Future Internet Routing: Motivation and Design Issues

Routing im Internet der Zukunft: Hintergründe und Gestaltungsansätze

Michael Menth, Matthias Hartmann, Phuoc Tran-Gia, Dominik Klein, University of Würzburg

Summary The fast increase of the routing table size in the default-free zone (DFZ) is a major concern for the scalability of the Internet and a threat for its effective operation in the future. Proposals exist to modify the current routing architecture in order to decelerate the growth of the routing tables in the DFZ, but they are difficult to deploy. The locator/identifier split (Loc/ID) principle is significantly different from routing and addressing in today's Internet, but it is expected to improve routing scalability. We explain its basic idea, address interworking issues, point out design options, and review current implementation proposals. ►►► **Zusammenfassung** Das starke Wachstum der Routingtabellen im Kern des Internets, der sogenannten Default-Free Zone (DFZ), ist für die Skalierbarkeit des Internets besorgniserregend und eine Bedrohung für seinen effektiven Betrieb in der Zukunft. Es existieren zwar Vorschläge zur Modifizierung der momentanen Routing-Architektur, um das Wachstum der Routing-Tabellen zu verlangsamen, aber sie sind nur schwer in die Praxis umzusetzen. Das Locator/Identifier Split (Loc/ID) Prinzip unterscheidet sich signifikant von der Adressierung und dem Routing im heutigen Internet, aber es soll die Skalierbarkeit des Routings deutlich verbessern. Die Idee des Loc/ID wird erklärt, auf Fragen des Interworkings wird eingegangen, Gestaltungsoptionen werden aufgezeigt, und es wird ein Überblick über aktuelle Implementierungsvorschläge gegeben.

KEYWORDS C.2.1 [Computer Systems Organization: Computer-Communication Networks: Network Architecture and Design] network communications; C.2.2 [Computer Systems Organization: Computer-Communication Networks: Network Protocols] routing protocols; C.2.6 [Computer Systems Organization: Computer-Communication Networks: Internetworking] standards

1 Introduction

The Internet is the nervous system of today's modern society and vital for its operation and evolution. However, it was never planned as such but rather evolved from a small ARPA testbed between a few research sites to an ever growing interconnection of all kinds of networks. Its simplicity and initial successful services like email, file transfer, and the world wide web fostered its fast deployment before highly complex services like peer-to-peer (P2P) applications, service-oriented architectures (SOA), or content distribution networks (CDN) entered the scene. The original Internet was designed for interconnection of a manageable number of hosts that were attached to a growing but conceptually rather static network topology. This is no longer the case since today all sorts of devices are interconnected over the Internet protocol (IP), many Internet service providers (ISPs) compete for customers causing modifications of the logical topology with each customer change, and mobile devices require fast support by local networks when hopping from one to another. To meet the changing requirements, several mechanisms have been changed or added, e. g., the domain name system (DNS) has been introduced to decouple names from addressing, the border gateway protocol (BGP) to make routing more scalable, the transmission control protocol (TCP) to avoid a congestion collapse, and mobile IP to accommodate nomadic users. The Internet was always subject to changes that were pushed by the insight that its future operation was at risk or at least its further expansion was hampered. An excellent overview is given in [6].

Currently, we witness movements towards fundamental changes in the Internet, at least for newly deployed infrastructure. In spite of BGP, interdomain routing does not scale anymore. Too much information needs to be exchanged between

border routers, their routing tables become larger and larger, and it is not clear whether router technology can keep pace with the growth of the routing tables and increased traffic volumes in the future at reasonable costs. Therefore, operators and router vendors have already recognized the need for a new change of the interdomain routing system and the Routing Research Group (RRG) of the Internet Research Task Force (IRTF) [9] provides a forum to discuss problems and proposals.

This paper gives an introduction into the problems and summarizes some of the current ideas to overcome them. Section 2 briefly reviews routing in today's Internet and Section 3 explains why it does not scale. Section 4 describes ideas to decelerate the growth of the routing table sizes in today's Internet architecture. Section 5 presents the locator/identifier split (Loc/ID), interworking issues, design options and first proposals. Finally, Section 6 gives a short conclusion.

2 Routing in Today's Internet

The Internet is an interconnection of multiple autonomous systems (ASes) using IP as the common base to exchange messages. IP networks use destination-based forwarding, routers look up the next hop for a packet in their forwarding information bases (FIBs) which are derived from their routing tables. The FIB entries consist of address prefixes and next hops. The longest prefix match for a destination address determines the interface over which the packet is transmitted. A default route can be provided that is taken when no matching prefix is found.

Each AS may use its own method to generate entries in the routing tables. Basically, they assign administrative costs to all links within the AS and forward the traffic along least-cost paths. This is mostly realized by distributed routing protocols like OSPF or IS-IS. For larger ASes, a subdivision of the network into several routing areas helps to manage the routing complexity and to keep intra-domain routing scalable.

To reach nodes in other ASes, inter-domain routing uses the border gateway protocol (BGP). Each BGP router tells its neighbors which destination prefixes can be reached over its own network and also provides a list of ASes that need to be traversed on the path towards the destination AS. Therefore, BGP is called a path vector protocol. Routers in edge networks usually have a manageable number of prefixes in their routing tables and packets to unknown destinations are forwarded to a default router. However, BGP routers in the core of the Internet do not have default routes. They constitute the so-called default-free zone (DFZ) of the Internet. The DFZ routers need an entry for each prefix that should be reachable in the Internet and, as a consequence, their routing table size increases with the number of reachable prefixes.

The early Internet consisted of a relatively small number of ASes and customer edge networks that were only sparsely connected. Each AS was assigned a rather large chunk of the available IPv4 class A, B, or C addresses. The corresponding class prefixes were announced individually into the interdomain routing system. It soon became evident that this practice leads to address exhaustion because most of the assigned class A and B address space is never used. The address space of a class C network is often too small for a company or institution such that they need several of them. This heavily burdens the routing tables because each of the many (about 2 millions) and long class C prefixes requires a separate entry in the routing tables of the DFZ. The problem was alleviated by the introduction of classless interdomain routing (CIDR) in 1993 which removed the strong classification into class A, B, and C addresses. CIDR allows IP address assignment on a more fine grained level, i. e., the address space of a single class A or B address can be assigned in small portions to various customers. In addition, prefix aggregation is possible, i. e., ISPs announce only one short prefix to BGP instead of multiple longer prefixes when these prefixes cover a contiguous address block.

3 The Scalability Problem

Currently, we observe that the number of entries in the routing tables of the DFZ is increasing at an alarming rate. Figure 1 shows that the growth rate is quadratic or even exponential. To cope with larger routing tables, routers need to be more powerful. Advances in routing technology might be able to compensate the increased routing tables, but this can only be achieved at disproportionately high costs.

When the Internet finally runs out of IPv4 addresses, the introduction of IPv6 eventually brings

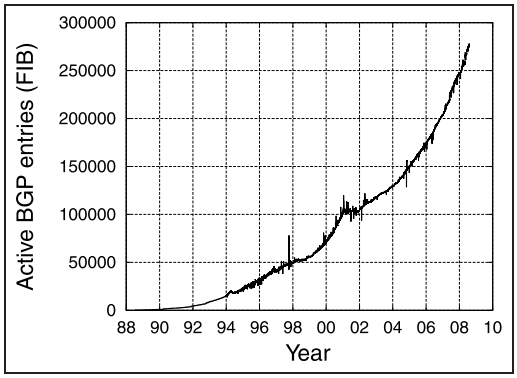

Figure 1 Growth of the routing tables in the DFZ. [http://bgp.potaroo.net/as2.0/bgp-active.txt].

an almost unlimited number of IP addresses. This solves the problem of address depletion, but routing tables are going to grow even more because the vast amount of available IPv6 addresses require even more prefixes to be announced in the DFZ. Experts discussed and analyzed this problem at the IAB Workshop on Routing and Addressing [16] and came to the following conclusions. The main causes for the current growth rate of the routing tables in the DFZ are the use of provider-independent addresses, multihoming, traffic engineering for edge networks, and countermeasures against prefix hijacking. In the following, we explain these issues in more detail.

3.1 Provider-independent addressing

The IP address space can belong to providers or to customers. In the first case, the addresses are called provider-aggregatable (PA). The provider rents subspace, i. e. prefixes, to customers for the duration of their contract, but remains the owner of the IP addresses. When the contract is over, the provider rents the prefixes to other customers. This has no impact on interdomain routing because packets to theses prefixes are still routed into the same AS. PA addresses limit BGP change rates and the fragmentation of the address space, i. e., they preserve the aggregation of IP addresses such that short prefixes continue to be announced through BGP. However, when a company using PA address space changes its provider, all computers and devices in that company must be renumbered to the address subspace of its new provider. This is a time-consuming and expensive task. Hence, companies prefer to obtain their own address space, i. e. so-called provider-independent (PI) addresses. This allows them to easily change providers without renumbering. For the global routing system, a provider change for PI addresses means a BGP update. In addition, the moved prefix possibly cannot be aggregated with other addresses in the AS of the new provider and needs to be announced separately to BGP which increases the number of entries in the interdomain routing tables.

3.2 Multihoming for increased reliability

Customers like to be connected to more than one ISP to increase the reliability of their Internet connection. In case that the connection to one ISP fails, their traffic can be switched to the other ISPs. This requires that different paths towards these prefixes need to be announced to BGP to make the customer network reachable over multiple providers. This leads to several entries for a single prefix in the BGP routing tables.

3.3 Multihoming for traffic engineering

Customers with PI-addresses may wish to use different providers for service differentiation. They subdivide their address space into smaller chunks each of them serving a different purpose and being attached to a different provider. In addition, multihoming may be used for load balancing purposes. As a result, the address space is split and several longer prefixes are announced via different providers to BGP.

3.4 Countermeasure against prefix hijacking

IP's destination-based forwarding uses the longest prefix match principle, i. e., when several prefixes in the FIB match the destination address of a packet, the packet follows the route specified for the longest prefix. Malicious ASes may inject prefixes they do not own. If they are more specific than other prefixes, they attract the traffic. To avoid this risk, ASes like to announce the longest possible prefixes which are 24 bits long at least for most important services such as DNS. This also leads to an increase of the routing table sizes in the DFZ.

4 Tuning BGP and Simple Overlays

The size of the Internet, its inevitable changes, and failures lead to a large rate of BGP update messages stressing router CPUs. As this rate is increasing, many proposals have been made to modify BGP in order to reduce it [1; 3; 19; 22].

The current BGP system cannot be adjusted to be truly scalable in terms of routing table sizes. Routing schemes with preferably logarithmic scalability in network size are desired to allow for almost unlimited future growth of the global Internet and a lot of research has been done on that topic. Unfortunately, it has been shown that logarithmic scaling on Internet-like topologies is impossible in the presence of topology dynamics and/or topology-independent addressing [11].

Any routing mechanism replacing BGP, possibly on a flag day, is almost impossible to deploy in the widely distributed Internet. Therefore overlay architectures have been proposed which leave the current BGP system in place, but take most of the load away from it. They shrink the routing tables to make interdomain routing more scalable. We explain two fundamentally different ideas for that purpose.

4.1 Aggregation proxies

With aggregation proxies, ISPs announce some of their supported prefixes not via BGP, but only to special aggregation proxies. An aggregation proxy receives many long prefixes and announces aggregated and shorter prefixes to BGP. Packets in the DFZ are carried to this aggregation proxy, which tunnels them to the ISPs that announced the longer prefix to the aggregation proxy. The aggregation proxy in Fig. 2 receives the long prefixes X.Y.0/24, X.Y.1/24, X.Y.2/24, and X.Y.3/24 and announces the prefix X.Y.0/22 to BGP. Therefore, it receives the traffic addressed to these destinations and forwards it to them over a direct tunnel to the border router of the corresponding networks. In our

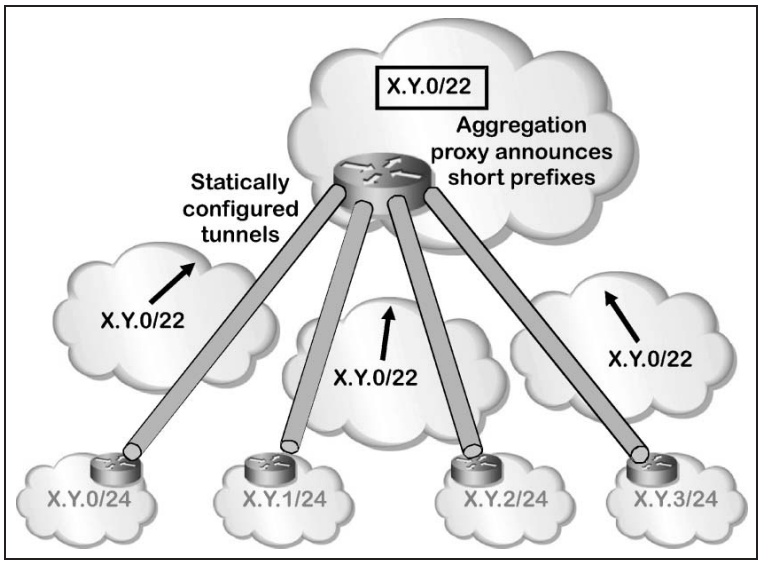

Figure 2 The aggregation proxy announces a short prefix instead of many long prefixes. Packets addressed to the long prefixes are routable in the DFZ, but are forwarded to the aggregation proxy which tunnels them to their destination network.

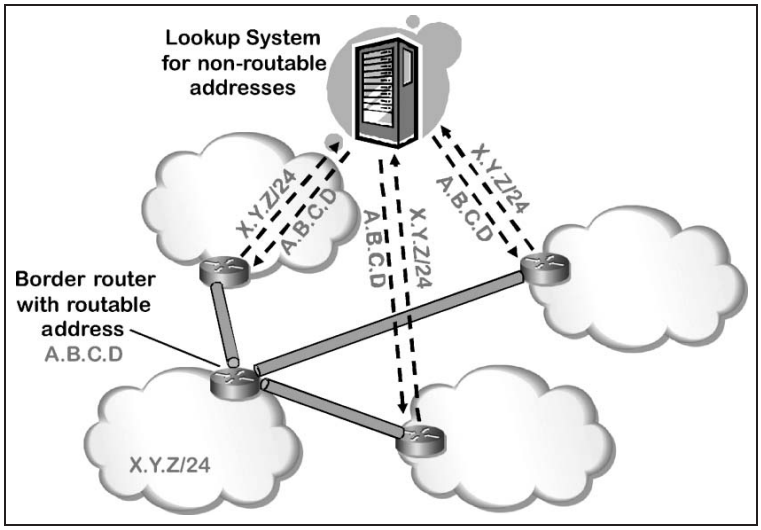

Figure 3 Some long prefixes (e. g. X.Y.Z/24) are not announced to BGP. Therefore, they are not routable in the DFZ. The lookup system provides a router with a routable address for them in the destination AS. Then, packets with non-routable addresses can be tunneled to and decapsulated by such a router, and forwarded from there to their destination via intradomain routing.

simple example, the routing table size in the DFZ is reduced by 3 entries. However, packets in the DFZ destined towards the long prefixes are always carried via an aggregation proxy. This kind of triangle routing possibly leads to longer paths compared to the shortest AS-path or the normal BGP path. Thus, path prolongation is the cost of aggregation proxies. Furthermore, networks should be customers of aggregation proxies or peering partners. Hence, this concept also requires substantial economic support for effective deployment. Several aggregation proxies may exist for the same prefix. The Core Router-Integrated Overlay (CRIO) implements this concept and [25] gives insights into tradeoffs like routing table size reduction vs. path length prolongation and many more.

4.2 Lookup system for nonroutable prefixes

Another concept is to retain long prefixes from BGP and to record them in a DNS-like lookup system together with a router having a routable address and being part of the destination AS. As a result, the long prefixes are not routable in the DFZ, but the lookup system knows a router from which corresponding packets can be forwarded without interdomain routing information. This concept is depicted in Fig. 3. If a router in the DFZ does not find an entry for the destination address of a packet in its routing table, it queries the lookup system for a tunnel endpoint into the destination AS and forwards the packet over that tunnel. After decapsulation in the destination AS, the packet can be forwarded via intradomain routing. The tunneling route reduction protocol (TRRP) [7] implements this idea. It requires the introduction of a mapping service, and the DFZ routers must be changed to perform the lookup and tunneling. The solutions in Section 5 are similar, but they do not require an upgrade of the DFZ routers.

4.3 Discussion

The presented methods leave today's routing system and in particular the meaning of the IP addresses basically as they are, but they still require changes to the Internet that are only hard to deploy.

5 The Locator/Identifier Split

The locator/identifier split (Loc/ID) concept has been implemented by various proposals using different nomenclature and technical realizations. We first give some motivation for Loc/ID. Then we explain the actual concept in a general way and discuss interworking issues and design options. This is an original contribution of this paper. It shows that many interworking and design

options are rather a property of Loc/ID itself than a property of the specific implementation. Finally, we look at implementation details of current proposals.

5.1 Motivation

Experts analyzed the current Internet structure and many agreed that a complete redesign of the routing architecture may be needed [16]. The current approach cannot scale, because the IP addresses have two different functions. They determine the position of a node within the Internet topology, i. e., they serve as *locators* for routing purposes. In addition, they also serve as node *identifiers*. The scalability of interdomain routing is based on hierarchical structures since an ISP can aggregate many long prefixes and announce them as a single short prefix. To take advantage of that principle, locators must be assigned according to the topology and should change only if the topology changes. In contrast, end users see IP addresses as identifiers and prefer to keep them if they change providers. This observation proposes a locator/ identifier split (Loc/ID), such that locators can be easily changed without renumbering the devices in the network of a customer when he moves to a different provider. Identifiers should be used only to give names to devices [13]. The actual split is often made at the network edges, and hence, identifiers are also used for routing in the local routing domain. Only few proposals use local locators for that purpose [15].

5.2 Basic Idea

Loc/ID is a two-level routing architecture. Figure 4 shows that there is only a single upper layer domain (global routing domain), but there are many local routing domains (intradomains) in the lower layer. Routing in the upper layer is based on locators while routing within the lower layer is based on identifiers. As long as communicating entities are in the same local routing domain, only their identifiers are needed to exchange messages. Communication between entities in different local routing domains is more complex. Local routing domains have gateways towards the global routing domain. These gateways are part of the global routing domain and have own locators. A packet with a destination identifier outside the local routing domain is forwarded to such a gateway. The source gateway determines the locator of the destination gateway by some mapping service and adds this locator to the packet. Then, the packet is carried through the global routing domain according to the locator to the destination gateway. The destination gateway strips off the locator and the packet is forwarded according to its destination identifier.

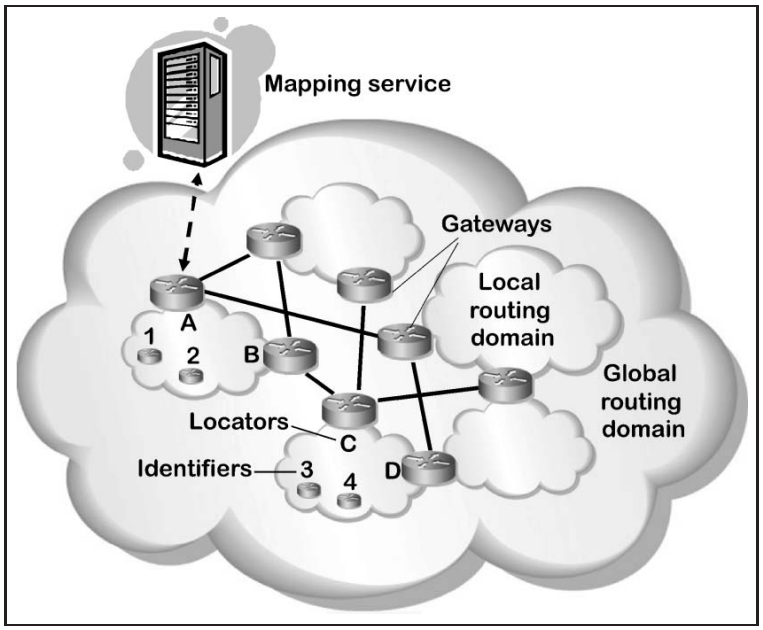

Figure 4 Loc/ID is a two-level architecture. Packets are forwarded within a local routing domain according to their identifier. Within the global routing domain they are forwarded according to a locator provided by the gateway when leaving the local routing domain.

5.3 Provider Change by Customer

When a customer changes its provider and connects its network to a different gateway, it can keep all the identifiers in its network as they are and there is no need for renumbering. Only the mapping service needs to be updated about the locator change.

5.4 Mobility

Mobility requires that nodes can move from one network to another while being reachable and without changing IP addresses on the transport layer. The latter is important for the maintenance of TCP connections. In theory, this could be achieved with Loc/ID, but this implies two major challenges. First, updates of the mapping system must be fast enough, and/or the gateways must implement handover functionality. Second, the IP addresses of the visiting nodes must be routable in the visited domain. The identifier space usually reflects some structure of the local routing domain to improve the routing scalability. Therefore, it is rather hard for the routing in the visited local routing domain to quickly integrate the address of the visiting node. This is due to the fact that identifiers are also used as local locators. Currently, a major opinion is that other mechanisms like mobile IP should be used for mobility support, but

certainly tradeoffs need to be considered.

5.5 Traffic Engineering

Apart from the improved flexibility, Loc/ID also opens new possibilities for interdomain traffic engineering [20]. Local routing domains can be multihomed and have several gateways and locators. This entails degrees of freedom for load balancing supported by the mapping service.

5.6 Interworking with the legacy Internet

We assume that the current legacy Internet evolves to the future upper layer domain, but it is certainly also possible to think of the legacy Internet being one of many local routing domains in the future Internet. For interworking with the legacy Internet, identifiers and locators should be IPv4 or IPv6 addresses. Here, we do not distinguish between IPv4 and IPv6 and think of one common IP address format. The address spaces of locators (including the legacy Internet) and identifiers must be disjoint to avoid ambiguities between devices in the legacy and the future Internet.

A simple solution uses proxy gateways in the upper level domain, i. e., in the legacy Internet. They announce all identifier prefixes into BGP. Therefore, the identifier address space should be aggregatable to avoid additional significant growth of the routing tables. When a legacy node sends a packet to a node in some local routing domain, it can use the identifier of that node as destination address because then the packet is carried to a proxy gateway. The proxy gateway requests the locator for the destination identifier from the mapping service and adds the destination locator to the packet. Hence, the proxy gateway performs the same operation as a common source gateway. Then the packet is eventually delivered to the correct destination gateway where the destination locator is removed. From there the packet is forwarded to its destination within the local routing domain using only the destination identifier. Communication from a node in the new part of the future Internet to the legacy part is even simpler. A packet from a local routing domain is addressed to a node in the legacy Internet. As the destination address is not part of the local routing domain, the packet is forwarded to a source gateway. The gateway realizes that the address is part of the legacy Internet and, therefore, the packet can be forwarded in the upper layer domain without any modification. Proxy gateways in this context and aggregation proxies used for BGP tuning look similar, but there is a subtle difference. While proxy gateways can basically attract all traffic addressed to globally non-routable identifiers, aggregation proxies attract only traffic whose prefix lies within their aggregation ranges.

The other solution for interworking with legacy networks is network address translation (NAT). A node of a local routing domain in the future Internet can send a packet addressed to some node in the legacy Internet. The gateway detects this and performs NAT. More specifically, it translates its source identifier to a locator address belonging to the gateway and forwards the packet to the destination in the legacy Internet. Potential answers are returned to the gateway which translates its own locator in the address field of the packet to the initial source identifier and forwards the packet to the local routing domain. Communication in the reverse direction can be more complex, depending on the actual implementation.

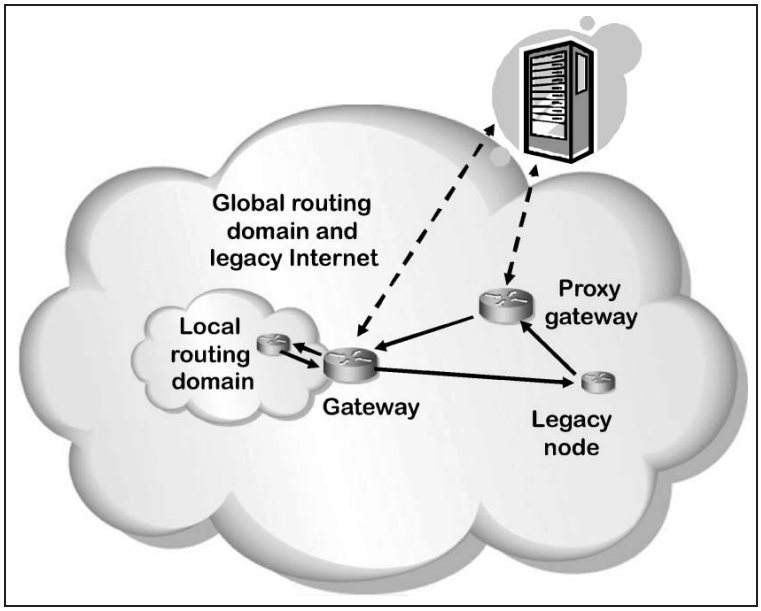

Figure 5 Nodes in a local routing domain communicate directly with legacy nodes as the gateway just forwards the packets. In the reverse direction proxy gateways attract traffic with destination in local routing domains and add locators to guarantee that they reach the correct local routing domain.

5.7 Design options for locator/identifier transport

The general idea does not require a special technique for the source gateway to add a locator to a packet. Two major alternatives exist: encapsulation and address rewriting.

In case of encapsulation, the source gateway tunnels the packet addressed to the destination identifier in another packet addressed to the destination gateway locator,

and the destination gateway decapsulates the packet. This is known as the map-and-encap paradigm [16]. Although the overlay solutions in Section also use tunneling, they are significantly different as they do not implement the Loc/ID approach.

In case of address rewriting, there is a 1 : 1 mapping from local addresses (identifiers) to global addresses (locators). Gateways replace the local source and destination addresses of outgoing packets by their global addresses, and for incoming packets the reverse operation is performed. Note that this kind of address rewriting is different from today's network address translation (NAT) for private networks. It is stateless and, therefore, relatively simple. In particular, nodes within a local routing domain are reachable from outside by their global addresses. A provider change implies a locator change which results in a modified global address, but renumbering of the nodes in the affected local routing domain is not needed.

Packet en- and decapsulation costs CPU cycles on routers and encapsulation can cause problems with maximum transfer units (MTU) and/or packet fragmentation. Address rewriting can also be expensive. Depending on the implementation, an additional header is added to record the identifiers. In IPv4, additional headers cause packets leaving the fast-path of a router and being processed by the router's CPU which can severely impact its performance.

5.8 Design options for the mapping service

There are even more design options to implement the mapping service. It can be a single server or a server overlay where each server keeps the locator-identifier mapping for the entire identifier address space [10]. However, this information may also be partitioned among many servers taking advantage of some hierarchical structure in the identifier address space to facilitate effective information retrieving [2; 5]. As an alternative, a distributed hash table may be used for that purpose [14].

It is obvious that communication overhead between the source gateways and the global mapping service is prohibitive when locators are queried for each and every identifier. The natural solution to that problem is a local cache from where queries can be immediately answered without consulting the mapping service [8]. This not only saves communication overhead, it is also faster than a remote lookup. There is the question what to do with packets while waiting for locators when a cache miss occurs. Packets are either stored and delayed, or they are simply dropped. To make this a rare event, the cache size must be large enough. In the extreme case, the local cache is a copy of the mapping information for the entire identifier address space [12]. If a cache miss occurs, it usually hits the first packets of a communication. Their number is relatively small, but it adds delay when some of them carry important signaling information such as a TCP SYN. Therefore, it makes possibly sense to extend the functionality of the mapping service with packet forwarding capabilities [5]. Initial packets causing cache misses are encapsulated and sent to the mapping service. The mapping service retrieves the locator, and sends it to the source gateway, while at the same time it acts as a proxy for the source gateway by adding the locator to the initial packets and forwarding them to the destination gateway.

The mapping service is a vital element of Loc/ID which needs to be up to date to achieve global reachability for identifiers. This becomes more difficult with caches because the information stored in caches may be obsolete. It raises discussions about push and pull architectures, i. e., either the mapping service triggers the update of local caches or the gateways are responsible for that action. Hybrid mechanisms are possible where the mapping information is pushed to a (probably large) set of cache servers from where it can be pulled quickly to any point in the Internet.

5.9 Proposals implementing Loc/ID

The Locator/Identifier Separation Protocol (LISP) [17] is Cisco's solution and the most advanced implementation of Loc/ID. It uses the map-and-encap paradigm to add a locator to a packet. Locators are called routing locators (RLOCs), identifiers are called endpoint identifiers (EIDs), and the source and destination gateways are called ingress and egress tunnel routers (ITR, ETR). Various concepts exist for the implementation of its mapping service. IVIP [24] is very similar to LISP with extra features for the handling of mobile users. APT [10] adds features to the mapping service in order to provide protection mechanisms in case of network failures.

Six/One Router [23] is a different proposal and uses address translation instead of tunneling. The gateways are called Six/One routers. The Node Identity Internetworking Architecture [21] is as well based on Loc/ID. It integrates ideas from the host identity protocol (HIP) [18] to achieve increased security and provides improved mobility support.

6 Conclusion

The scalability of the Internet is at risk since a major and intensifying growth of the interdomain routing tables has been observed. Several repairs have been proposed for BGP to reduce the rate of its update messages, and some others try to decelerate the growth of the routing tables, but they require substantial change of today's interdomain routing architecture. The locator/identifier split (Loc/ID) concept also imposes changes, but often only to a very limited number of border gateways. It is a promising candidate to effectively cope with the scala-

bility problem. Loc/ID comes with many design options regarding its operation, mapping service, and interworking. They provide degrees of freedom leading also to performance tradeoffs that need to be studied. First proposals have been presented implementing the Loc/ID concept.

Our view in this paper has been from the perspective of increasing routing table sizes. However, there are other issues to respect when redesigning the Internet [4]. Apart from improved routing scalability and Loc/ID split, a new Internet routing architecture must provide scalable support for traffic engineering, multi-homing, and mobility. Renumbering of an entire AS should be simplified, routing quality and security are very important. Last but not least, deployability of a solution is a prerequisite for its adoption in practice. Solutions similar to the interworking of IPv6 and IPv4 may be found to facilitate the interworking of a future Internet routing solution with the legacy Internet.

The current proposals are not the end of future Internet routing, they are rather the beginning of a topic which has gained so much importance that it is now driven by major players in the Internet.

Acknowledgements

The authors would like to thank Christian Vogt and Stefan Mühleck for fruitful and stimulating discussions.

References

[1] Y. Afek *et al.* Improved BGP Convergence via Ghost Flushing. In: *IEEE Journal on Selected Areas in Communications*, vol. 22, no. 10, pp. 1933–1948, 2004.

[2] S. Brim *et al.* LISP-CONS: A Content distribution Overlay Network Service for LISP. http://tools.ietf.org/id/draft-meyer-lisp-cons-04.txt, Apr. 2008.

[3] J. Chandrashekar *et al.* Limiting Path Exploration in BGP. In: *IEEE Infocom*, 2005.

[4] T. Li (Ed.). Design Goals for Scalable Internet Routing. http://tools.ietf.org/id/draft-irtf-rrg-design-goals-01.txt, July 2007.

[5] D. Farinacci *et al.* LISP Alternative Topology (LISP-ALT). http://tools.ietf.org/id/draft-fuller-lisp-alt-02.txt, Apr. 2008.

[6] M. Handley. Why the Internet Only Just Works. In: *British Telecom Technology Journal*, vol. 24, no. 3, July 2006.

[7] W. Herrin. Tunneling Route Reduction Protocol (TRRP). http://bill.herrin.us/network/trrp.html, 2008.

[8] L. Iannone and O. Bonaventure. On the Cost of Caching Locator/ID Mappings. In: *ACM Conf. on Emerging Network Experiment and Technology (CoNEXT, formerly QoFIS, NGC, MIPS)*, Dec. 2007.

[9] Internet Research Task Force (IRTF). Routing Research Group (RRG). http://www.irtf.org/charter?gtype=rg&group=rrg, 2008.

[10] D. Jen *et al.* APT: A Practical Transit Mapping Service. http://tools.ietf.org/id/draft-jen-apt-01.txt, Nov. 2007.

[11] D. Krioukov *et al.* On Compact Routing for the Internet. In: *ACM SIGCOMM Computer Communications Review*, vol. 37, no. 3, July 2007, fIR.

[12] E. Lear. NERD: A Not-so-novel EID to RLOC Database. http://tools.ietf.org/id/draft-lear-lisp-nerd-04.txt, Apr. 2008.

[13] D. Massey *et al.* A Scalable Routing System Design for Future Internet. In: *ACM Int'l Workshop on IPv6 and the Future of the Internet (IPv6)*, Kyoto, Japan, Aug. 2007.

[14] L. Mathy *et al.* LISP-DHT: Towards a DHT to Map Identifiers onto Locators. http://inl.info.ucl.ac.be/system/files/draft-mathy-lisp-dht-00.txt, Feb. 2008.

[15] M. Menth *et al.* Global Locator, Local Locator, and Identifier Split (GLI-Split). Under submission.

[16] D. Meyer *et al.* RFC4984: Report from the IAB Workshop on Routing and Addressing. Sep. 2007.

[17] D. Meyer. The Locator Identifier Separation Protocol (LISP). In: *The Internet Protocol Journal*, vol. 11, no. 1, pp. 23–36, Mar. 2008.

[18] R. Moskowitz and P. Nikander. RFC4423: Host Identity Protocol (HIP) Architecture. May 2006.

[19] D. Pei *et al.* BGP-RCN: Improving BGP Convergence through Root Cause Notification. In: *Computer Networks*, vol. 48, no. 2, 2005.

[20] B. Quoitin *et al.* Evaluating the Benefits of the Locator/Identifier Separation. In: *ACM Int'l Workshop on Mobility in the Evolving Internet Architecture (MobiArch)*, Kyoto, Japan, Aug. 2007.

[21] S. Schuetz *et al.* Node Identity Internetworking Architecture. http://tools.ietf.org/id/draft-schuetz-nid-arch-00.txt, Sep. 2007.

[22] W. Sun *et al.* Differentiated BGP Update Processing for Improved Routing Convergence. In: *IEEE Int'l Conf. on Network Protocols (ICNP)*, 2006.

[23] C. Vogt. Six/One Router: A Scalable and Backwards Compatible Solution for Provider-Independent Addressing. In: *ACM Int'l Workshop on Mobility in the Evolving Internet Architecture (MobiArch)*, Seattle, WA, USA, Aug. 2008.

[24] R. Whittle. IVIP – A New Routing and Addressing Architecture for the Internet. www.firstpr.com.au/ip/ivip/, 2008.

[25] X. Zhang *et al.* Scaling IP Routing with the Core Router-Integrated Overlay. In: *IEEE Int'l Conf. on Network Protocols (ICNP)*, 2006.


1


2


3


4

1 Dr. rer. nat. Michael Menth is leading a group on "Next Generation Networks". The focus of the group is on future Internet design, performance analysis, planning and optimization of communication networks, and in particular resilience issues. Matthias

Hartmann and Dominik Klein are members of that group and pursuing their PhD and Diploma degree. Prof. Phuoc Tran-Gia is director of the institute. He is member of different consortia concerning Future Internet design.
Address: Department of Distributed Systems, Institute of Computer Science, University of Würzburg, Am Hubland, 97074 Würzburg, Germany, Tel.: +49-931-888-6644, E-Mail: menth@informatik.uni-wuerzburg.de

2 Dipl.-Inform. Matthias Hartmann
Address: Department of Distributed Systems, Institute of Computer Science, University of Würzburg, Am Hubland, 97074 Würzburg, Germany, Tel.: +49-931-888-6642, E-Mail: hartmann@informatik.uni-wuerzburg.de

3 Prof. Dr. Phuoc Tran-Gia
Address: Department of Distributed Systems, Institute of Computer Science, University of Würzburg, Am Hubland, 97074 Würzburg, Germany, Tel.: +49-931-888-6631, E-Mail: trangia@informatik.uni-wuerzburg.de

4 Dominik Klein
Address: Department of Distributed Systems, Institute of Computer Science, University of Würzburg, Am Hubland, 97074 Würzburg, Germany, E-Mail: dklein@informatik.uni-wuerzburg.de