

## Event title: 50MS PROTECTION FOR IP/LDP NETWORKS

Duration: 00:38:22

### Presenter

- Hannes Gredler

**Hannes Gredler:** ...Then number three you can see the [Jupital] router event propagation inside the modern [unclear]. So typically you have got three different elements, which is the embezzled [OS] off spur. There's the kernel in the middle and on the right hand side there's the routing software. Now let's see what happens if the link goes down. So the link driver reports that the link is down to the micro kernel. The micro kernel passes that information up to the kernel. The kernel notifies the routing software, first the IGP typically get them engaged and later on route resolution [unclear] is executed but route resolution tries to figure out has there been any dependency of, you know, [BGP] roles on, are any [BGP] routes affected by the IGP change. If there are, then obviously the change forwarding [state] needs to get downloaded again to the kernel, which further replicates the information down to the micro kernel. We call this long haul path, we call this the global repair path. So this is really control plain based conversion.

Since quite a while, we've figured there is also a more efficient sort of shortcut path directly embedded into the micro kernel, which is the local repair path, as illustrated by the light grey arrow. The reason why I first explained the terminology of this is that in the past many people used to try to optimise the control plane behaviour in order to seek fast restoration. What we try to always tell people that control plane conversions, control plane conversations and service restoration are most often separate things. Irrespective how you optimise the software down, there will be always be a short gap between the actual links break up until the system reacts and up until the software finally changes the forwarding state. Depending on you have multiple [TFEs], single [TFE] systems, depending

on the CPU horsepower of those systems, that gap which is here illustrated and that has the green ribbon, he is roughly between 100, 150 up to several hundreds of milliseconds.

So what local repair actually tries to do is to really make sure that the data plane catches up the forwarding path up until the control plane convergence. So what's wrong about existing [MPLF] transport behaviour. What is wrong is that well from three routes effectively is available for our [DRSUP] protocol only since 2001 and typically it requires a PE to be [at full] mesh for transport plane protection. Now the full mesh is exactly the problem here. So if you have let's say 200 PEs, then you end up with roughly 40k LSPs, not including all the detours and bypasses going through your network and that may be a burden to central core [alters].

Or worse, the RSVP protocol still has got this soft state tendency in here where we have constant control plane periodic refreshes to make sure that the LSP is still working. [LEP] So, this is a little bit different, using [sink trees] which tend to be more efficient. Also there is no soft refresh being built into the protocol and also it relies on TCP transport.

Now, so how do you scale RSVP based transport mesh. Well, there are two rules to make that happen, which is to deploy a so-called forwarding hierarchy. The idea is very simple. You basically have a PE to PE [LSP] and then you have the top level core router to core router LSP and what LSP hierarchy does, is it shares the control, it shares the top level LSPs among the PE to PE LSPs, such that you only have a limited amount of control plane and data plane state in the core.

The problem here is a bit what happens with note protection. So for example, consider like in the illustration, the [unclear] point of the top level mesh goes down and also provisioning, it may be an issue. So people typically come up with their home-grown provisioning roll outs in order to add routers to the full mesh. Don't get me wrong – we are not telling RSVP is the sort of evil, non-scalable protocol. No, that's what we're not saying. We're saying please make use of [LDP] for basically infrastructure of where it's just connectivity using [filler] transport path and use RSVP really where we must for tactical reasons, like traffic engineering, point-to-multi point LSPs. Also, whenever you seek a source routing paradigm, those are all natural places where RSVP is the best protocol around and

I'll give you later on in the presentation an example where we actually use a source routing LSP for that purpose of [implementing] the backup LSP coverage.

So what is this loop free [unclear] technology. Usually [loop free] [unclear] just adds basic reroute capability to [ISIS, OSPF and LDP] protocols as well as any other dependant protocols which use the path resolution, which inherit path resolution from those protocols. Normally the link state routing protocols only use the best path for forwarding purposes. Now what we're doing here is we have a non-best path for backup purposes. The only thing that we have to make sure is that the non-best path is loop free. How do we do that? Well we have the common shared link state database. So what we can do is just place the SPF routes not ourselves but rather at one of our neighbours and then we can explore the graph from our neighbour's point of view. How do we do that? You can see that here in the illustrations.

So usually you have the best path from router S to router D. If you sum up the metrics, you figure out that the green arrow is the shortest path from S to D. So what we're now doing with LSA is in addition to our main SPF, so router with the green circle is, we engaged so-called backup SPF operations, calculations sorry. That means we've figured out, let's say router N1, N2 do they also have [wireable] paths going to destination D and are those paths not going through ourselves. So this would be a condition of the looping path. So here N2 has also shortest path to D. The path does not loop through to ourselves so that makes it [illegible] for a backup path.

Configuration is pretty easy - so all of that complex calculation machinery we want to keep away from the operator. All you need to do is set a knob on the, on the IGP interface configuration, whether you seek link or no link protection. What does that mean? It means whenever, let's say, a link protection knob is set for interface, all the routes, all the primary routes, the point or that interface are automatically fast reroute protected. If for administrative reasons you do not wish to add certain interface to carry backup traffic at all, you can basically remove the interface from the set of illegible backup interfaces; because by default any interface that has an adjacency in the upstate may be used as viable backup. So again, if you don't want it, you can turn it off.

Next question that we typically get is well even running those extra SPF's, isn't that a harmful thing? Typically large core routers you have an order of 30, sometimes even 60 IGP adjacencies and computing a potential large cloud on behalf of the neighbours, isn't that routing excessive CPU usage? Well and the answer to that is yes and no. In previous journals releases we basically have a classical textbook style [unclear] for implementation, which typically has all the major elements and some product complexity of log and behaviour; and what we have done is we really have streamlined all the data structures and made them slick and lightweight, such that most operations that you have to do in that extra calculation are either a constant time or near constant time operations, which essentially boils down, which essentially results in [unclear] implementation which is pretty linear, which is a very nice scaling property.

Let me just share some test results. As a rule of thumb, on a high-end control plane CPU we have, it takes about one millisecond to obtain backup SPF calculation for 1,000 [unclear]. As you can see in the tracing outputs the main SPF roughly takes [one milliseconds] and backup draft processing for two neighbours takes two milliseconds. So one millisecond each. So what we think with those optimizations in place it is actually safe to deploy LSA protection, even in largest networks in the world.

Don't get too terrified by the following formula. So this is basically what the underlying RFC asks you to implement. So those are the so-called [unclear] for link protection, downstream path protection and note protection. All of that makes sure that the chosen backup path doesn't loop back through your server. The problem is RFCs are rarely a good implementation advice and the specific problem with RFC52/86 is that at the end the implementation instructions or the notes for the implementers means that you should really all cache the data. So for each calculation for the main SPF and for all your backup SPF calculation you need to really keep the calculation results in memory. Now that in turn results into almost a [state] explosion, as you can imagine this metric may be very big. So consider on the y-axis you have potentially a 1,000 routers and on the x-axis you have 30 neighbours that requires you to maintain a metrics of 30,000 values just to assess if a path is looping or not and we thought well that [state] explosion probably really will slow down our implementation.

So we have implemented the whole thing a little bit differently. So what we used to have is now so-called track list. The track list is basically a

sort of flight recorder. It is a list of important notes that while traversing through the SPF graph, we have been flying by and recording that note. So it helps us to pick good backup [unclear] tops. What are important notes? If ourselves of course were a loop protection, it's also [unclear] and the tail end of NPLSPs. So in the illustration you can see an example. So we start our backup SPF calculation from our bottom neighbour and because the shortest path is going through ourselves, so this is a classical example of looping. One big extra exploration arrives at ourselves. We have the sort of flags that in the link state database which says please put me on the track list. So that note is being accumulated up until we reach the destination.

So once we evaluate the destination, we take a look okay, what are the notes on the track list and once we figure out that for example ourselves is on the track list, we know that this looping path and we have now perfect assessment if the path is looping or not without building up those massive metrics of IGP calculation result.

Here you can see a typical UI output. So for example here you see the second view. So each view starts with the entry route. Here you can see there are two items on the track list, routers, [pro13] H, [pro13] F and as you can see on the command line output we are now on router [pro13] F and if [pro13] F is on the track list that means that this route is not a legible backup path because the path loops. This show ISIS or show SPF, backup SPF results command is also your first point for troubleshooting if you don't get the necessary coverage. So the implementation here will exactly tell you what is broken and why a certain backup path has not been used at all.

Backup - well [LSE] backup coverage has got a disadvantage which is it is not perfect. So typically we cannot protect 100% of the backup, 100% of our primary destination point: typically can only protect 65 to 85%. So this varies again depending on the [pathology]. What can you do in order to get you from 65 up to 90 or even 100%? Well, add links. However, well, we're happy to sell you interfaces but we do realise that there is in these times also a lot of CAPEX constraints.

So one idea of how to further augment the [pathology] is what can we just use, let's say, a tunnel link technology which is available in all of our routers and use that tunnel link technology as a way to extend the [one

hub] neighbourhood and this is exactly what we have been doing. So RSVP module has been extended with a knob saying backup and so you can use all of your standard RSVP machinery constraints, your [unclear], whatever. However, if the backup knob has been set then the interface is not available to other routing protocols. It's only available for backup. What is the advantage, you may ask yourself now.

Here in the illustration you have an example of the looping path. So our bottom neighbour is for a given destination D, sees his shortest path through ourselves. So we cannot use that neighbour as a viable backup. However, what we could do is setting up a unidirectional RSVP [LSP] from router S to the router in the red circle. The interesting property here is this is an RSVP tunnel. So it's source routed. So even if from a forwarding point of view we cross our immediate neighbour, the traffic, the backup traffic will be dropped off at the tail end of the MPLS LSD. So in other words the traffic will be dropped off deeper in the [pathology], which might give you a lot of better backup coverage. We do know from simulation that if you just provide a few static RSVP LSP to the [two hub] or let's say [three hub] [pathology] that is sufficient in order to get you to 100% coverage.

LDP integration. Actually I was a bit surprised when we had to do the LDP integration how easy it was. So typically the IGP and LDP they are very tightly coupled with each other. So LDP uses the IGP as a sort of resolution protocol in order to figure out what is the best, the shortest path for a given [FEC] and we just had to extend our LDP [unclear] such that it also inherits the backup information. In Juniper's implementation to backup information is indicated by a weight being set for forwarding [next hub]. A weight of one is the primary next hub and the weight bigger than one is usually the backup next hub. So LDP was just taking a look at all the next hubs from the IGP and simply cloning those next hubs, both in terms of forwarding information as well as debating. So that turned out to be very straightforward. Nothing fancy. [unclear] of course the same thing for transit LSPs.

So what about if we do LDP, LDP [unclear] route now by turning on IP fast rerouting the IGP. What does mean to, with regard to backup RSVP tunnels, because you know keep in mind the RSVP tunnel drops off the traffic not at the one hub neighbourhood, it drops off the traffic somewhere deeper in the [pathology] and we might not even know what is the LDP

findings for that tail end router. Well of course if you want to run LDP across this RSVP tunnel then that LDP tunnelling knob which that are the labels are properly learnt. So this is nothing different than today with LDP over RSVP.

Finally, let me share some test results with you. So we set up two test cases. Test case number one is trying to figure out the P-router behaviour. So what is the behaviour of, let's say, a core router that is deep within the core only, typically a BGP free core and just needs to change un-reach ability within a very short amount of time. So in order to also get some scaling figures and scaling problems here and maybe expose some scaling problems, we have added 1,000 [note] virtual grids as our router simulation [unclear] and added also a couple of 1,000 IGP prefixes.

Test traffic was being resolved, has been injected in the order of 3 megabits per second, such that you get a prefix granularity of at least 1 millisecond. So primary path was over R2 and the backup path was over R3. The test methodology has been just to shut down the laser at R2 and wait until the last IGP prefix or FCG converges and that figure is for 1,000 notes and the [pathology] is roughly 30 milliseconds. So we basically have [unclear] test runs and took the average.

That figure is not really yet a constant time operation. So let's say if you do, if you execute the same test with 10,000 SECs then it takes roughly in the order of 18 milliseconds. So you can see here there is some dependency of the amount of transport SECs which are in the transport mesh.

In addition to pure core router behaviour, we also wanted to assess what is the PE-router behaviour at the [ingress]. We wanted to figure out what is if a link goes down, a directly connected link goes down, how does that affect any service label. So from the large VPN that is being loaded on the PE-router. Similar test methodology we measured the time until the last VPN prefix conversion. In order to make it a little bit more, let's say, challenging we have ejected 400,000 IP layer free VPN routes and added the, change the [link allocation] on the router test that were labelled for prefix allocation so that each VPN prefix also has a unique label.

This has quite some impact on the way, how you have to engineer your forwarding planes because if you just have, let's say, 400,000 prefixes pointing to one, let's say, indirect next stop, which is typically a VGB speaker and you change just the forwarding information, then the only thing you need to do, given that you have a proper hierarchy] support in the forwarding plane is just to flip one pointer. So no big deal. However, if you have basically 400,000 prefixes pointing to 400,000 indirect next [tops], then you need to engineer your forwarding plane differently because otherwise you would be exposed to what we call a linear behaviour. The good news is we have done our homework right and the PFEs are on the MX. The theory is don't expose that behaviour. So we truly have constant time updates there. It doesn't matter whether you change forwarding state for 100,000, 400,000 routes or even 1 million routes, it's always in the order of 25 to 30 milliseconds.

There's also some future work that needs to be done with that technology. The technology is brand new. So we are the first vendor to ship and implement, a [unclear] implementation, both for IP as well as LDP that supports link protection, note protection as well as backup coverage extension. So that should actually be a fairly good feature to start with. What we have not added is, or invested is, in the work of micro loop mitigation. So that is a problem when you have let's say [pathologies] typically large rings or so and are the downstream note changes its forwarding state and the upstream note is not ready yet. So then you have two routers in the path with sometimes conflicting forwarding states such that you have micro loops.

So there is a couple of ideas how to fix those micro loops. The micro loops, let me also emphasise that it doesn't happen in all [pathologies]. So if you have a large meshing in your network, so a lot of redundant links, then micro loop is probably in most of the cases a non-issue. If it is, then a technology called [Ordered FIB] is a potential solution. Ordered FIB simply tries to synchronise somehow or order this phase, order the rerouting and the calculation of the rerouting of the routers along a forwarding path. In the IP paradigm typically all the routers do their thinking dependently and Ordered FIB tries to change that and make them to converge in an ordered fashion.

Also we have got enquiries from customers who'd love to see [SORT] support. Keep in mind [SORT] support has been available for RSVP for a

long time. The idea is that if you have let's say distinct IT supplements which run over the same layer one infrastructure, you want to tell and flag this property to the redundancy path calculation software such that the backup and the primary do not really run over the same physical link. Also an area of work is protect multitask traffic. As of today only newly [unclear] traffic is pre-route protected and right now we're also working on a technology called [EAGRIS LSP Protection] because IP fast reroutes for LDP as well as RSVP only protects [unclear] or transit LPS.

However, we figure out that especially for interconnection points like ABR or ASPRs LSP EAGRIS failure becomes an area of increasing concern and in order to protect the tail end of LSP we have bundled all of our efforts in a programme called End-to-End Service Restoration.

We're close to an end so let me briefly sum up things. Look for your alternates. What is it? It is a fast reroute technology for ISIS, LSPF and LDP. Juniper's implementation scale is well up to several thousand nodes. It is easy to provision – you just need to consider knob on the interface and there you are. You have incremental deployments which is a very nice property. So you can just turn it on, on the node and that node immediately has got an advantage as we do not rely on any protocol extensions. We are also not dependant on, let's say, other vendors supporting anything like that. So what LFAs are really doing is changing the box local behaviour. [unclear] and the implementation today for reasonable sized networks – [sub 15 millisecond] protection both for transport labels. On the [unclear] transit as well as for service labels on the [ingress]. Many IT firms pre-route LDP [faster] implementation suffer from less than 100% backup coverage. We are suggesting to use technology which is available and you may be familiar on the routers which is the technical use of RSVP tunnel to close the backup coverage gap and more important it's available now. So [RDPIS/AS] implementation has been available since April and [RDO] implementation will be available in June or September, which is either shipping or shipping next week. Let me turn back to Natasha, opening the phone lines for asking questions.

**Operator:** Thank you. We will now begin the question and answer session. If you would like to ask a question, please press \*1 on your telephone and wait for your name to be announced. If you wish to cancel the request please press the # key. That's \*1 on your telephone keypad if you'd like to

ask a question. There doesn't appear to be any questions. Please continue.

**Hannes Gredler:** Okay. Then let me thank you for your attention. I see there has been no and your interest. I see there has been lots of questions on the chat. Please [audio ends abruptly].