LINGI2141 - Individual Project Analysis of APT-GET

Benoît Baufays

Université Catholique de Louvain

Abstract

This paper will deal with analysing the managing software apt-get wich are deployed on several linux distributions

I. Introduction

Apt-get is a software develloped for linux OS to centralize the management of your software. Apt-get install packages containing precompiled code, configuration files, and meta-information about the package. Because it is not very usefull to manage manually all your packages, apt-get was created. With him, you can update your system and yours packages but you can also install or remove packages. The utility of apt-get is the management of the dependencies and, with one program, you can maintain your system up to date.

II. Apt-get

Apt-get have several commands, depending on what you want to do.

- update : with this command, apt-get search, on remotes servers, the last version for all packets. It get also the entire list of packets you can install via apt-get;

- upgrade or dist-upgrade : with this command, apt-get downloads packets installed on your system wich are to be updated. The difference between the command "upgrade" and "dist-upgrade" is that, whit hthe first, apt-get doesn't install new packages. For example, some packages must requires new dependencies. If you update with the first command, apt-get doesn't install new dependencies and, of course, doesn't update the packet. In other hand, with the second command, apt-get install all dependencies and update all packages;

- install : with this command, you can install new package. Apt-get search for dependencies and install the package and the required packages;

- remove : with this command, apt-get remove the package mentionned. It can also remove packages wich are not still required on your system. For example, if a package is only installed because it is a dependency for an other package and you remove this package, it is not require to have the dependency's package on your system.

Apt-get have several others command but the main has be presented below. To analyse how apt-get works and deal with network, we will use the same order presented in the list below. But, before, we discuss about the IPv 6 and apt-get.

III. Apt-get and IPv6

To know where it must searching informations about packages, apt-get have one file "/etc/apt/sources.list" which contains addresses of servers. According to several bloggers, apt-get have some troubles with IPv6 because some servers have no Ipv6 address or the ISP doesn't support it.
To verify this information, we have used dig to send a DNS request to archive.ubuntu.com, the main server for every sources.

```
;; QUESTION SECTION:
;archive.ubuntu.com.            IN      AAAA

;; ANSWER SECTION:
archive.ubuntu.com.     425     IN      AAAA    2001:67c:1360:8c01::19
archive.ubuntu.com.     425     IN      AAAA    2001:67c:1360:8c01::1a
archive.ubuntu.com.     425     IN      AAAA    2001:67c:1360:8c01::22
archive.ubuntu.com.     425     IN      AAAA    2001:67c:1360:8c01::23
archive.ubuntu.com.     425     IN      AAAA    2001:67c:1360:8c01::15
archive.ubuntu.com.     425     IN      AAAA    2001:67c:1360:8c01::18
```

As we can see, archive.ubuntu.com have several IPv6 address, the problem do not come form that. When we go deeper in the code of apt-get, we see that some functionnalities are still using IPv4 libraries.

IV. Apt-get analysis

I. Update

I.1 DNS Request

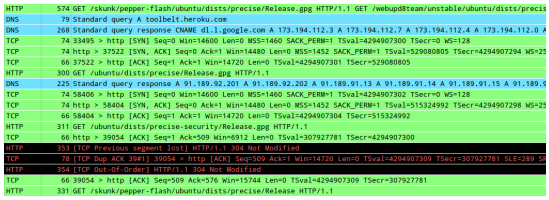
With this command, apt-get read first the file containing all server's name. With this information, it sends DNS request to know IP address of each server.

DNS	79	Standard query A security.ubuntu.com
DNS	79	Standard query A security.ubuntu.com
DNS	73	Standard query A dl.google.com
DNS	77	Standard query A ppa.launchpad.net
DNS	77	Standard query A extras.ubuntu.com
DNS	79	Standard query A toolbelt.heroku.com
DNS	81	Standard query A be.archive.ubuntu.com
DNS	223	Standard query response A 91.189.92.201 A 91.189.92.202 A 91.189.91.13
DNS	79	Standard query A security.ubuntu.com
DNS	223	Standard query response A 91.189.92.190 A 91.189.92.200 A 91.189.92.201
ICMP	251	Destination unreachable (Port unreachable)
DNS	268	Standard query response CNAME dl.l.google.com A 173.194.112.6 A 173.194
DNS	73	Standard query A dl.google.com
DNS	93	Standard query response A 91.189.95.83

As we can see, Apt-get send first all his DNS requests before contacting servers. We see also that dl.google.com, an entry that we have added manually to access to packages from Google (GoogleTalk, ...) is, in reality, reachable via dl.l.google.com. Finally, we can see that apt-get send twice the DNS request about dl.google.com. It's not beacause it doesn't receive the response but because we have added twice this entry in the config file.
To test the DNS system, we have put manually a arbitrary IP address for the second DNS server. This test is visible in the schema below with the "Destination unreachable" message. After that, apt-get doesn't reuse this IP adress to send DNS query.
When we analysing packets receive by apt-get, we see that the time life of the information is 3 minutes 30. Also, we see that it receive more than one IP address for every server name. With this solution, apt-get doesn't want to resend a DNS query if a IP adress down. With multiple IP adresses, it can also use multi threading and request informations on multiple servers.

I.2 Getting information

With the IP address, apt-get can now getting informations about packages. These informations are getting in two step.



First, as we can see in the schema above, apt-get get some files with HTTP 1.1. Before, it do a three handsake (SYN, SYN-ACK,ACK) to open the connection. For example, after having open a connection with the server, apt-get download "http://extras.ubuntu.com/ubuntu/dists/precise/Release". Because it's a file readable, we have downloaded it to see how apt-get works.

```
MD5Sum:
 7e0a2dc5fb8c8bf6922bbe396d66addc            136 main/binary-amd64/Release
 529267ce92bc20387c1de25a745a8f31          11858 main/binary-amd64/Packages.gz
 c3265772d26cc0846fe30c18477c9d8f          35415 main/binary-amd64/Packages
 4019d042f1f2fe7644056a366b3364f1          10788 main/binary-amd64/Packages.bz2
 5d3e88adf3808442dfd95ac307793647          29551 main/binary-armel/Packages
 7345b995ca20588334bbf61f225eb13a           8776 main/binary-armel/Packages.bz2
 54a0f788fa15a5732e41ce407c94865f            136 main/binary-armel/Release
 ee008f3671557b31e107483e54272e9f           9606 main/binary-armel/Packages.gz
 ee008f3671557b31e107483e54272e9f           9606 main/binary-armhf/Packages.gz
 e977bbd456b73f23a02d3231429b5149            136 main/binary-armhf/Release
 7345b995ca20588334bbf61f225eb13a           8776 main/binary-armhf/Packages.bz2
 5d3e88adf3808442dfd95ac307793647          29551 main/binary-armhf/Packages
 d236b946108974634dd3f4ea9c593aa3          35407 main/binary-i386/Packages
 cd8c2ae9970ced4a773275ef46c0bbe5            135 main/binary-i386/Release
 a7111e0bc5e918b8ef316ef933bf6e23          10788 main/binary-i386/Packages.bz2
 90a15cdf190b2c7ba2b67ca12be21806          11855 main/binary-i386/Packages.gz
 7345b995ca20588334bbf61f225eb13a           8776 main/binary-powerpc/Packages.bz2
 ee008f3671557b31e107483e54272e9f           9606 main/binary-powerpc/Packages.gz
 5d3e88adf3808442dfd95ac307793647          29551 main/binary-powerpc/Packages
 d4ec4b93f4d563c89dfd4875a3daa7c4            138 main/binary-powerpc/Release
```

For each subfiles, containing informations about packages, it receive the MD5 sum or the SHA1 sum (not present in the schema), the size of the file you will download and, finally, the path to download the file. In fact, it's not a file, it's a archive. With this solution, ubuntu compress informations.
With the schema above, we can also see that apt-get still send and receive DNS query while we have noticed, i nthe previous subsection that apt-get send all his queries before getting informations about packages. It's true for main addresses but, for addresses you have added manually, apt-get begin to download informations about packages before it have finished DNS queries for personnalized addresses.
We see also how apt-get retrieve from lost packets (dark line in the schema). Because apt-get use TCP, it receive first a packet that indicate the lsot segment. With this information, apt-get send back a duplication acknowledgement. In this example, we have added a delay for some packets and we can see that, after the duplication acknowledgement, apt-get receive the lost packet and TCp say "TCP out of order ". it means that this packet arrive not in the correct order. it's logical because with put a delay for one packet.
After having downloaded the main file, apt-get download archive listed in the main file. If you download also the archive, you can see that it contains one file, listing all packages and informations about each : checksum, name, description, last version, dependencies, ... With this file, apt-get can updated his local informations and test if you have the last version. If not, it's mark the package.

II. upgrade

Upgrade is just a succession of install command for every package which requires an update. So, you can find our analysis of install in the next section.

III. Install

To analysis packets and network activity, we have installed a package with dependencies.
Again, apt-get work in two step : DNS query and download. After checking that the requested package exist in its database, it extracts, from informations taken during the update command, the address of the server where it can download package.
In our example, it search also informations about dependencies and if not installed, it adds these packages to download. In fact, apt-get have a cache system and if the connection is closed before it have finished to download, you can restart the command and apt-get resumes with the last packet received. It is possible thanks to HTTP protocol that allow to resume download, if this option is enabled on the distant server. Here, ubuntu servers have enabled this option.

III.1 DNS Request

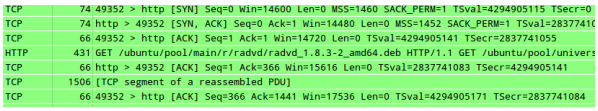
DNS	81	Standard query A be.archive.ubuntu.com
DNS	81	Standard query A be.archive.ubuntu.com
DNS	225	Standard query response A 91.189.91.13 A 91.189.91.14 A
DNS	81	Standard query A be.archive.ubuntu.com
DNS	225	Standard query response A 91.189.92.202 A 91.189.91.13 A
ICMP	253	Destination unreachable (Port unreachable)
DNS	225	Standard query response A 91.189.91.13 A 91.189.91.14 A

Like in the subsection "update", we see that apt-get send DNS request twice. Here, it sends to the principal and the second DNS server address. Again, it receives an "Destination unreachable" packet for the second DNS server.
We can also see that it sends a third time a DNS request to "archive.ubuntu.com". This time, it

sends to the principal DNS server and it's for a dependency package. He could have used the information from the previous query, but it's a consequence of multi thread : to increase the speed of downloading, apt-get download two packages simultaneously.

III.2 Download packages

After having IP adresses of servers where apt-get can download packages, apt-get open a TCP connection with the server to download packages.



Like every TCP connections, it starts with the three handsake between the client and the server (SYN,SYN-ACK,ACK). After the connection is opened, apt-get send a HTTP request to download the package.
As yo ucan see, the server sends a lot of packets and our system send back acknowledgment but not for every packet. In fact, Wireshark indicates "TCP segment of a reassembled PDU". This label notices that Wireshark reassembles a higher level protocol packets. In this example, it's normal to see this label because the reply of the HTTP request take more than one packet. When we analysis this reassembled packet, we can see the next sequence number and the acknowledgment number. When we get the next packet sended by our system, we see that it's a ACK packet with the correct number linked to the previous reassembled packet.

At the end of the download, our system receive a HTTP packet with the type of the file downloaded. in this exmaple, it's an application/x-debian-package, a binary package.
When all packages are downloaded, we see that our system close the TCP connection with FIN-ACK,FIN-ACK,ACK packets.
With this sequence, we see that apt-get doesn't send informations about your installation or errors. To check more deeper this preliminary conclusion, we have analysed networck activity during the last phase of install, where apt-get install packages on our system. We see not packets from apt-get program or from Ubuntu. When we stop the processus, we also don't see packets. We can see that apt-get doesn't send informations about your system

IV. Remove

Normally, apt-get should not send or receive packets when you remove a package but we wondered if, for statistical reasons, Ubuntu track remove command. For example, Ubuntu can track remove action to compute ranking for every package. If you open Software Manager, an UI application for apt-get, yo ucan see a ranking for every main packages.
After having remove small packages, we see that, with Wireshark, no packets was echanged. So, we remove kernel package (previous version of course) and, again, no packets was echanged. We can say that apt-get doesn't track remove action.