Gateway High Availability
Principals and Practices

1. OBJECTIVE

When we talk about High Availability (HA), we quickly think about component redundant concepts, single points of failure, service level agreements (SLA), costs involved and maximization of Return of Investment (ROI).

All this points are valid and must be observed during the project and operation of structures in HA, but there are other factors that must be evaluated and maintained, so that correct functionality of HA is insured.

This paper is meant to enumerate and discuss these points so that, after full comprehension of these concepts, all the continuity principles are satisfied.

2. DEFINITIONS

We define High Availability (HA) as a principle of project and structure of devices in order to insure the service Levels (SLA) required from the related resources.

Its understood for Service Level Agreement (SLA), as a measure of the quality associated to a certain service, resource or system and agreed upon by its provider and associated user, defining minimum acceptable parameters. When talking about SLA values associated to high availability we use the availability calculations during a determined interval of time (usually 1 year) of these services. In Appendix A this definitions of availability and SLA are presented in mathematical form.

Giving the name of Point of Failure to the element inside the service flow that when it has its normal operation interrupted or degraded, compromises the accorded SLA.

Naming the cluster node of one of the machine that belongs to this cluster.

Defining as recovery time the time interval from the failure of a node to the transfer of the service load corresponding to the other cluster nodes.

IPSO is an operating system that runs in the Nokia IP Security line. It is an UNIX operational system with specific optimizations to obtain maximum performance and security, and best suited to applications that demand maximum security and reliability.
3. HA PREMISES

Using as an example a firewall cluster composed of two machines (figure 1), in case of failure of one of the firewalls, the node that remains working has to incorporate the performance demand of the whole cluster.

![Figure 1 – Topology example in HA.](image)

For the high availability objective to be reached, we must orient / direct the project under the following aspects: Topology, Architecture, Synchronism, Sizing and Monitoring.

3.1 Topology

We must provide redundancy of resources and set them up in a way to eliminate single failure points. Accordingly all the devices employed for the implementation of the cluster must have redundancy, including switches, routers, links WAN, etc.

Using the example in figure 1, we notice that all the component interfaces of the cluster must be connected to the same networks, whichever the environment (LAN & WAN).

Parameters of high importance during topology definition of what parts/components should be placed with redundancy (including internal ones like power supply, HDD, communication links, etc) are MTBF and MTTR\(^1\). As explained on appendix A, these values are invaluable to predict statistically the resulting cluster availability.

MTBF should be provided by the equipment/component manufacturer, which uses several variables for its calculation including the way these components are connected/placed in the data flow and environmental conditions (mainly temperature).

It’s worth noticing that while the Amdahl’s\(^2\) law is true for system performance, stating that “the performance enhancement possible with a given improvement is limited by the amount that the improved feature is used”, the Weakest Link principle is better suited for High Availability. For this reason, the efforts and investments should be done accordingly.

---

\(^1\) MTBF is the “Mean Time Between Failures” and MTTR is the “Mean Time To Repair”.

\(^2\) For detailed explanation, see “Computer Organization and Design” reference.
3.2 Architecture

So that the redundant resources can be utilized in case of failure of one of its components, we must provide a mechanism, be it an automated or manual process, to achieve the values of availability with the SLA agreed upon with the users.

Thinking about MTTR, it concerns not only by the time to replace the defect part of equipment or the SLA to replace it according to a service contract. It includes also other variables or phases that must be followed and considered in order to assess the correct time required to return to the normal state:

1) Failure Detection
2) Analysis and diagnose of the problem
3) Repair, defect part replacement and return to the normal state

The chosen architecture should account for each of those phases and propose solutions to minimize the time required on them, resulting on a MTTR that can be used to calculate the expected availability.

3.2.1 Manual Operation

Also called of “cold stand-by” HA architecture, where the component in redundancy remains ready to begin its operation, but depends on manual handling to be activated. This process can involve only quick reconfigurations such as the IP addressing or route change, or even more complex operations involving link and external equipment (routers, switches, etc.) reconfiguration.

The effectiveness of this method resides in its power of monitoring / detection of failures and on the efficiency of the support and operations team in charge of applying the procedure.

As this process implies a high undetermined\(^3\) recovery time it can be applied for non-critical environments, with SLA based upon low availability values.

For highly critical environments, this kind of HA architecture involving manual operation is quite inefficient, or inadequate.

3.2.2 Automatic Operation

Through mechanisms (protocols and processes) that do not require human intervention to maintain the availability (D) inside standards agreed upon in the SLA. The detection of a failure, re-routing of the load (entering a contingency state), the recovery detection (returning to normality state) are services performed by this operational method.

An important characteristic is that automated operation allows the determination of the cluster’s recovery time, allowing prediction of availability D of the environment and justifying more precisely the amount of investment required to maintain the SLA.

There are basically two strategies:

---

\(^3\) Not predictable, from minutes to hours, according to the elevated number of variables involved and to inevitable human factors.
• Active/stand-by

In this case one or more cluster nodes remain active while others are on stand-by, waiting to get into operation in case of failure by one of the active nodes.

An example of this strategy type is the VRRP\(^4\) (Virtual Router Redundancy Protocol). Through configured priorities of each node in the cluster, the protocol checks which node must be active (master) and insures that all the others remain in stand by and will only enter operation in the appropriate moment, in the right priority order, to maintain availability.

The recovery time of a cluster implemented with VRRP is defined by:

\[
T_{\text{Recuperação}} = 3 \cdot T_{\text{Advertisement}} + \left( \frac{256 \cdot \text{Prioridade}}{256} \right)
\]

As predicted in the protocol, the master sends multicast advertisement packages in 1 by 1 second intervals by default, so that recovery time of the VRRP is 3 to 4 seconds. Starting with IPSO 3.8.1, VRRP configurations are supported with IPv6 protocol in addition to IPv4.

• Active/active

In this case all the cluster nodes remain active; in case of a node failure, the service is maintained by the remaining nodes. Although in a superficial analysis we can be tempted to use this kind of architecture to have all the investment in equipment being effectively used, meaning, without having idle equipment, we will see ahead in the sizing section that in order to maintain the HA premisses we are obliged to have cluster resources (memory for allocation tables, traffic processing capability, simultaneous connections quantity, etc) in idle, independently of the technology used, have a active-active or active-stand-by philosophy.

Some examples of technologies that implement active-active architectures are:

- **VRRP with load sharing** – Even though it uses VRRP, that is a protocol with active-standby philosophy, it is possible to implement a structure on top of it, in a way to have more than one machine as master, as with different VRIDs. Example, one cluster with two machines, where one will be master for a virtual address and the other master for a second virtual address, and each is a stand-by for the other; in case of failure of one of them, the other machine will answer for both virtual addresses.

- **Nokia IP Clustering** – This technology was incorporated to the IPSO in the 3.6 version, with major functionality improvements on IPSO 3.7 (forwarding “unicast” mode) and IPSO 3.8 (Switch Failure Recover), with the benefit to allow the implementation of an active-active cluster structure in a very easy form. Supporting up to 4 nodes, it is a great option for those environments that demand scalability at a low cost. Much more than one HA protocol, it allows balancing of the load between the nodes that compose the cluster. Its recovery time is by default 4 seconds.

---

\(^4\) IETF RFC 2338
OSPF routing protocol– It is possible to use the link state algorithm of the OSPF to obtain HA, allowing it also to distribute the load when used in conjunction with the functionality of route with gateways of the same cost (Equal Cost Multipath). The problem is that the multipath algorithms tend to be different depending of the manufacturer of the equipment, allowing generation of asymmetrical routes (and consequently packet being discarded by the stateful inspection of Firewall-1), especially when NAT and the tunneling are used (ex IPSec) ending in the cluster. Another limitation is in relation to the recovery time that becomes a function of convergence time of the OSPF.

Load Balance Switch (L4) – As known as “Switch Sandwich”, this method is an efficient way to create clusters with load balance policies and potentially more scalability of the number of cluster nodes. At the same time, these switches have higher costs than regular L2 switches and must include High Availability solution to avoid itself being a single point of failure.

It is important to note that Architecture and Topology influenciate on each other's requirements. For instance, IP Clustering requires the so called “Cluster Management Network” to be implemented Out of Band on at least one Fast Ethernet port using L2+ switches, meaning that the IP Clustering Architecture demands Topology connections to work properly.

3.3 Synchronism

When we talk about many equipments processing information in a way to combine efforts and make a cluster, starting from the premiss that in case of failure of one of the nodes, the cluster will maintain the connections that were active, without sacrificing availability, it is essential that there is information that is common to all of the nodes.

Clock (hour), policies, connection parameters such as routes and virtual addresses are important information to be synched. All of these resources have a way of synchronism through protocols or dedicated services.

A good example is the NTP\(^5\) for clock synchronism. This protocol implements a structured hierarchy of time reference. When we talk about cluster firewalls, it is important that an internal clock reference exists, meaning, one NTP server managed and protected in the internal network.

In relation to synchronism of pre-established connections, such as the Check Point Firewall-1, implementing a way to inspect the packages called stateful inspection, each new connection (or connection context, for connectionless protocols) that is validated by the security policy configured in the firewall is added to a connection table. All the subsequent packages from this same connection don’t have to be checked/validated by the security policy, since there is an entry for that connection in the connection firewall table. To insure absolute no loss of connections during a failure process of one of the nodes of the cluster of firewalls, it is fundamental that this mechanism of synchronization be functioning correctly\(^6\) and efficiently. That can be done by dedicating a specific network for that end, with appropriate network bandwidth and selecting the services (FTP, SMTP, etc) that really must be synchronized or those that are not to be synchronized\(^7\).

\(^5\) Network Time Protocol – IETF RFC 1305
\(^6\) Synchronism of the clocks between all the cluster nodes is essencial for the connections synchronized operation.
\(^7\) The less data to be synched and more efficient the communication between the machines, the more efficient the sync will be.
3.4 Sizing

Giving that the cluster in figure 2, composed of N nodes. To simplify the approach, we will suppose an architecture with all nodes active with equally distributed loads to each node.

Defining P as the acceptable share of loss/failure of the cluster, in number of nodes, with no deterioration of total processing capability.

In case of failure of a quantity of P nodes, the resulting cluster with N-P nodes should support the C load demanded of the cluster. Soon considering Q as being the individual capacity of each node of the cluster\(^8\), then:

\[
C = (N - P) \cdot Q \quad \text{with} \quad N > P \geq 1
\]

Defining yet, R as being the real load applied in each active node of the cluster, under normal conditions with all operational nodes, then:

\[
R = \frac{C}{N} \quad \Rightarrow \quad R = \frac{(N - P)}{N} \cdot Q
\]

\(^8\) Considering only the cases where all nodes in the cluster have the same capacity
In reality, what we seek to size as a premiss of HA is the maximum percentage load that must be used in each node (lets call it $F$) and the minimum percentage idle charge that must be maintained in each cluster node (lets call it $O$).

$$F_{\%} = \frac{R}{Q} \cdot 100 = \left(\frac{N - P}{N}\right) \cdot \frac{Q}{Q} \cdot 100 \quad \Rightarrow \quad F_{\%} = \frac{(N - P)}{N} \cdot 100$$

$$O_{\%} = (1 - F) \cdot 100 = \left(1 - \left(\frac{N - P}{N}\right)\right) \cdot 100 \quad \Rightarrow \quad O_{\%} = \frac{P}{N} \cdot 100$$

<table>
<thead>
<tr>
<th>$P$</th>
<th>$N$</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>50%</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>33%</td>
<td>67%</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>25%</td>
<td>50%</td>
<td>75%</td>
<td></td>
</tr>
</tbody>
</table>

Table 1 – Average Idle Factor (%) of each node calculated for some values of $N$ and $P$.

As we see in table 1, for each value of $P$ the idle factor falls a lot as the cluster grows in $N$ nodes quantity.

We can draw some conclusions from the result in that table:

- For a cluster with 2 nodes, the premiss of HA demands that there is 50% of idle per node. That means in this case it is mandatory to have the equivalent of a machine reserved to HA, independently if the architecture employed follows an active-active or active stand-by structure.
- The values of idle with $P=1$ are the lowest, favoring a project configuration that we will call n+1, that means, using $n$ machines to supply the service demand and adding one more node to supply the HA premiss. In Appendices B, I present a study of concept proof saying that this kind of project is the one that offers the best-cost efficiency rate.

---

$^9$ Reminding that $F$ as well as $O$ are average values per cluster. In our example with $N=2$, we would have in this case an active-standby architecture up to 100% of load in one of the nodes and 0% in the stand-by node (maintaining the 50% average idle in the cluster).
3.5 Monitoring

The fifth and last aspect to be observed in HA projects is the configuration of an efficient monitoring scheme, important not only for failure identification in the cluster and SLA maintenance, but also for a more pro-active performance in eventual problems and promoting the necessary scalability to keep up with the traffic demand growth.

Monitoring allow us to establish history data of devices failure and how much time taken to return to the normal condition, allowing then to compare these data with other data provided by manufacturers (MTBF) and agreed on service contracts (MTTR), providing arguments to talk to these suppliers about quality problems.

This monitoring can be done automatically through:

- **SNMP** – Defined in the RFC 1157, this protocol can be used for monitoring through one SNMP Manager (HP Openview, Tivoli among others), with the use of configurable traps in the cluster nodes or with the use of MIB for a more detailed information search such as memory allocation, etc. Object specific information (OIDs) and configurations, can be obtained in the documentation of the operational system IPSO or through the whitepaper listed in the references. From the IPSO version 3.6, when activated, the SNMPv3 protocol is mandatory because the version 3 has basically the same functionality of the SNMPv2 with additional security resources. In Appendix C, the most useful HA monitoring traps are listed.

- **Check Point SmartView Monitor e SmartView Reporter** are tools that can supply important information of the Firewall/VPN and useful reports for monitoring the performance, an important point so that the HA sizing premises can be maintained. Details can be obtained directly at the Check Point site (http://www.checkpoint.com).
• OPSEC solutions, certified by Check Point. ([http://www.opsec.com](http://www.opsec.com)).
• Open source tools are available to help on monitoring, for instance MRTG and Cricket. It is out of scope of this document to list all these tools, but some of these tools are quite interesting and worth trying depending on the application.

An interesting parameter to be monitored is the Defects per Million (DPM). It is a method to assess availability describing the number of failures during one million hours of operation time of a device (HDD, Power Supply, etc), equipment (appliance, switch, router, etc) or network. While MTBF is a parameter provided by the device manufacturer, the DPM is a value particular to a network or equipment. Appendix A describes the method used to calculate DPM in detail.

4. CONCLUSION

We presented the theories and practical fundamentals for the HA project and the premisses that must be observed during the operation of the solutions so that the objective is reached. The conclusion we draw is that the final objective of HA is only obtained when observing all the project's points and premisses. It is not enough, for example, to dedicate efforts only in the architecture, without the correcting and monitoring.

As a last recommendation, periodical tests must be done in order to insure that all the five aspects presented (Topology, Architecture, Synchronism, sizing and monitoring) are in accordance with the expected results.

5. REFERENCES

- IETF RFC 2338 – Virtual Router Redundancy Protocol"
- “Nokia Clustering IPSO 3.6 / Check Point NG FP3 Load Balancing Configuration Document”, Jeff Mousseau – [http://www.digitalmigrations.com](http://www.digitalmigrations.com) -
- "Performance Monitoring using SNMP for Nokia Platforms", Resolution 11032, Nokia Knowledge Base (support.nokia.com)
- “What is the MTBF rate for Nokia IP Security Platforms?”, Resolution 375, Nokia Knowledge Base (support.nokia.com)
- “Computer Organization and Design”, David A. Patterson, John L. Hennessy, Morgan Kaufmann
APPENDIX A

Availability and SLA Mathematical Bases

We can mathematically define the availability of a device through two formulas, the first being based on statistics of the Mean Time Between Failures (MTBF) and the Mean Time Through Repair (MTTR), which we will call as forecast availability ($D_p$), the second being through the unavailable logs in a determined period of time (usually 1 year\(^{10}\)), which we will call measured availability ($D_m$).

$$
D_p = \frac{MTBF}{MTBF + MTTR}
$$

$$
D_m = \frac{t_{year} - \sum \Delta t_{unavailability}}{t_{year}}
$$

where $t_{year}$ it stands for the amount of unavailable seconds per year

The previewed availability is an important data in the sizing and the allocation of resources/ investments attempting to service a determined SLA. As far as mean availability goes, it is an important value for evaluating if the service has effectively reached the agreed upon levels of quality. On both cases, the conditions underneath should be respected.

$$
SLA_{year} \geq (1 - D) \cdot t_{year}
$$

Another way to describe availability is through the Defects per Million (DPM) method, which evaluates historically the number of failures occurring on a million of operation hours of a device (HDD, Power Supply, etc), equipment (appliance, switch, router, etc) or network. While MTBF is a parameter provided by the device manufacturer, the DPM is a variable that can be collected during the operation and compared with other data in a way to check for quality problems or expected failure behavior.

In a more general point of view, to keep DPM of a whole network, the total time of operation in hours is the sum of operation hours of all devices in this network. For instance, for a network with 5000 devices (PCs, switches, routers, communication links, etc) the total time of operation is $5000 \cdot \Delta t$.

Basically, to calculate the DPM we use the following proportion rule:

$$
\frac{n \text{ (failures)}}{\Delta t \text{ (hours)} \cdot N \text{ (devices)}} \quad \rightarrow \quad \text{DPM} = \frac{n \cdot 1.000.000}{N \cdot \Delta t}
$$

As we see, monitoring the number of failures $n$ during a period of $\Delta t$ hours of operation of $N$ devices, we can calculate the DPM during that period of time. For instance, we can have the DPM for the last month, for the last year, since the beginning of the year and/or for the last 12 months, and so on.

An substantial increment of DPM during a period to the other (for instance, from on month to another) could be a excellent indication that it is time to take actions more proactive or maybe contact the manufacturer of the device.

\(^{10}\) 1 year = 8766 hours, accounting for leap years; 1 month = 1/12 year = 730.5 h, and so on.
CONCEPT EXAMINATION: PROJECT n+1\(^{11}\)

In order to prove that the best option for project is the n+1, we will use three concepts:

- Total ownership cost (TCO) of HA
- Return over investment (ROI) of HA
- Availability

When we refer to a n+1 configuration, it means that we are taking under consideration that n machines will handle the cluster traffic and we have reserved one more machine for redundancy, therefore, under these conditions we have \(P=1\). What we mean to prove is that the functions \(TCO=f_1(P)\), \(ROI=f_2(P)\) and \(D=f_3(P)\) represent an excellent solution to \(P=1\).

Determining \(TCO=f_1(P)\)

Considering the cost per cluster node \((I)\), the HA cost may be defined as

\[
I_{HA} = N \cdot I \cdot O
\]

As we have previously seen,

\[
O = \frac{P}{N}
\]

If we substitute the previous equation, we will have

\[
I_{HA} = I \cdot P
\]

Therefore, the high-availability cost is directly proportional to \(P\) (number of nodes in the clusters that may fail without compromising the SLA).

Determining \(ROI=f_2(P)\)

We may translate the return over the investment in qualitative form as the cost benefit relationship

\[
ROI = \frac{Benefit}{Cost}
\]

The fundamental benefit of HA is to satisfy the compromised SLA, and this condition is satisfied from \(P+1\). If \(P>1\), the benefit would only be clear when equipment repair time is very high. Therefore, the benefit (to attend the SLA) tends to be constant, independently of the value of \(P\).

The costs related to \(P\) have already been defined on the previous item.

\(^{11}\) We have used \(n\) in lower case in order to differentiate the \(N\) (total number of nodes in the cluster). In this case, \(N=n+1\).
Then, we have an ROI equation that can be rewritten as such:

\[
ROI = \frac{Benefit}{I \cdot P} \quad \Rightarrow \quad ROI = k \cdot \frac{1}{P}
\]

where \(k\) is a constant.

Therefore, the return over the HA investment is inversely proportional to \(P\) (number of nodes in the clusters that may fail without the SLA compromise).

**Determining \(D = f_3(P)\)**

Considering that \(q_{\text{failure}}\) as the probability of cluster failure due to \(P+1\) nodes having presented failures (meaning, if SLA is impacted) and defining \(q_{\text{node}}\) as the probability of failure in one node, we then have

\[
q_{\text{failure}} = \left(q_{\text{node}}\right)^{P+1}
\]

In order for us to reach such failure conditions of \(P+1\) nodes, we will have short intervals of unavailability from the recovering time of redundant nodes. Therefore, we will have a \(P\) total of times the recovering time of each node.

Considering the sum of the unavailability periods as the sum of the parcel considering \(q_{\text{failure}}\) and the parcel coming from the recovering time of \(P\) nodes, we have

\[
\sum \Delta t_{\text{unavailability}} = q_{\text{failure}} \cdot t_{\text{year}} + t_{\text{recovery}} \cdot P = \left(q_{\text{node}}\right)^{P+1} \cdot t_{\text{year}} + t_{\text{recovery}} \cdot P
\]

Considering availability as described in the equation underneath.

\[
D = \frac{t_{\text{year}} - \sum \Delta t_{\text{unavailability}}}{t_{\text{year}}}
\]

and substituting by using the above developed equation, we have

\[
D = \frac{t_{\text{year}} - \left(q_{\text{node}}\right)^{P+1} \cdot t_{\text{year}} + t_{\text{recovery}} \cdot P}{t_{\text{year}}} = t_{\text{year}} \left[1 - \left(q_{\text{node}}\right)^{P+1}\right] + \frac{t_{\text{recovery}} \cdot P}{t_{\text{year}}}
\]

We know that \(t_{\text{recovery}} \ll t_{\text{year}}\), therefore, we can infer the following

\[
D = 1 - \left(q_{\text{node}}\right)^{P+1}
\]
Conclusions

Figure 4 shows the graphic\(^{12}\) normalizing the three parameters analyzed as a function of \(P\).

Through the graphs, we can conclude that:

- For a very low recovery time \((t_{\text{recovery}} < t_{\text{year}})\), the availability is little influenced by \(P\).
- ROI reaches its maximum with \(P = 1\).
- TCO reaches its minimum with \(P = 1\).

Therefore, as we have wished to demonstrate, the excellent point of these three parameters is reached with \(P = 1\), which means, with a n+1 configuration.

\(^{12}\) Take note that the functions presented are only defined with \(P \geq 1\).
APENDIX C

List of useful Traps for HA\textsuperscript{13}

- SNMP v2
  - coldStart

- Interfaces
  - linkDown
  - linkUp

- System
  - systemTrapLowDiskSpace
  - systemTrapNoDiskSpace
  - systemTrapDiskFailure
  - systemTrapDiskMirrorSetCreate
  - systemTrapDiskMirrorSetDelete
  - systemTrapDiskMirrorSyncFailure
  - systemTrapDiskMirrorSyncSuccess
  - systemOverTemperature
  - systemPowerFailure
  - systemFanFailure
  - systemTrapSnmpProcessShutdown

- Link Aggregation
  - ipsoLinkAggregationMemberActive
  - ipsoLinkAggregationMemberInactive

- VRRP
  - vrrpTrapNewMaster

- IP Clustering:
  - ipsoLBClusterNewProtocol
  - ipsoLBClusterMemberJoin
  - ipsoLBClusteringMemberLeft
  - ipsoLBClusteringNewMaster
  - ipsoLBJoinReject

\textsuperscript{13} From IPSO v 3.8.1 on (see References). This list does not have all the existing traps. The MIB of IPSO, which has all the traps and OIDs may be found in the directory /etc/snmp/mibs.