UDP PING: a dedicated tool for improving measurements of the Internet topology
Fabien Tarissan, Elie Rotenberg, Matthieu Latapy, Christophe Crespelle

To cite this version:

HAL Id: hal-01208361
https://hal.archives-ouvertes.fr/hal-01208361v2
Submitted on 19 Oct 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
This problem is nowadays widely acknowledged: as network measurements rely on intricate procedures which give limited views of the network, the obtained views may have properties induced by the measurement procedure, and thus differ significantly from the ones of the studied network. However, only very limited and unsatisfactory solutions exist to cope with this problem: despite a few exceptions [5], [6], [7], most results on this topic are negative and show that the observed properties should not be trusted [2], [3], [5]. One approach could be to derive the properties of the network of interest from the observed ones (rather than simply consider that the observed ones are true), but this turns out to be very difficult. Another approach could be to conduct larger measurements, and this is indeed done (Caida [14], Dimes [15]). However, the network to measure generally evolves faster than our ability to measure it, and thus such approaches, even though they provide interesting data, do not solve the problem. 

II. OBJECTIVE

The degree distribution of a network topology (i.e. for each integer \(k\) the fraction \(p_k\) of nodes with \(k\) links) is one of its most basic properties. It has a strong impact on
key features of the network like efficiency of protocols and robustness to failures and attacks. Until the seminal papers [16], [12] in the late 90's, it was commonly assumed that the degree distribution of the Internet followed a homogeneous law, generally modeled by a Poisson distribution. Based on actual measurements, though, these papers gave evidences of the fact that it may be much more heterogeneous, better modeled by a power-law. Since then, much effort is devoted to studying this distribution and capturing it in appropriate models of the Internet.

However, as stated above, recent works have shown that the observed degree distribution of the Internet may be significantly biased by the measurement process. Indeed, it is deduced from partial maps obtained through intricate and unreliable measurements which tend to give biased views.

Our objective here is to show the implementation of a new approach able to rigorously estimate the degree distribution of the Internet, in a much more reliable way than before. Instead of building maps of this topology, we design a method to accurately estimate the degree of a given router; then we build a process that allows to select core routers uniformly at random, and estimate their degree. We therefore obtain an accurate estimate of the degree of a set of random nodes in the considered topology, representative of the whole.

It is worth noticing that the present work is related to two complementary papers. In [17], we showed by simulations that the approach succeeds in accurately estimating the degree distribution of various topologies and is free from bias. In [18], we deployed a real world-wide measurement campaign using this tool and we presented the first results related to the degree distribution of routers in the core Internet.

We implement here this approach with UDP PING, a tool that we designed and implemented, which sends a UDP packet to a target on an unallocated port. See Section IV-A for details on the implementation and behavior of this tool.

Notice that not all routers answer to such probes, and that some routers always use the same interface to answer. Likewise, some routers may answer with random interfaces. We will handle such issue below. The key point here is that, given an appropriate set of monitors and a correct target, we are able to accurately estimate its degree. This is already an improvement over previous situation, where no such tool existed and where the degree of a node could only be guessed from maps obtained with traceroute-based measurements.

B. Sampling targets

With the ability to reliably estimate the degree of a single router as explained above, one may estimate the degree distribution of a set of nodes (by applying the method to each node in the set). If this set is a set of nodes taken uniformly at random in the core Internet (i.e. all the nodes have the same probability to be chosen) then the obtained degree distribution is itself an approximation of the degree distribution of the core Internet, and the larger the sample, the better the approximation. However, choosing uniformly at random a node in the core Internet is not possible in general. We explain here how to bypass this issue.

First notice that it is easy to sample a random IP address, as this is noting but a 32 bit integer. But a random IP address is not a random node, as routers may have several interfaces (aka IP addresses) and then the probability to sample a router by sampling a random address is proportional to its number of interfaces. As a consequence, if we know the degree distribution \( p_k \) of nodes corresponding to IP addresses sampled uniformly at random, then the degree distribution of the set of all nodes is nothing but \( p_k/k \).

In summary, although we are unable to sample uniformly at random nodes in the core Internet, we are able to sample uniformly at random IP addresses and infer from the degree distribution of the corresponding nodes the one of the whole core Internet.

A. Measuring the degree of a given router

When a machine in the Internet receives an invalid packet from a sender \( S \), it is in general supposed to answer to \( S \) with an error message using the dedicated ICMP protocol. Many tools, including traceroute and many alias resolution tools, rely on this feature to generate errors which provide information regarding the network. An important feature of such error messages is that they are in principle sent using the interface of the machine that routes packets towards \( S \). Therefore, by generating an error on a target \( t \), a monitor \( m \) obtains an interface of \( t \). If many monitors distributed in the Internet do so, then they will probably obtain all interfaces of the target, and so its degree.

In addition, the random IP addresses we sample may belong to routers outside the core, to end-hosts, or even to routers in the core behaving incorrectly (they do not answer to probes or always use the same interface to answer). We easily identify IP addresses which do not belong to core routers: we see only one interface for them during our measurements, and simply discard them (as a consequence, we obtain no estimate of \( p_1 \), the fraction of nodes of degree 1). Notice that core routers which behave incorrectly also lead to nodes observed with degree 1, or even 0, and discarding them is correct. This induces no bias on the observed degree distribution if their behavior is not correlated to their degree, which seems a reasonable assumption.
Finally, we obtain a practical method to sample a set of uniformly random IP addresses of core routers, and to infer from their degree distribution the one of the whole core Internet. The quality of this estimate obviously depends on the quality of our set of monitors and on the size of our set of IP addresses. In-depth study of the achieved quality is a challenging task which belongs to future work, but we will discuss below ways to do so.

IV. DESCRIPTION OF THE TOOL

The implementation of the approach presented above led to the design of UDP PING, available at http://www.rotenberg.io/udpping/. We present below details of the implementation and how to use it to deploy a coordinated and distributed measurement campaign.

A. UDP PING

UDP PING is a measurement tool developed to discover interfaces of machines in the Internet. It is run on a machine M called the monitor. It takes the following parameters:

- Source interface (IS). An 4-byte integer corresponding to an interface (IP address) of M on which UDP PING is executed. When M is an end-host, it often has only one interface, called the IP address of M.
- Target interface (IT). A 4-byte integer corresponding to an IP address.
- UDP Destination Port (P). A 2-byte integer in the 49152 - 65535 range. UDP Ports in this range are assumed to be usually unallocated.

The goal of UDP PING is to detect if there is an active host T (the target) that owns the address IT and, if such T exists, to obtain an interface IT’ of T that is used by T to send packets towards IS (belonging to M). To do so, UDP PING forges an IP packet (see Fig. 1), carrying an UDP packet (see Fig. 2). The destination address of the IP packet is IS. The destination port of the UDP packet is P. IT’ is split in two 2-byte parts. The first 2-byte part is stored in the ID row of the IP packet header. The second 2-byte part is stored in the UDP Source Port row of the UDP message. By doing so, all the 4 bytes of IT’ are stored in the packet/message headers. The data field of the UDP packet contains a signature (typically a contact e-mail address) to allow network administrators to contact the sender upon receiving the message.

The IP/UDP packet is then sent by UDP PING through IS. Once the packet is sent, UDP PING listens to all incoming ICMP messages. If IT’ corresponds to an active host T, then after receiving the packet, it detects that P is unreachable. T generates a type 3 ICMP message (Destination Unreachable), with error code 3 (Port Unreachable). This ICMP message also contains the headers of the incriminating packet, and in particular, the original IP ID row and UDP Source Port row. T then chooses one of its interfaces, IT’, according to its routing policy, to send the ICMP message back to its sender, IS. Note that if T has more than one interface, IT’ can be different from IT. Eventually, UDP PING catches this ICMP message. It extracts the following information:

- ICMP attached information, including the original IP ID row and UDP Source Port row. This allows UDP PING to reconstruct the 4-byte integer corresponding to IT’, and to identify this ICMP message as the expected one.
- Source address of the ICMP message - that is IT’.

UDP PING then exits with a success code and returns IT’. After a set amount of time, if UDP PING has not caught any ICMP message properly identified as the expected one (using the 4-byte integer reconstruction), it exits with a failure code. This can happen for multiple reasons, the most common being:

- IT’ does not belong to any connected machine (target offline)
- IT’ does belong to an existing, active machine T, but T discards UDP errors without sending ICMP error messages back
- IT’ correspond to an active machine T that generates an ICMP error message but this message is filtered on its way back to M (usually near T).
- IT’ is located beyond a firewall that silently discards unsolicited UDP traffic.
- P is used by T, making it effectively reachable (and then no error is generated at the network level).

B. Deploying UDP PING

Since IT’ can vary depending on IS (and therefore depending on M), UDP PING may be used from multiple monitors to get a list of distinct interfaces belonging to the same target. Using UDP PING from a set of monitors towards the same target IT’ is called Distributed UDP PING. If the set of monitors is large enough and well distributed in the Internet, one may then obtain several (and even all) interfaces of the target.
Note that UDP PING requires privileges not only to execute binary code on the monitor, but also to forge and send packets at a very low level, and to listen and decode all the incoming ICMP messages. On most UNIX systems, UDP PING therefore requires root privileges, which can be restrictive. However, if one disposes of a set of monitors and is granted such privileges, UDP PING can be effectively distributed using shell scripts and ssh tools to launch remote execution of the tool and retrieve the answers.

Since receiving a large amount of UDP messages from distributed machines can look like a distributed attack by the target host, or even unintentionally disable it, extreme care must be taken not to send all the UDP messages at once. A delay can (and should) be set between the sending of successive UDP PING probes from the different monitors. Likewise, all monitors should use the same target port, or else the measurement may look like a UDP port scanning. Once all these precautions are taken, the tool has no specific parameters to tune and is rather simple to use.

V. CONCLUSION

In the work, we presented the details of the implementation of a new method able to accurately estimate the degree distribution of routers in the core Internet. We showed the principle on which it relied, how it can be used to measure the degree of a single router and how it can be deployed for a coordinated distributed measurement campaign, leading to an estimation of the degree distribution of such routers.

ACKNOWLEDGMENT

This work is partly supported by the European Commission EULER project (FP7 FIRE grant 258307) and by the Agence Nationale de la Recherche DynGraph grant ANR-10-JCJC-0202.

REFERENCES


