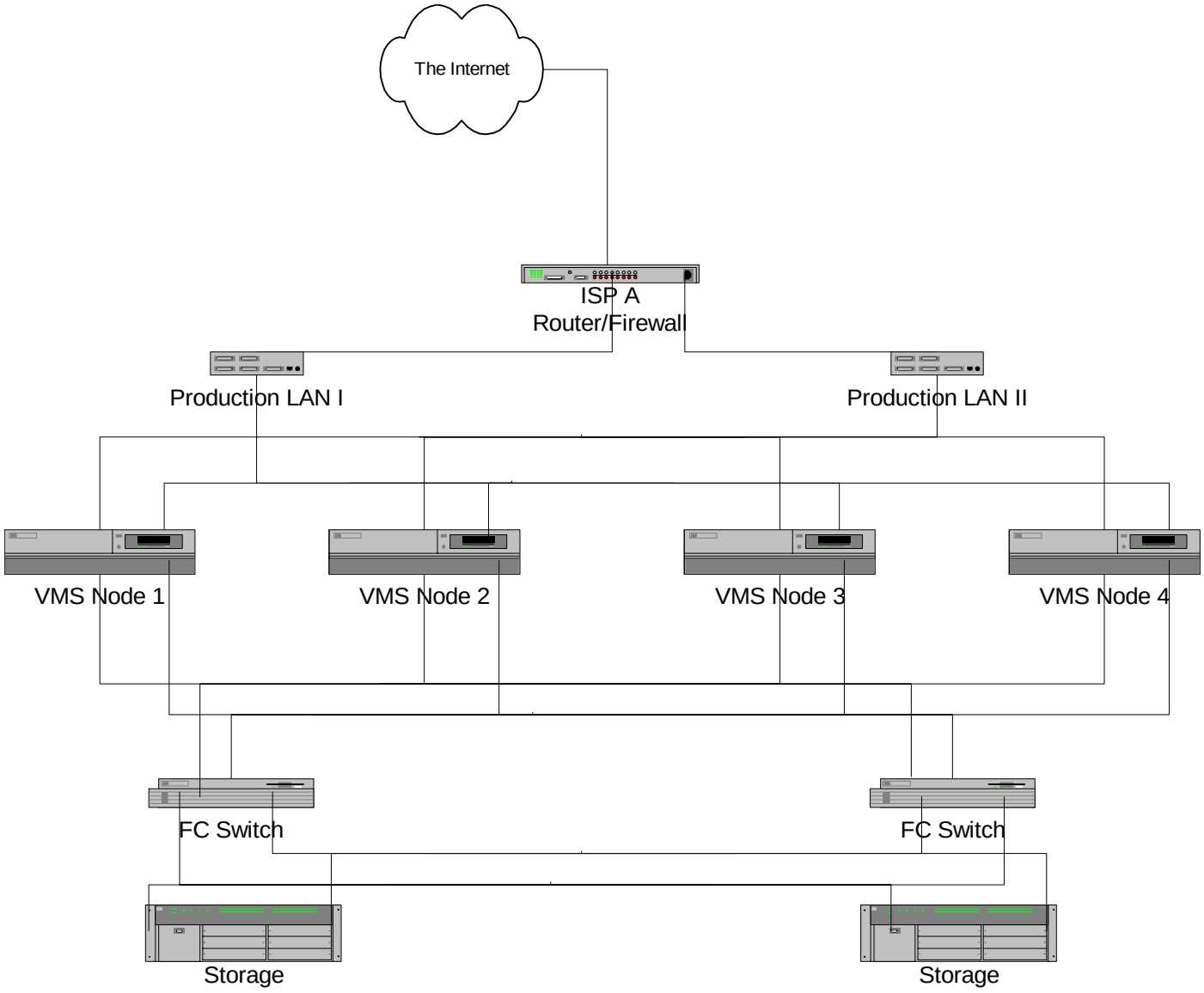
$ DEFINE/SYSTEM TCPIP$SMTP_COMMON cluster_device:<clusterdir>

represents the major changes to your configuration.

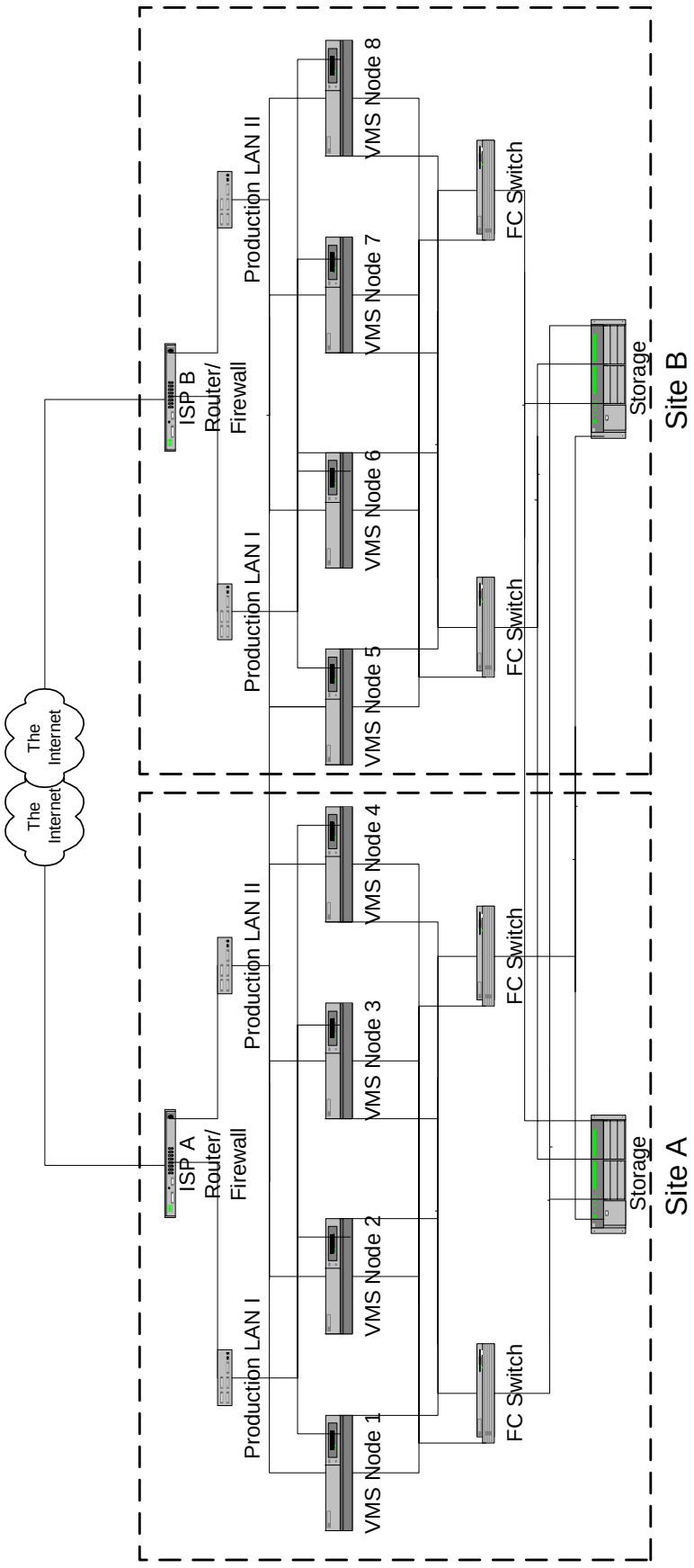
Multi Node

The next step is to integrate more nodes to balance the load among the members. As you can see, this setup has an obvious flaw – the contact to the Internet.

Figure 3: Multi Node Configuration



To eliminate this bottleneck, you have to choose the next complex solution.



To be honest, this is a setup needing a very careful engineering, but it is possible with justifiable effort. The most expensive part of this configuration are fibre channel links that have to configured, providing possible line lengths up to 100 kilometers per link, using single-mode fiber and up to 600 kilometers per link with FC/ATM links.

And – most important – you don't need to reengineer your previous setup – you can use all the setup, all the user data, all the files without any changes.

Benchmarking

How does one proof the proper configuration of his environment? There are many benchmarks available, I recommend the mstone benchmark developed by Netscape now available as an open source project.

The former Netscape Server development had a customer oriented stress test called Mstone.

Mstone is a multi-protocol stress and performance measurement tool. Mstone can test multiple protocols simultaneously and measures the performance of every transaction. The performance can be graphed throughout the duration of the test.

Where To Get Mstone

The complete source tree had been given to mozilla.org.

Surf to: http://www.mozilla.org/projects/mstone/

Supported Operation Systems

Mstone currently runs on recent versions of: Linux, Solaris, AIX, OSF, HPUX, and NT. Any OS with POSIX threads support should be an easy port. The test client machines can each be running different operating systems. Common utilities like perl, gnuplot, and gd are used and can be packaged with mstone for completeness.

Installation

The easieast way to install mstone is to use a linux box. In a later step we will provide a OpenVMS port of mstone.

Setting CVSROOT

```
$ CVSROOT=":pserver:anonymous@cvs-mirror.mozilla.org:/cvsroot"
$ export CVSROOT
```

The password for user anonymous is anonymous.

Getting the Source

```
$  cvs co mozilla/mstone
cvs server: Updating mozilla/mstone
U mozilla/mstone/Building
```

```
$ cd mozilla/mstone
$ gmake rpackage
```

You should now have a complete runnable tree under a platform-specific directory under build/package.

Initial Configuration

Run "mstone config." It will ask you about your system configuration. Fill in the appropriate values and create the optional user accounts and broadcast account. When it asks about client machines, enter them seperated by commas, with no spaces (e.g. host1,host2,host3). If you need to re-configure, run "mstone config".

The machine starting the test may also be a client. For accurate results, clients should not be run on the test mailserver machine (or its directory server). If all the client machines are not running the same operating system version, see "Configuring Client Machines" below to configure for different OSes.

Setup only configures the most important parameters. If you have more advanced needs, edit conf/general.wld appropriately.

Run "mstone setup". It will now push the necessary files to each client machine. If there are problems (i.e. with rsh permissions), fix them and re-run "mstone setup" until everything works.

Install Test Accounts

With this small DCL-program you can create for example 100 users.

```
$ ! DCL-Script to generate MSTONE Test-Accounts
$ !
$ !
$ ! Start User Number
$ NUM=1
$ ! Start Group
$ START_GROUP=%o4000*%o200000
$ loop:
$ ! Create an username based on num
$ USERNAME="MST_''num'"
$ ! Calculate the UIC
$ uic = f$fao("!%I",START_GROUP + NUM)
$ write sys$output "Creating User : ''USERNAME'"
$ mc authorize add /UIC='uic'
'USERNAME'/owner=mstone/account=MBench/
dev=$100$dka0:/dir=[mstone.accounts.'username']
$ create/dir/owner='username' $100$dka0:[mstone.accounts.'username']
$ num=num+1
$ if num.lt.100 then goto loop
$ exit
```

And with this one you can dump them

```
$ ! DCL-Script to remove MSTONE Test-Accounts
$ !
$ !
$ ! Start User Number
$ NUM=1
$ ! Start Group
$ START_GROUP=%o4000*%o200000
$ loop:
$ ! Create an username based on num
$  USERNAME="MST_''num'"
$ ! Calculate the UIC
$ uic = f$fao("!%I",START_GROUP + NUM)
$ write sys$output "del User : ''USERNAME'"
$ mc authorize rem  'USERNAME'
$ num=num+1
$ if num.lt.100 then goto loop
$ exit
```

If you want to, you can do some more programming to be more flexible with your test setup, for the start this will be enough.

Run Tests

Try it out. Use small process and thread counts until everything is working.

```
mstone smtp -t 30s
```

The script will tell you how many processes and threads it is running on each system and where errors are logged. At the end of the test, it will print out a URL for the test results and an indication of the size of the errorlog file (stderr).

The results of the mstone run will display statistics for each protocol that was tested. The results are presented in both a HTML web page and a text file. The text file is simple and uniform, while the web page is more user readable. The web page has links to the test configuration files, error log, and the text version.

Resources to watch

With this test suite it is possible to simulate all thinkable load profiles. So can first watch and tune for a single user with one message size, mixed message sizes, ascending user accounts, etc.

What you have to watch:

- TCPIP-Ressources like per process memory
- IO-Saturation
- CPU-Load

You can use the basic tools like MONITOR, switching over to more sophisticated tools like the „Availability Manager“ or for more professional (but costly) analysis use products like CAs performance advisor (formerly known as Polycenter Performance Advisor). The newer (and free) ECP (Enterprise Capacity Planner) should be worth a look as well.

Customize tests

Copy and edit the scripts (e.g. "conf/pop.wld") to define new tests. The CONFIG section specifies all the attributes used in the test.Other sections specify the protocols to be tested and the parameters for them.

All switches can be overridden on the command line to facilitate easier testing. The exact configuration (include command line overrides) is stored with the results from each test.

Maintenance

You can run "mstone setup" at any time (except during a test :-) to update the files on the client machines.

Use "mstone cleanup" to remove the files created by "mstone setup". After the test is finished, the directories under "tmp/" can be compressed or deleted to save space. All the information about a test run is stored in the "results/" directories.

Configuring Client Machines

Edit conf/general.wld to include CLIENT sections for each machines to use. You can also specify the OS type for each client machine. Set the "Arch" parameter in each CLIENT section as appropriate (e.g. SunOS5.6, Linux2.2_x86, AIX4.2, HP-UXB.11.00, IRIX6.5, OSF1V4.0, WINNT4.0). The directories under "bin" specify the available OS types. For NT4.0 clients with a UNIX test master, you will need to configure "command" and "tempDir" for proper operation. See the "HOSTS=winnt01" example in conf/sample.wld.

The total number of processes and threads that can be supported on a client is dependent on the number of commands in the test, the OS, and available memory. Check the stderr log for messages about not being able to create processes or threads. Check on the client machines during the test and make sure they aren't running out of CPU. The UNIX programs "top" and "vmstat" are good for this. If the client CPU is more than 75% busy, use more machines. Also watch out for network saturation. You may have to use machines with separate networks to the server to reach full server load.

Summary

The discussed configurations aren’t very complicated. And they are:

- Easy to design
- Easy to implement
- Easy to maintain

If you start your configuration with the smallest cluster configuration, the biggest advantage you'll get is an environment, that will have no down time, especially for:

- OS Upgrades
- HW-Upgrade of single nodes
- Adding additional nodes
- Doing site movements

These are the great demands companies have on IT solutions. OpenVMS meets the demands, and more …

This configuration can be used for many other IP-based services.

To name only a few of them:

- NTP (for high reliable time stamps)
- NFS (as a 100% available UNIX file server)
- SMB (as a 100% available windows file server)

For more information

Contact:

ProGIS Software & Beratung

Dipl.Ing. Andreas Fassl

Spichernstraβe 59

50672 Köln

Deutschland

http://www.progis.de
mailto:afassl@progis.de