XRoads Networks

XROADS NETWORKS – WHITE PAPER

Vector Routing

Vector Routing

The purpose of this paper is to provide an understanding of XRoads Networks' patent-pending Vector Routing technology that is built into its XRoads Edge product line.

Vector Routing is designed to provide two fundamental services, load balancing network traffic across many paths and ensuring redundancy in the case that one or several of those paths fail.

Background

In computer networks, such as the Internet, preventing a smaller portion of the network, or local network (one with only several connections to the rest of the network), from losing connectivity to the rest of the network can be accomplished by providing redundant paths to various points within the larger network.

The Internet as a whole is based on a routing scheme that uses IP address information in order to determine where a packet of information needs to be sent.

Vector Routing ensures redundancy by mitigating, and even eliminating network downtime by employing non-BGP multihoming. The term "multihoming" is used to describe a network that utilizes multiple connections to one or more Internet Service Providers (ISPs). Provisioning two or more connections to the Internet has become the primary means by which organizations build high availability into their access points.

It used to be that only by implementing a routing protocol, known as Border Gateway Protocol (BGP), could an organization deploy a multihomed solution. However, deploying BGP is costly, complex, and requires the cooperation of your ISP(s). In addition, network congestion is a limitation of BGP that causes over 50% of network traffic to be sent over sub-optimal routes.

Many products today are capable of providing connections to two or more diverse paths and use a variety of methods to determine when those paths are available or not available.

The problem is that many of these methods rely on complicated routing protocols to determine whether the path is acceptable for transmitting data traffic over it or not. Beyond being complicated, these routing protocols do not do a very good job of determining how well the path is performing for the end user. As long as data traffic is able to get to its remote destination, the path is used.

Several "network load balancing" products/methods have attempted to solve that problem by probing the local networks gateway routers in an attempt to determine the load of these gateways.

The problem with these solutions is that the load of the local gateway provides little to no information about the overall status of the network path that the local networks traffic is following. Issues that arise beyond the local gateway, within the local service providers network, or even within the 1st tier provider which provides transit for the local service provider, are not detected with this method, and thus do not provide true network redundancy and/or failover from one end of the communications session to the other.

Even if the device performs a per-packet test of the remote destination prior to sending the traffic, additional problems still exist. These problems include: slow response time for the initial packet, large memory requirements to cache routing information for routes which may never be used for long periods of time, and an overall increase in the costs associated with such a solution.

Summary

The XRoads Edge utilizes Vector Routing, which is a method for efficiently and accurately redirecting end-to-end communications sessions over the most appropriate network path when two or more diverse network paths are available without adding unneeded delay, or requiring large amounts of unused memory like many "network load balancing" devices require. This ensures a lower cost of total ownership and thus a higher ROI.

Vector Routing's diverse path selection is based on the continued measurement of multiple predefined remote nodes via two or more diverse network paths to a larger external network. This is accomplished via Multi-Path Probing, and Real World Monitoring. By monitoring these remote nodes and gathering specific data measurements via each diverse network path, the Vector Routing module (software code) running on the XRoads Edge can determine which diverse path traffic should sent.

If the Vector Routing module determines that all paths are operating normally, local network traffic is equally distributed across the multiple network paths. Load balancing can be applied via Vector Routing's Flexible Bandwidth Management. Using our flexible bandwidth manager, network administrators can determine what percentage of traffic they wish to forward over each of their diverse network paths. Unique to the XRoads Edge, these percentages can be applied per "critical network" (see Best Path Routing – White Paper).

In accord with the path selection by the Vector Routing module a DNS daemon running on the XRoads Edge can also be updated so that only those IP addresses of the network interfaces which a associated with the active network paths are provided in DNS responses to request made from external DNS clients.

The purpose of using diverse network path monitoring and route selection based on the analysis of the monitoring is to replace the existing complex and costly routing protocols used by many network routers today while still providing a more detailed status of the overall network path that many routing protocols do very well. At the same time, the reduced complexity ensures the lower overall cost of the Vector Routing enabled products.

Detailed Description

The following is a detailed description of the Vector Routing technology and its core functions:

Equal Cost Multi-Path Routing – This process is defined in RFC 2992 / 2391 / 1247 and various papers on equal cost multi-path routing. An example of how Vector Routing has implemented these standards is given below:

XRoads Edge Routing Table Entry

```
default gateway nexthop via x.x.x.x weight 1
nexthop via y.y.y.y weight 1

(the weighting determines how the load balancing is performed, when
both weights are equal, traffic is equally routed across both paths)
```

SNAT (Secure Source Network Address Translation) – When the Vector Routing module routes traffic through any secondary interfaces, that traffic must be NAT'd, meaning that the IP header information for those data packets must be translated from its original LAN address to the address on the WAN interface. The reason for doing this is to ensure that upon the response by the remote node, that the traffic returns to the originating WAN interface. If the traffic is not NAT'd the response would be direct back to the primary WAN interface, which may not be what is intended.

Dynamic Domain Name Service – The Vector Routing module incorporates a DNS daemon. This DNS daemon is dynamically updated with the latest IP address and active interface information.

The DNS daemon's purpose is to respond to remote clients inbound requests for IP address information based on the queried domain. By changing how responses to these requests are handled, the Vector Routing module can determine on which interface the inbound traffic is received from the remote client. This is a very effective method for load balancing and redirecting inbound traffic during a network outage.

In order for this method of "inbound routing" to work, the XRoads Edge, and the Vector Routing module, must be configured as the domain primary DNS server.

The method used determine how the DNS responds to remote clients is based on the interface address information, active path status (as determined by the Vector Routing module), and changes made to the dynamic DNS database based on those methods.

As the DNS responses are made to the remote clients, they have a limited TTL (time to live) value and include all of the IP addresses of the network interfaces which are associated with the active network paths. These addresses are provided in an order defined in RFC 1034 / 1035 / 1794 and BIND 4.9, September 1998. An example of how Vector Routing has implemented these standards is given below:

Equal Round-Robin Response

```
xroads  IN  A  10 10.0.0.100 300 1
xroads  IN  A  10 10.0.0.101 300 1
xroads  IN  A  10 10.0.0.101 300 1

(where 300 is the TTL specified in seconds)
```

Weighted Response

```
xroads  IN  A  10 10.0.0.100 1
xroads  IN  A  20 10.0.0.101 1  (where the ‘20’ is preferred interface)
xroads  IN  A  20 10.0.0.101 0  (where the ‘0’ represents a bad path)
```

```
BIND currently considers any TTL under 300 seconds as "irrational", and substitutes
in the value of 300 instead.  This greatly hampers the functionality of volatile zones.
In the fastest of all cases - a 0 TTL - information would be used once, and then thrown
away.  Presumably the new RR information could be calculated every 5 seconds, and the RRs
handed out with a TTL of 0.  It must be considered that one limitation of the speed of
a zone is going to be the ability of a machine to calculate new information fast enough.
```

Weighted Route Selection – As seen in the above example, weighted route selection is performed for both outgoing and incoming connections.

Outbound connections can be routed directly, or load balanced between two or more interfaces and their gateways. The method used by the Vector Routing module is to increase the weight of each default route, and thus increase the likelihood that the route will be used.

Inbound connections are similarly load balanced using the Vector Routing module’s dynamic DNS daemon. In this case the IP addresses provided in response to DNS requests are similarly weighted so that the more highly weighted addresses are provided as the first address in the response.

Routing Updates Based On Path Monitoring – The Vector Routing module performs network monitoring to multiple remote nodes in order to obtain network metrics which are then used to determine how the XRoads Edge routing table is updated. Several methods of network monitoring are employed to provide the most efficient route optimization possible. These methods include:

1) Core Monitoring – This method monitors core Internet routers (high traffic routers that are the crossroads of the Internet). Using modified ICMP packets, each response packet is then examined for latency, packet loss, calculated jitter and changes to the data sent within the ICMP query.

These core addresses are also updated from time to time by our remote global management systems (GMS), which is continuously monitoring and adjusting its addresses to reflect the ever changing Internet.

These updates are handled via XRoads' secure, automated client/server managed system. More information on the core XRoads routers can be found at www.internetxroadsreport.com.

2) Real World Monitoring – This method uses a predefined list of high traffic web servers which the Vector Routing module monitors via TCP SYN packets. Each response includes latency, packet loss, one-way packet loss, and calculated jitter information.

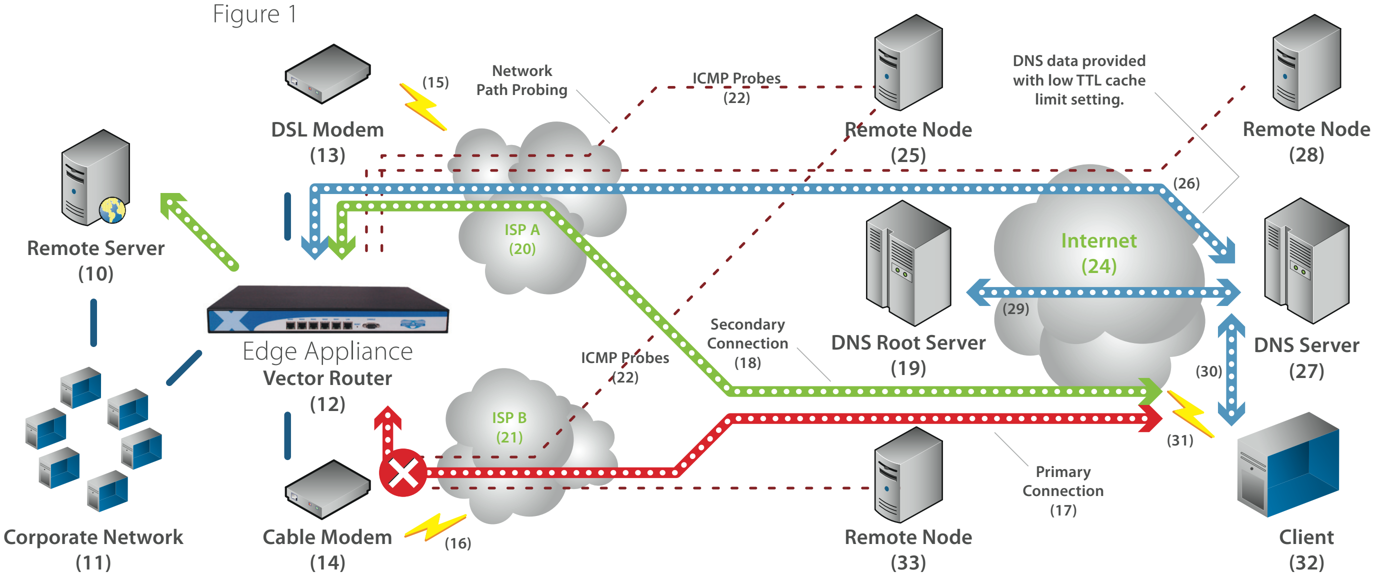
These addresses can be changed by the administrator, but must be available web servers or they will be seen by the Vector Routing module is unavailable, and that would effect how route changes are updated.

3) SuperNet Monitoring – This method provides incremental network specific routing based on probes to various addresses within each supernet. These probes are similar to the core monitoring and assist in providing more refined routing to specific areas of the Internet.

The Vector Routing module uses a complex algorithm with predefined weight information to determine whether a network path has been unacceptable and thus a routing change is required. Routing changes effect where outbound traffic is directed (interface/gateway) and how the DNS daemon responds to inbound DNS queries. For more information on the XRoads Edge and Vector Routing modules, please visit our website at www.xroadsnetworks.com, or call a sales representative at 888-9-XROADS.

Vector Routing Diagram

Figure 1



The diagram above provides an overview of how the Vector Routing module functions.

Reference is now made to Figure 1 that provides the general flow of vector routing. Vector Routing in this diagram consists of two diverse network paths connected to the XRoads Edge which is running the Vector Routing module (software code). The two networks paths consist of broadband connection devices 13 and 14, logical broadband data connections 16, their associated networks 20 and 21 and the larger external network (in this case the Internet) 24. To ensure that the local network 11 has the is optimally using the two diverse network paths, the XRoads Edge 12 sends probes via ICMP to multiple remote nodes 25, and 26 via both networks 20 and 21 to gather network measurements for those remote nodes via each network path, including latency, packet loss, and calculated jitter. These measurements are then stored within the XRoads Edge for later comparison and manipulation by the Vector Routing algorithm to determine whether each diverse network path is still within the acceptable range and whether the route for that path should remain in the apparatus' routing table and DNS daemon.

Assume that for some reason the network path through ISP B 21 is unable to provide connectivity from vector router 12 to the remote nodes 25 and 28. The vector router 12 would detect this via its probing and algorithm and change its routing table to reflect this change. The Vector Routing module within the XRoads Edge 12 would also set all IP addresses assigned to the Edge's network interface card of the associated non-acceptable network path within the DNS daemon to inactive thus causing DNS responses to no longer provide those IP addresses to DNS clients. During the next interval that the client 32 requests the DNS information for the remote server 10, the address has been updated and now the client 32 will use the secondary inbound connection 18 through ISP A 20 to maintain the communication session(s).

References Cited

Vector Routing White Paper, September 2001.
Vector Routing provisional patent filed June 2001.
RFC 2992, Analysis of an Equal-Cost Multi-Path, November 2000.
RFC 2391, Load Sharing using IP Network Address Translation, August 1998.
RFC 2136, Dynamic Updates in the Domain Name System, April 1997.
RFC 1794, DNS Support for Load Balancing, April 1995.
RFC 1322, A Unified Approach to Inter-Domain Routing, May 1992.
RFC 1247, OSPF Version 2, July 1991.
RFC 1034, Domain names - concepts and facilities, November 1987.
RFC 1035, Domain names - implementation and specification, November 1987.