ANNALS OF GEOPHYSICS, 62, 5, DM567, 2019; doi: 10.4401/ag-8050

“MAKING LINKED DATA MORE RELIABLE WITH A FAILOVER SERVER SYSTEM: A CASE STUDY WITH SEISMOLOGICAL DATA AT INGV”

Luca Nannipieri*,[1], Stefano Cacciaguerra[2], Santi Mirenna[3], Mario Locati[3], Marco Marletta[4], Emanuele Gucciardi[4]

(1) Istituto Nazionale di Geofisica e Vulcanologia, Sezione di Pisa (Italy) – INGV
(2) Istituto Nazionale di Geofisica e Vulcanologia, Sezione di Bologna (Italy) – INGV
(3) Istituto Nazionale di Geofisica e Vulcanologia, Sezione di Milano (Italy) – INGV
(4) The Italian Academic and Research Network – GARR

Article history
Receveid **December 5, 2018**; accepted **April 11, 2019.**
Subject classification:
Failover; Networking; Monitoring; Internet services, Disaster recovery.

ABSTRACT

Since the introduction of the Linked Data concept more than a decade ago, today the way scientists use and share their research data is increasingly changing. The growing availability of big amounts of Open Research Data is challenging the old-fashioned habit of replicating large datasets locally prior to their processing, and the new Open Science paradigm tries to update the way researchers works by pushing them into solutions enabling a quicker, lightweight and more efficient circulation of research results. Granting access to its data resources is an activity taken very seriously at INGV, moreover as it is head of two important European Research Infrastructure Consortium (ERIC), namely the European Multidisciplinary Seafloor and water-column Observatory (EMSO), and the European Plate Observing System (EPOS). As more and more data are becoming accessible via web services, an increasing number of researchers are starting to include their use in their workflows, thus, heavily relying on their correct and continuous operational functioning, in other words, in their service continuity. Experimenting solutions for enhancing the reliability of web services is at the core of the present work. We propose here a solution that is able to guarantee the functioning of a service in case of an IT infrastructure fail, a mechanism that is able to automatically switch to a secondary server located in another network. The solution is made possible thanks to a collaboration with GARR, the ultra-broadband network dedicated to the Public Italian Research, and is able to deal with network problems, server hardware problems, as well as service level problems.

1. INTRODUCTION

The Istituto Nazionale di Geofisica e Vulcanologia (INGV) is a geographically distributed Italian public research Institute operating in the broad field of natural science, especially active in seismology and volcanology. INGV is part of the broader Italian Civil Protection system, and plays a key role in case of any seismic, volcanic o tsunamigenic activity occurring in the national territory. INGV operates a 24/7 emergency monitoring facility in close cooperation with the Civil Protection Department that coordinates the overall complex National Civil Protection System [Lucini, 2014; Pennisi, 2014], and must guarantee that all of its monitoring services are fully functional and accessible at all times, even in case of heavy stress of any sort. Altering any aspect of such a complex infrastructure is a very complicated and sensitive issue, from many point-of-views, both technological, and political, as it may affect the security of the Italian population.

In addition to the continuous monitoring of the national territory by means of a complex instrumental networks, INGV conduct research activities on a variety of fields and is involved in a multitude of regional, national, European and worldwide projects. These activities are carried out by dedicated working groups or researchers that organize or simply publish their findings on the Internet and relies on web and data servers usually hosted at data centres operating in one of the INGV departments.

In the last few years, the concept of Research Infrastructure (RI) emerged, a term indicating facilities, resources and services that are used by the research communities to conduct research and foster innovation in their fields. RIs have been established thanks to Institutional, National and European public investments, and their key role have been recognized at all levels. The European Strategy Forum on Research Infrastructures (ESFRI) set up a series of roadmaps for supporting European RIs, and European Research Infrastructure Consortiums (ERICs, legal entities under European Union law, see Council Regulation n.723/2009) are strongly involved in these roadmaps. INGV hosts two ERICs, the European Multidisciplinary Seafloor and water-column Observatory (EMSO), established in 2016, and the European Plate Observing System (EPOS), established in 2018. Institutions of various nature (e.g. Public and Private Universities, Public and Private Research Institutions) are involved in both EMSO and EPOS, some acting as a single node data provider, some acting as a collective node gathering data from multiple data providers. In order to become a data provider node, an Institution must sign an agreement and guarantee the continuous operational access to their data infrastructures by means of services compliant with specific standards.

The reliability of services operating at Institutional level is of fundamental importance to guarantee the expected functioning of the overall architecture. These aspects are particularly important for huge infrastructures like the European Integrated Data Archive [EIDA; Clinton et al., 2014] that is meant for sharing instrumentally recorded seismic data for the entire European continent. EIDA approached the reliability of its nodes by means of a own routing service at application level [Quinteros, 2017] working in a federation of data centers, a solution that guarantee to securely archive seismic waveform data and related metadata, and, at the same time, provide transparent access to data for the geosciences research communities.

Various projects funded by the European Commission tried to provide solutions addressing specific tasks and aimed at increasing the reliability of services. Among others, the Project European Research Data Services, Expertise & Technology Solutions [EUDAT; Lecarpentier et al., 2013] set up very useful services such as remote data storage or remote user authentication and authorisation (AAAI, Authentication, Authorization, Accounting and Infrastructure). However, we could not find a viable solution aimed at solving problems occurring to a single-node data service provider that is part of a bigger and interlinked network and that may cause its unreachability. In a network with non-redundant nodes problems like that decrease the reliability of the entire network, causing a breach in the Linked Data [Berners-Lee, 2006; Bizer et al., 2009] implementation.

The solution presented in this work is a failover system able to overcome issues occurring to a data service node (i.e. service continuity) by exploiting the geographically distributed nature of INGV, paired with the network solutions provided by the GARR Consortium, the ultra-broadband network dedicated to the Italian research and education community. Thanks to an innovative combination of already existing, well-developed and well-documented tools and protocols, the proposed solution can be easily adopted by other data repositories and related services running in a similar environment.

2. THE CASE STUDY

As previously mentioned, INGV must guarantee the remote access to its data repositories and web services in order to let remote workflows transparently and safely incorporating INGV data. This work exploit the combination of three factors:

1) the geographically distributed nature of INGV, having its head office in Rome, then departments in Milan, Bologna, Pisa, Naples, Catania, and Palermo (Figure 1);
2) the availability of a data centre operating in each INGV department, each run by fully dedicated and highly qualified IT personnel;
3) the involvement of INGV with GARR, the ultra-broadband network dedicated to the Italian research and education community.

INGV was created back in the year 2000 by merging pre-existing Public Research Centres, more or less active in the seismological and volcanological scientific fields. As a result, INGV is a geographically distributed institution with a department in various Italian cities, thus providing a good coverage of the entire national territory.

The INGV head office in Rome collects all data recorded by the Italian Seismic Network [Michelini et

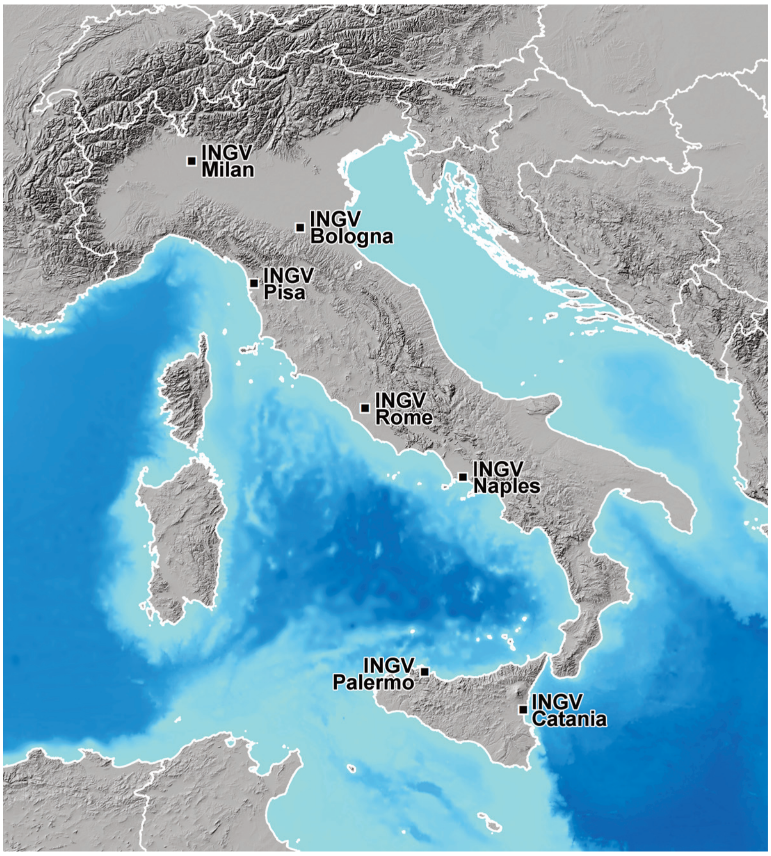


FIGURE 1. INGV departments, as they are geographically distributed over the entire Italian territory.

al., 2016], as well as other multi-parametric networks generate a massive volume of seismic, volcanic and tsunamigenic measures [Pintore et al., 2016] that could be considered "Big Data" as multiple petabyte of heterogeneous data are generated every year. The primary use of these data is for Civil Protection purposes such as the real-time monitoring of natural disasters, and, in the long run, for setting up Early Warning Systems [EEWS; Gasparini and Manfredi, 2014]. The Emergency Response Coordination Centre (ERCC) that is operating in the framework of the European Civil Protection and humanitarian aid operations is trying to exploit the availability of multiple interoperable services providing real-time monitoring data on various types of natural disasters in the European Union, and the EC Project ARISTOTLE [Michelini at el., 2017] is investigating this possibility. These activities demonstrates the importance for the entire society of publishing and connecting structured data on the Web, which is in fact the definition of "Linked Data".

In addition to the importance for Civil Protection purposes, data coming from the INGV monitoring networks enters the *Research Data Lifecycle* and it allows researchers to perform extensive analysis for a better understanding of the natural phenomena. Most INGV departments contribute to the national network with their respective area of competence, and, in addition, manage and process on the fly locally collected data using their local infrastructure depending on the type of needs of each working group operating at that specific department. In fact, data and the required expertise is not concentrated in one Department only, and both are distributed over the entire INGV.

As a consequence, of all the above-described activities, each INGV department operates a more or less complex data centre, and all departments are interconnected via the network managed by the GARR Consortium (Figure 2). GARR implements the ultra-broadband network dedicated to the Italian research and education community, providing high-performance connectivity and developing innovative services for the daily activities of researchers, professors and students as well as for international collaborations. The GARR Consortium is a non-profit association founded under the auspices of the Ministry of Education, University and Research.

Among the various research fields, there is an INGV working group working on "Historical Seismology", a field of seismology dealing with data on the past, historical seismicity retrieved from documentary sources, instead than relying on instrumental recordings. The

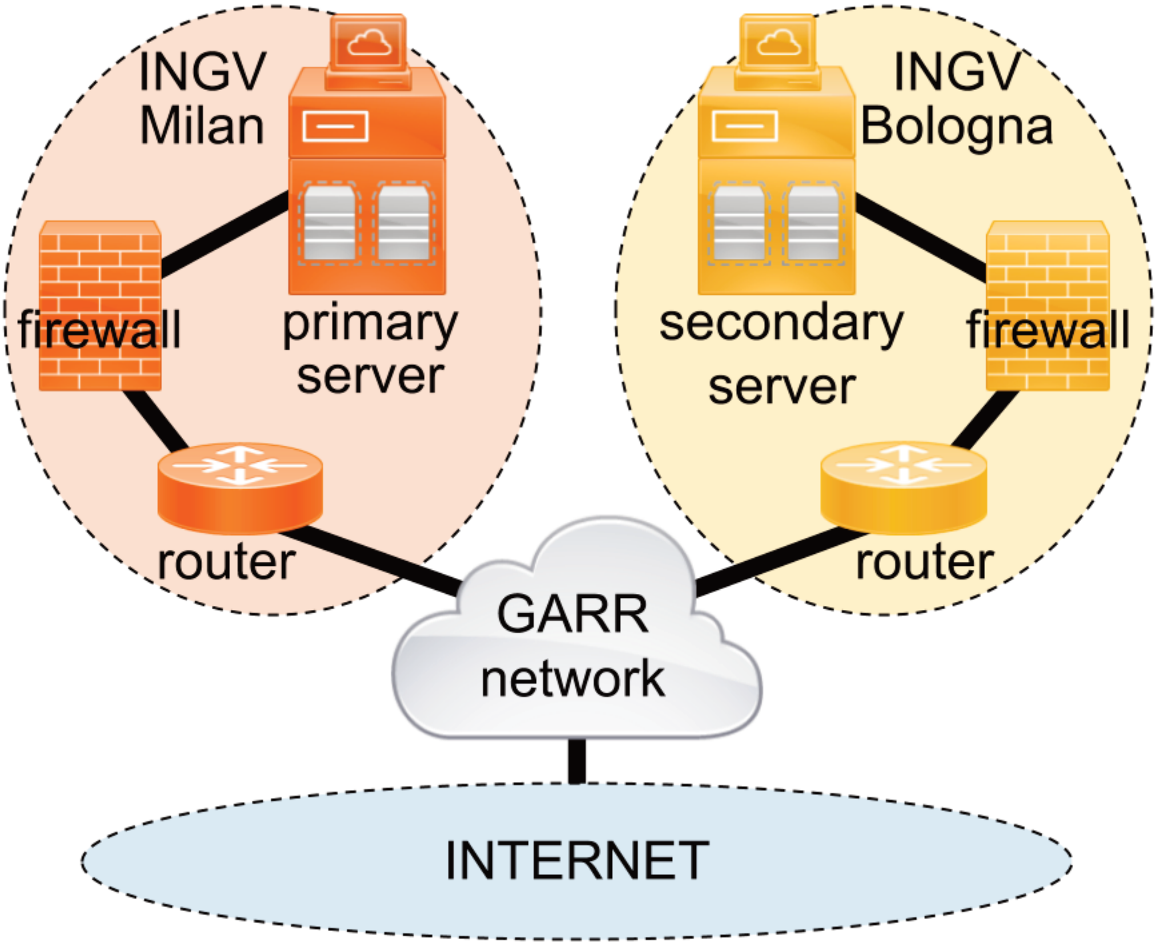


FIGURE 2. The network scenario in which operates the solution presented in this paper.

Italian seismological community has a long tradition of collaborations with historians, that in the past forty years lead to the publication of few hundreds scientific papers, thus taking advantage of the huge amount of well-preserved historical archives existing in Italy. The final result of this huge and long-term effort, is the longest and most complete datasets on past seismicity in the world [Valensise et al., 2003], tracing back earthquakes over the past millennium and beyond, and it is one the fundamental input data for assessing the Italian seismic hazard.

Thanks to its long experience in the historical research of the past seismicity, INGV coordinate and managed AHEAD, the European Archive of Historical Earthquake Data [Locati et al., 2014], also. Via AHEAD, a series of European Institutions active in Historical Seismology collaborate, and deliver their data on the past seismicity to the EPOS ERIC.

Up to now, all data on historical seismicity published on the Internet by the above mentioned repositories (roughly estimated in 60GB circa on the filesystem) was made available using a unique web server called “emidius”, operating at INGV in Milan. Even if a series of daily backups are in place, in case of failure of the data centre in Milan, these repositories simply became inaccessible. As a consequence, any remote activity relying on these data services stop working, which is an unacceptable situation for an up-to-the-standards modern Research Data Infrastructure.

3. FAILURES AND FAILOVER SYSTEMS

In computing and networking terms, a failover is a mechanism aimed at switching to a redundant or standby server, system, hardware component or network in case of *failure* or abnormal termination of the previously active application, server, system, hardware component, or network (https://en.wikipedia.org/wiki/Failover).

A failover is automatically enabled and usually operates without any warning, and any human intervention. Used to make systems more *fault-tolerant* (https://en.wikipedia.org/wiki/Fault_tolerance), a failover is a primary mechanism of every mission-critical service that requires to be constantly available. Fault-tolerance is the property that enables a system to continue operating properly in case of the failure of any component. In order to implement a data repository and/or a web service on a fault-tolerant system, the failover should be applied to three levels of failure:

1) *Network Failure* - the server becomes unreachable for problems due to a local network device. In this case, even if the server is correctly working, its services becomes unreachable.
2) *Server Failure* - the server is not working due to either a software fault or an hardware crash.
3) *Service Failure* - the server is up and running but its services stopped working, usually due to problems on the software side.

In this paper, we show how to implement a failover technique applied to a server that must guarantee access to its data repositories and/or its web services, and that is able to tackle the above mentioned three different levels of failure. The proposed solution is based on managing the way to route the IP packets (i.e. level 3 of the stack TCP/IP; https://tools.ietf.org/html/rfc791) of the traffic coming from users of our service. In other words, this means that we are operating on the mechanisms that rule/control how the Internet works. In this sense, our system reacts to the different levels of failure exploiting the mechanisms on which is based the Internet. So while the EIDA routing service works at application level [Quinteros, 2017], our approach works mainly at the network level and we configure the best network route according to criteria of better performance of the network connections, better site with computing power or preference to a site for reasons of institutional policies.

4. SOLVING A NETWORK FAILURE WITH BGP

In order to prevent a network failure, we implemented a network failover by means of the Border Gateway Protocol (i.e. BGP-4) for dynamic routing (https://tools.ietf.org/html/rfc4271). BGP is a standard gateway protocol to exchange routing and reachability information among different Autonomous Systems (AS) on the Internet by means of a session mechanism. This means that, periodically, the routers involved in a BGP session exchange information among them. Figure 3 shows the local networks in Milan and Bologna that are designed as private AS connected to the public AS implementing the GARR network. In this way, servers in Milan and Bologna are reachable from the Internet via the GARR network (Figure 2). For the sake of clarity, the GARR network has the public AS number 137 assigned by Regional Internet Registry - RIPE (https://www.ripe.net/about-us), while the INGV networks in Milan and Bologna have their own private AS assigned by GARR. For this reason, the GARR Consortium assigns the public IP addresses to both the local networks in Milan (*i.e. 193.206.88.0/24*) and in Bologna (*i.e. 193.204.89.128/26*). In order to implement our solution, both servers in Milan and Bologna

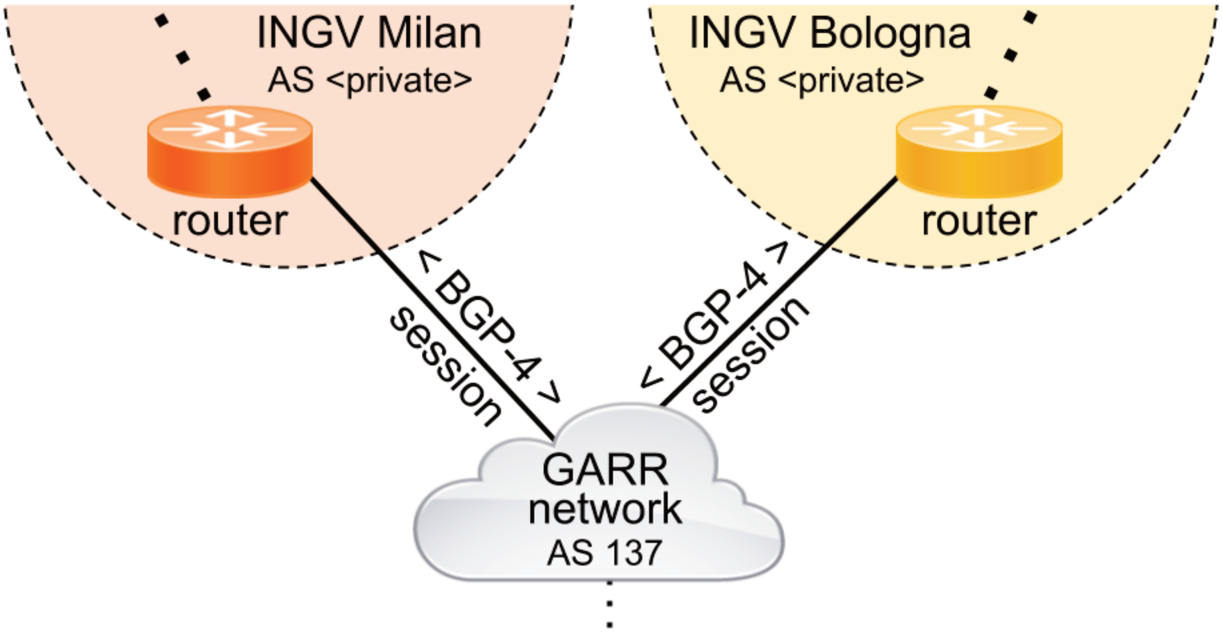
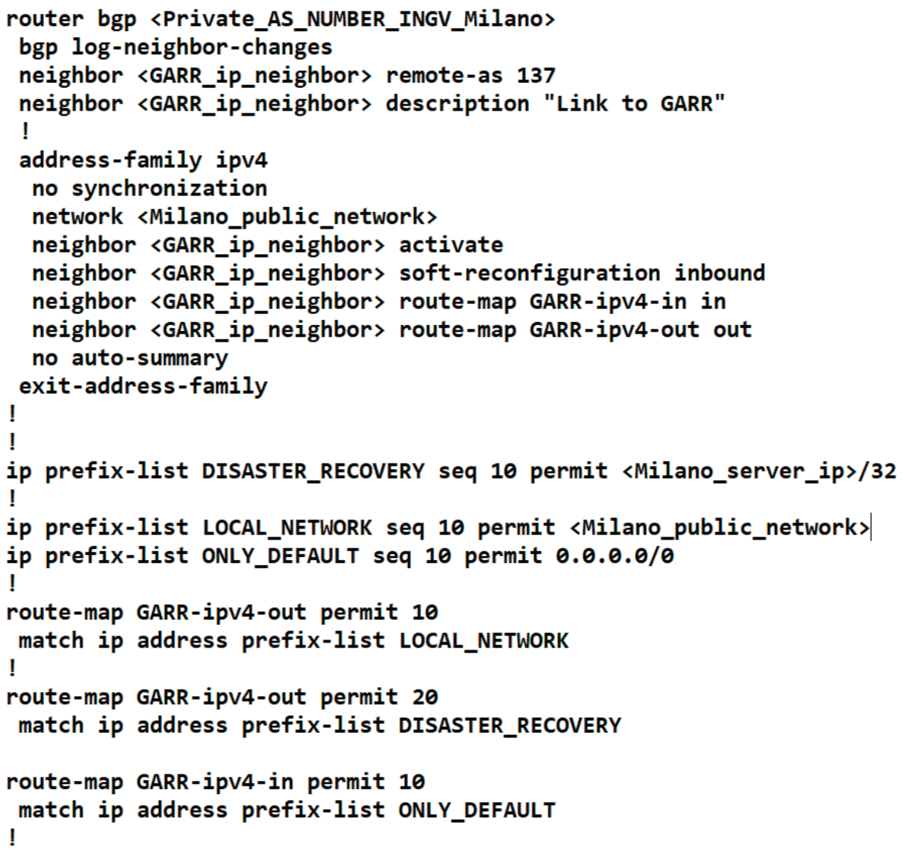


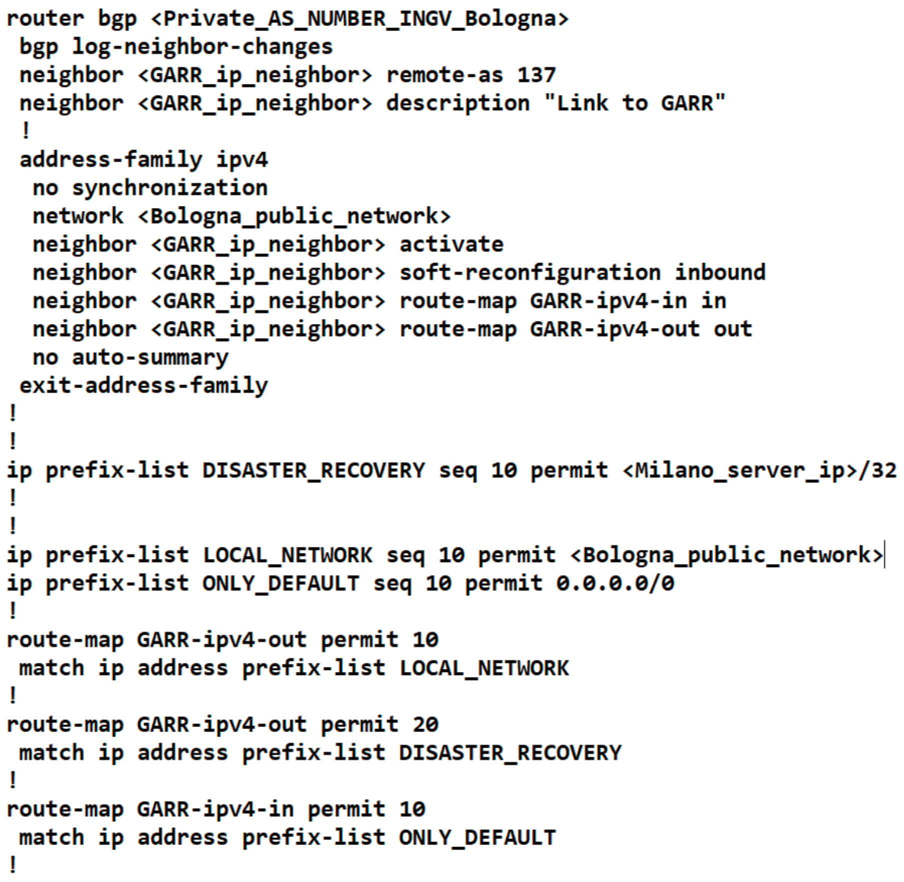
FIGURE 3. Private and Public Autonomous Systems (AS).



```
router bgp <Private_AS_NUMBER_INGV_Milano>
 bgp log-neighbor-changes
 neighbor <GARR_ip_neighbor> remote-as 137
 neighbor <GARR_ip_neighbor> description "Link to GARR"
 !
 address-family ipv4
  no synchronization
  network <Milano_public_network>
  neighbor <GARR_ip_neighbor> activate
  neighbor <GARR_ip_neighbor> soft-reconfiguration inbound
  neighbor <GARR_ip_neighbor> route-map GARR-ipv4-in in
  neighbor <GARR_ip_neighbor> route-map GARR-ipv4-out out
  no auto-summary
 exit-address-family
!
!
ip prefix-list DISASTER_RECOVERY seq 10 permit <Milano_server_ip>/32
!
ip prefix-list LOCAL_NETWORK seq 10 permit <Milano_public_network>
ip prefix-list ONLY_DEFAULT seq 10 permit 0.0.0.0/0
!
route-map GARR-ipv4-out permit 10
 match ip address prefix-list LOCAL_NETWORK
!
route-map GARR-ipv4-out permit 20
 match ip address prefix-list DISASTER_RECOVERY

route-map GARR-ipv4-in permit 10
 match ip address prefix-list ONLY_DEFAULT
!
```

FIGURE 4. BGP primary router configuration (INGV Milan).



```
router bgp <Private_AS_NUMBER_INGV_Bologna>
 bgp log-neighbor-changes
 neighbor <GARR_ip_neighbor> remote-as 137
 neighbor <GARR_ip_neighbor> description "Link to GARR"
 !
 address-family ipv4
  no synchronization
  network <Bologna_public_network>
  neighbor <GARR_ip_neighbor> activate
  neighbor <GARR_ip_neighbor> soft-reconfiguration inbound
  neighbor <GARR_ip_neighbor> route-map GARR-ipv4-in in
  neighbor <GARR_ip_neighbor> route-map GARR-ipv4-out out
  no auto-summary
 exit-address-family
!
!
ip prefix-list DISASTER_RECOVERY seq 10 permit <Milano_server_ip>/32
!
!
ip prefix-list LOCAL_NETWORK seq 10 permit <Bologna_public_network>
ip prefix-list ONLY_DEFAULT seq 10 permit 0.0.0.0/0
!
route-map GARR-ipv4-out permit 10
 match ip address prefix-list LOCAL_NETWORK
!
route-map GARR-ipv4-out permit 20
 match ip address prefix-list DISASTER_RECOVERY
!
route-map GARR-ipv4-in permit 10
 match ip address prefix-list ONLY_DEFAULT
!
```

FIGURE 5. BGP secondary router configuration (INGV Bologna).

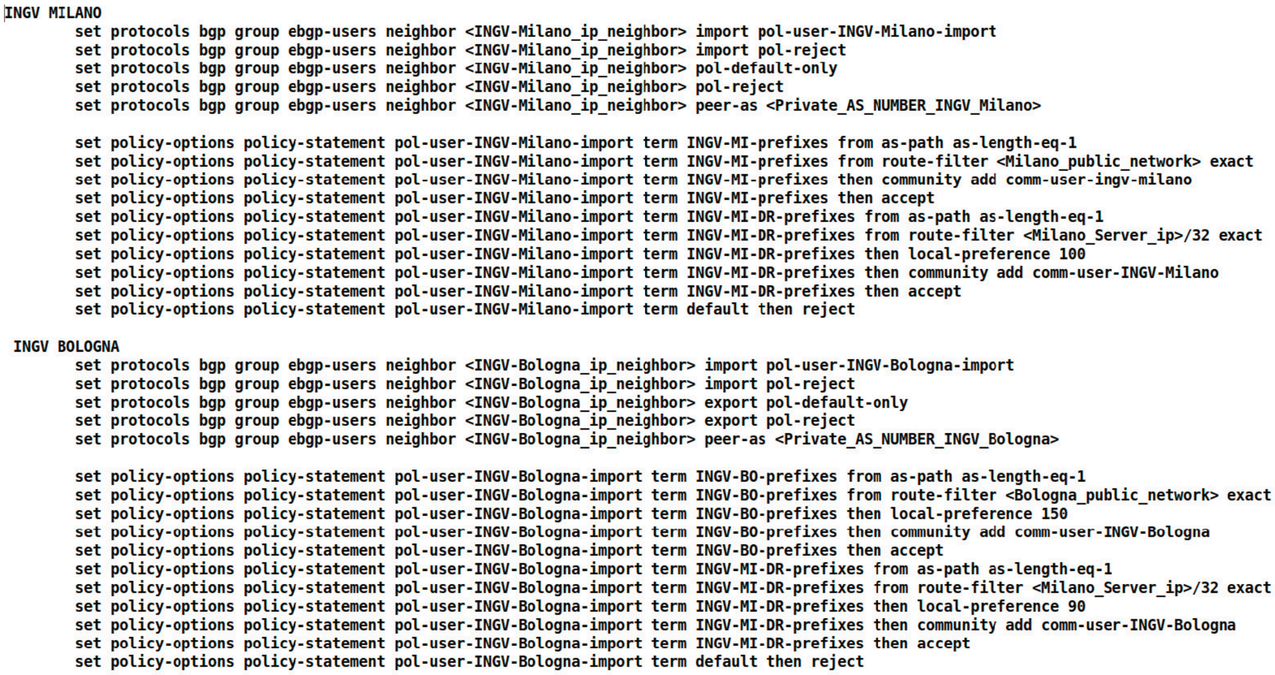
exploit the same IP address: this means that we create a primary/secondary architecture where both primary and secondary servers share the same public IP address even if they are located in different networks (i.e. different ASs). In order to create this primary/secondary architecture, we set the *higher local-preference* parameter for the dynamic routing in the 137 AS towards Milan (where the primary is located). In this way, if no problem occurs, all the traffic for the web service is directed to Milan, and, in case of problem, all traffic is diverted to Bologna. In order to implement this solution, we activated the BGP-4 protocol either in the INGV routers placed on private AS in Milan and Bologna or on GARR routers situated on the border of 137 AS (Figure 3). The setup of the *higher local-preference* in BGP prevents that a network failure in the local network makes unavailable the web service avoiding directing the traffic towards the unreachable server. This means that on GARR border routers, the reachability of server located in Milan network will take precedence by the higher LOCAL-PREFERENCE (well-know) attribute configured in routing policies. This parameter has to be set inbound on routes being received to influence the outbound GARR routing behaviour, toward to Milan (LOCAL-PREFERENCE = 100) and Bologna (LOCAL-PREFERENCE = 90) respectively. A higher local preference is preferred and the default is 100. The LOCAL-PREFERENCE is the first attribute to be defined as the "best" one and deserves a spot in the router's routing table. If you look at the BGP best path algorithm, you'll see no fewer than 13 steps to make this determination.

Below is an extract from the configuration made on Backbone's GARR routers (Figure 6).

The configuration section of our interest for the failover mechanism in case of Network Failure, is the one defined by term "INGV-MI-DR-prefixes" where we can see that the attribute "local-preference" configured on border router located in Milan PoP, has a greater value than that configured on the border router located in Bologna PoP.

5. SOLVING A SERVER FAILURE WITH AN INTERNAL BGP SESSION

Implementing a single external BGP session between the INGV and the GARR routers is necessary to prevent that a network failure makes unavailable the web service, but it is not sufficient in order to avoid a server failure obtains the same result. In order to prevent, also, a server fault, it is necessary to implement an internal BGP Session between the web server and its router. Both INGV routers in Milan and Bologna announce their assigned networks (respectively *193.206.88.0/24* and *193.204.89.128/26*) in addition to the *same single server IP address* of the primary server in Milan (*i.e. <Milano_server_ip>/32*).



```
INGV MILANO
        set protocols bgp group ebgp-users neighbor <INGV-Milano_ip_neighbor> import pol-user-INGV-Milano-import
        set protocols bgp group ebgp-users neighbor <INGV-Milano_ip_neighbor> import pol-reject
        set protocols bgp group ebgp-users neighbor <INGV-Milano_ip_neighbor> pol-default-only
        set protocols bgp group ebgp-users neighbor <INGV-Milano_ip_neighbor> pol-reject
        set protocols bgp group ebgp-users neighbor <INGV-Milano_ip_neighbor> peer-as <Private_AS_NUMBER_INGV_Milano>

        set policy-options policy-statement pol-user-INGV-Milano-import term INGV-MI-prefixes from as-path as-length-eq-1
        set policy-options policy-statement pol-user-INGV-Milano-import term INGV-MI-prefixes from route-filter <Milano_public_network> exact
        set policy-options policy-statement pol-user-INGV-Milano-import term INGV-MI-prefixes then community add comm-user-ingv-milano
        set policy-options policy-statement pol-user-INGV-Milano-import term INGV-MI-prefixes then accept
        set policy-options policy-statement pol-user-INGV-Milano-import term INGV-MI-DR-prefixes from as-path as-length-eq-1
        set policy-options policy-statement pol-user-INGV-Milano-import term INGV-MI-DR-prefixes from route-filter <Milano_Server_ip>/32 exact
        set policy-options policy-statement pol-user-INGV-Milano-import term INGV-MI-DR-prefixes then local-preference 100
        set policy-options policy-statement pol-user-INGV-Milano-import term INGV-MI-DR-prefixes then community add comm-user-INGV-Milano
        set policy-options policy-statement pol-user-INGV-Milano-import term INGV-MI-DR-prefixes then accept
        set policy-options policy-statement pol-user-INGV-Milano-import term default then reject

 INGV BOLOGNA
        set protocols bgp group ebgp-users neighbor <INGV-Bologna_ip_neighbor> import pol-user-INGV-Bologna-import
        set protocols bgp group ebgp-users neighbor <INGV-Bologna_ip_neighbor> import pol-reject
        set protocols bgp group ebgp-users neighbor <INGV-Bologna_ip_neighbor> export pol-default-only
        set protocols bgp group ebgp-users neighbor <INGV-Bologna_ip_neighbor> export pol-reject
        set protocols bgp group ebgp-users neighbor <INGV-Bologna_ip_neighbor> peer-as <Private_AS_NUMBER_INGV_Bologna>

        set policy-options policy-statement pol-user-INGV-Bologna-import term INGV-BO-prefixes from as-path as-length-eq-1
        set policy-options policy-statement pol-user-INGV-Bologna-import term INGV-BO-prefixes from route-filter <Bologna_public_network> exact
        set policy-options policy-statement pol-user-INGV-Bologna-import term INGV-BO-prefixes then local-preference 150
        set policy-options policy-statement pol-user-INGV-Bologna-import term INGV-BO-prefixes then community add comm-user-INGV-Bologna
        set policy-options policy-statement pol-user-INGV-Bologna-import term INGV-BO-prefixes then accept
        set policy-options policy-statement pol-user-INGV-Bologna-import term INGV-MI-DR-prefixes from as-path as-length-eq-1
        set policy-options policy-statement pol-user-INGV-Bologna-import term INGV-MI-DR-prefixes from route-filter <Milano_Server_ip>/32 exact
        set policy-options policy-statement pol-user-INGV-Bologna-import term INGV-MI-DR-prefixes then local-preference 90
        set policy-options policy-statement pol-user-INGV-Bologna-import term INGV-MI-DR-prefixes then community add comm-user-INGV-Bologna
        set policy-options policy-statement pol-user-INGV-Bologna-import term INGV-MI-DR-prefixes then accept
        set policy-options policy-statement pol-user-INGV-Bologna-import term default then reject
```

FIGURE 6. The GARR router BGP configuration.

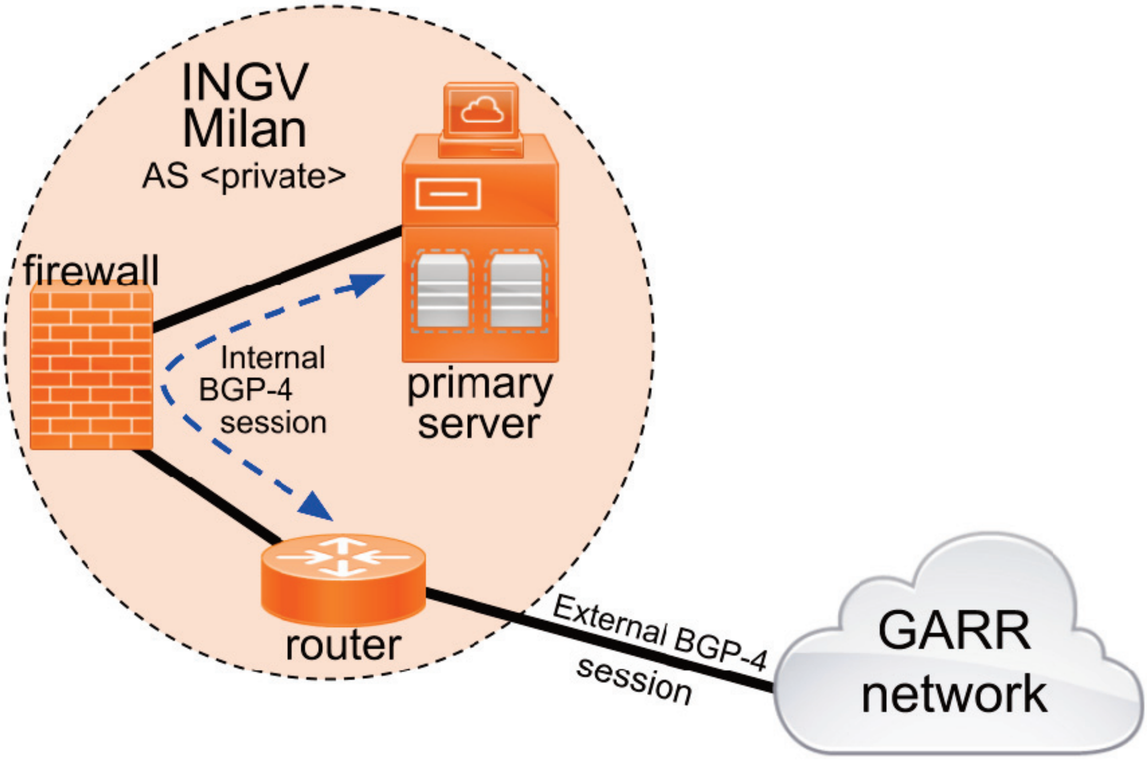
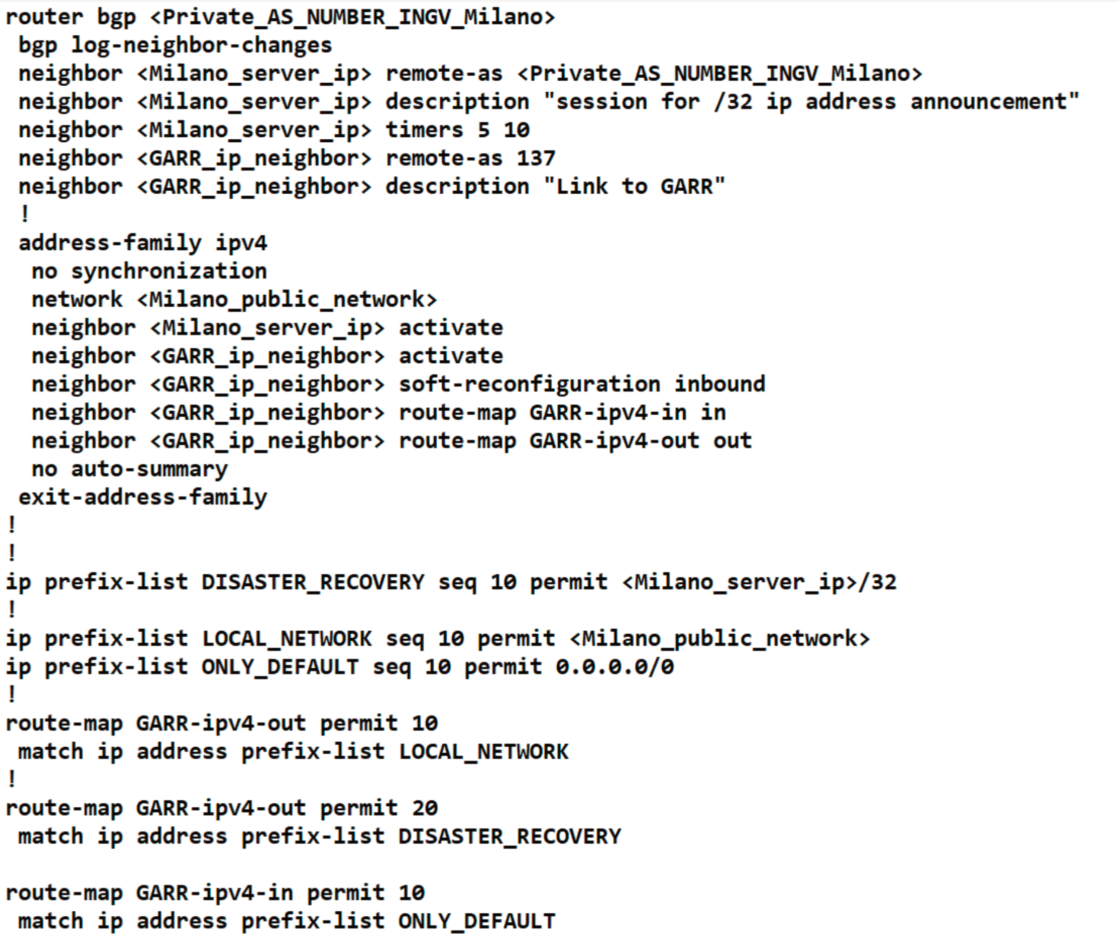


FIGURE 7. External and Internal Session in Milan.



```
router bgp <Private_AS_NUMBER_INGV_Milano>
 bgp log-neighbor-changes
 neighbor <Milano_server_ip> remote-as <Private_AS_NUMBER_INGV_Milano>
 neighbor <Milano_server_ip> description "session for /32 ip address announcement"
 neighbor <Milano_server_ip> timers 5 10
 neighbor <GARR_ip_neighbor> remote-as 137
 neighbor <GARR_ip_neighbor> description "Link to GARR"
 !
 address-family ipv4
  no synchronization
  network <Milano_public_network>
  neighbor <Milano_server_ip> activate
  neighbor <GARR_ip_neighbor> activate
  neighbor <GARR_ip_neighbor> soft-reconfiguration inbound
  neighbor <GARR_ip_neighbor> route-map GARR-ipv4-in in
  neighbor <GARR_ip_neighbor> route-map GARR-ipv4-out out
  no auto-summary
 exit-address-family
!
!
ip prefix-list DISASTER_RECOVERY seq 10 permit <Milano_server_ip>/32
!
ip prefix-list LOCAL_NETWORK seq 10 permit <Milano_public_network>
ip prefix-list ONLY_DEFAULT seq 10 permit 0.0.0.0/0
!
route-map GARR-ipv4-out permit 10
 match ip address prefix-list LOCAL_NETWORK
!
route-map GARR-ipv4-out permit 20
 match ip address prefix-list DISASTER_RECOVERY

route-map GARR-ipv4-in permit 10
 match ip address prefix-list ONLY_DEFAULT
```

FIGURE 8. The configuration of the router in Milan.

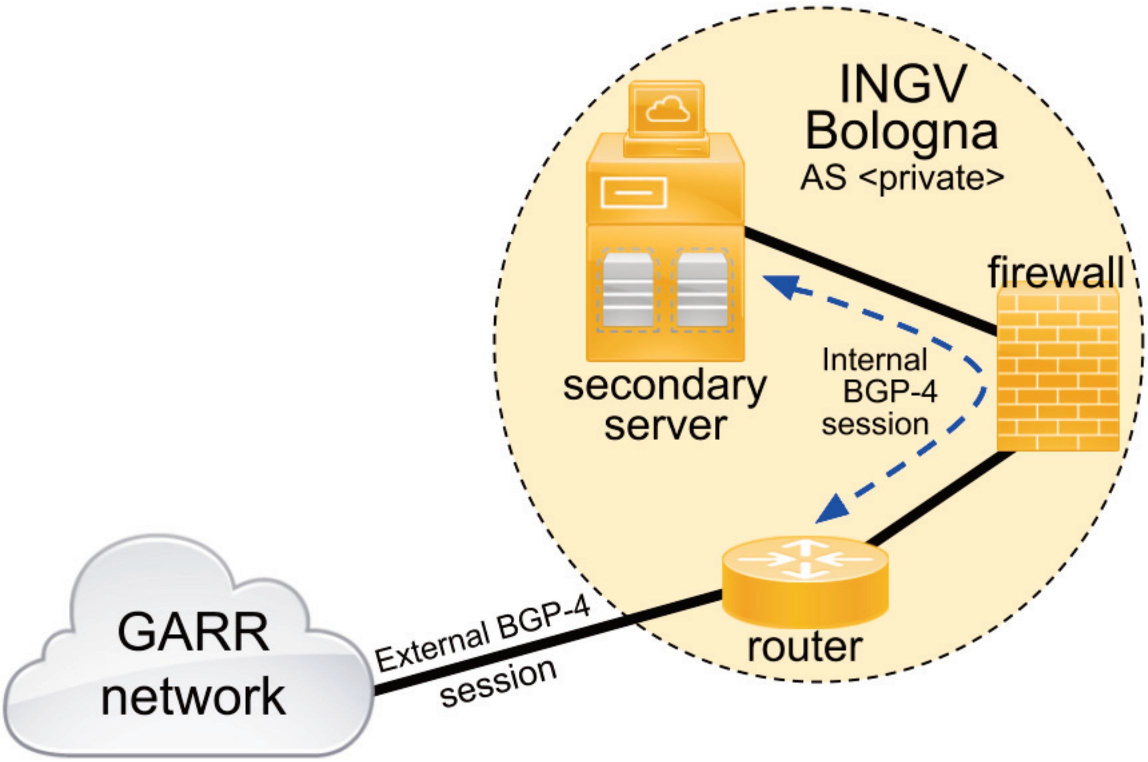
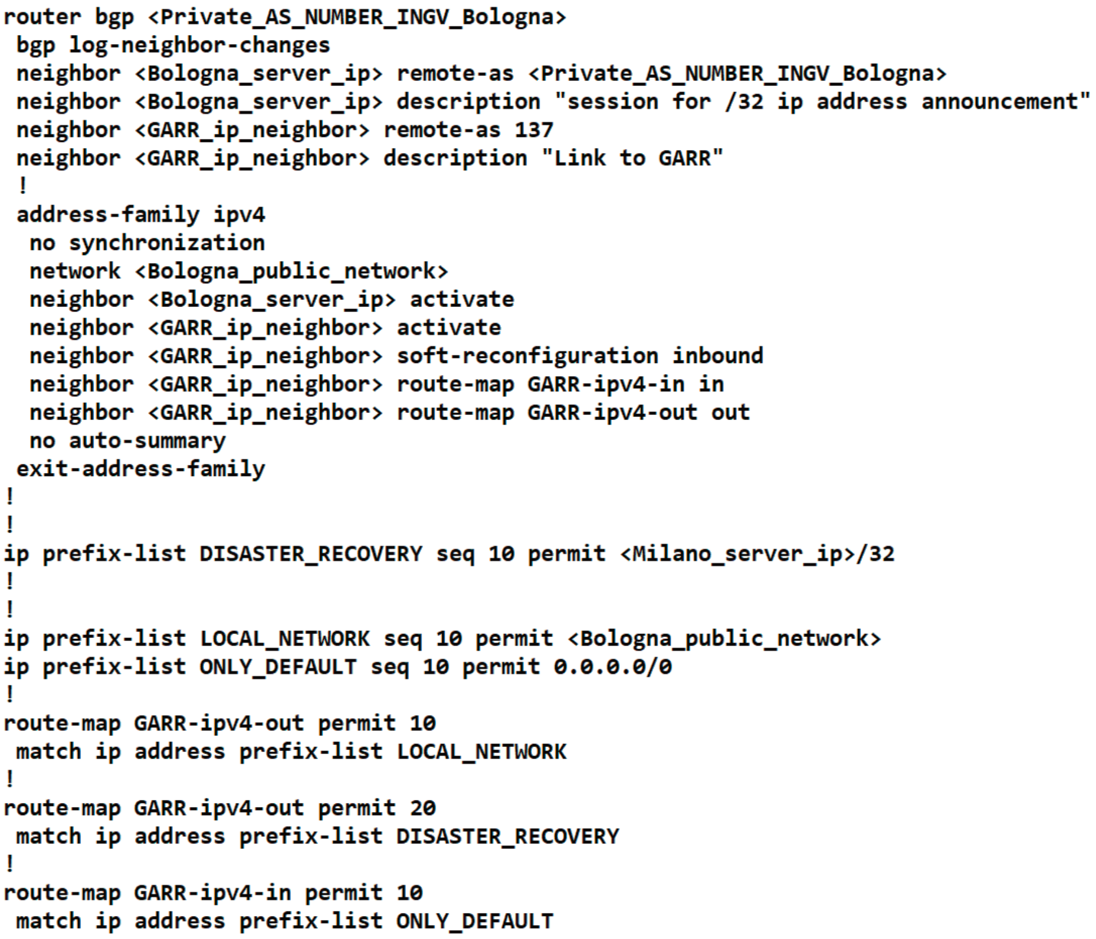


FIGURE 9. External and Internal Session in Bologna.



```
router bgp <Private_AS_NUMBER_INGV_Bologna>
 bgp log-neighbor-changes
 neighbor <Bologna_server_ip> remote-as <Private_AS_NUMBER_INGV_Bologna>
 neighbor <Bologna_server_ip> description "session for /32 ip address announcement"
 neighbor <GARR_ip_neighbor> remote-as 137
 neighbor <GARR_ip_neighbor> description "Link to GARR"
 !
 address-family ipv4
  no synchronization
  network <Bologna_public_network>
  neighbor <Bologna_server_ip> activate
  neighbor <GARR_ip_neighbor> activate
  neighbor <GARR_ip_neighbor> soft-reconfiguration inbound
  neighbor <GARR_ip_neighbor> route-map GARR-ipv4-in in
  neighbor <GARR_ip_neighbor> route-map GARR-ipv4-out out
  no auto-summary
 exit-address-family
!
!
ip prefix-list DISASTER_RECOVERY seq 10 permit <Milano_server_ip>/32
!
!
ip prefix-list LOCAL_NETWORK seq 10 permit <Bologna_public_network>
ip prefix-list ONLY_DEFAULT seq 10 permit 0.0.0.0/0
!
route-map GARR-ipv4-out permit 10
 match ip address prefix-list LOCAL_NETWORK
!
route-map GARR-ipv4-out permit 20
 match ip address prefix-list DISASTER_RECOVERY
!
route-map GARR-ipv4-in permit 10
 match ip address prefix-list ONLY_DEFAULT
```

FIGURE 10. The configuration of the router in Bologna.

The failover system, in Milan, is composed by (Figure 5):

1) the external BGP session between the Milan border router and the GARR border router
2) the internal BGP session between the **primary** server and the Milan border router

As mentioned before, the GARR routers assign a *higher local preference* to the IP address announced by the Milan router. If the local network is up, the *<Milano_server_ip>/32* is routed via the Milan router to the primary server. Otherwise, if the local network is down, the routing path of *<Milano_server_ip>/32* is switched to the Bologna router. Instead, in order to manage the switch from the primary to the secondary server, if, only, the server in Milan has a problem, but all other devices work, we activated an internal BGP session between the primary server and Milan router (Figure 5). According to this fact, the primary server announces, continuously, its own IP address (*i.e. <Milano_server_ip>/32*) to the Milan router and when a server failure occurs, the internal BGP session is broken. When the internal session is broken the announce of IP of the primary server disappears from the BGP table of the Milan router, that it is not able to exchange this information with the border router of the GARR network. In this case, the same announce appears, only, from the BGP session coming from Bologna and the BGP-4 gives precedence towards this direction.

In order to detect and manage the broken session, the command *neighbor <Milano_server_ip> timers 5 10* sets timers for the announcement of primary server in the BGP session has been used.

The default value for the hold time suggested in the BGP specification (RFC 4271; https://tools.ietf.org/html/rfc4271) is 90 seconds, and keepalives should be sent at intervals of one third the hold time (30 seconds).

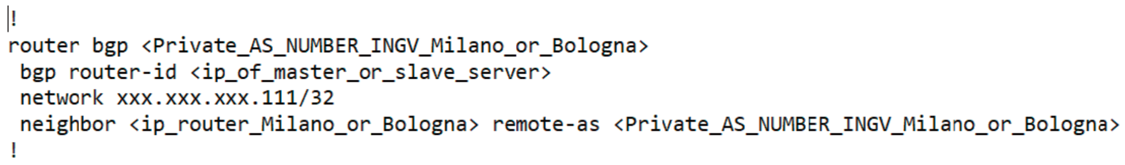
However, Cisco uses defaults of 180 and 60 seconds (https://www.cisco.com/c/en/us/td/docs/ios/iproute_bgp/command/reference/irg_book/irg_bgp4.html). So when two Cisco routers have established a BGP session and exchanged prefixes, 60 seconds later they'll each send a KEEPALIVE message. Upon reception of the keepalive by the other router, that router's hold time for the session will have counted down from 180 to 120, but it now gets reinitialized to 180. This continues every 60 seconds. To reduce the time to switch routing in case of server problem the values of 5 and 10 for keepalive and hold time have been set. In the "address-family ipv4" section is configured the Milan network announced, which sessions are active (both GARR and internal neighbors) and which networks are permitted in inbound and outbound traffic through the route-map definitions. In the "prefix-list" section is configured the networks: the <Milano_server_ip>/32, the Milan network and the default network. These networks are recalled in route maps, the <Milano_server_ip>/32 plus the Milan network in the GARR-ipv4-out route map and the default in the Garr-ipv4-in.

In Bologna, the failover system exploits similar BGP sessions (Figures 5 and 6):

1) tthe external session between the Bologna border router and the GARR border router
2) tthe internal session between the **secondary** server and the Bologna border router

In addition, the Bologna router has to manage the routing for the *<Milano_server_ip>/32* to secondary server because this IP address does not belong to Bologna network (*193.204.89.128/26*). In both cases (in Bologna and Milan), it is necessary to open on the firewall the port for BGP communication between servers and routers (i.e. port 179 of TCP protocol [https://tools.ietf.org/html/rfc793]). Further, in Bologna, it is necessary to open also the web ports for the *<Milano_server_ip>/32* in the firewall.

The primary and secondary servers were installed with the *Ubuntu Server* distribution of the Linux Operating System. In order to set up a internal session to announce the *<Milano_server_ip>/32*, we installed Quagga (https://www.quagga.net/) on the primary and secondary servers. Quagga is a routing software suite, providing implementations of OSPF, RIP, RIPng and BGP-4 for Unix/Linux platforms. The Figure 11 shows the Quagga configuration to announce the server IP ad-



```
!
router bgp <Private_AS_NUMBER_INGV_Milano_or_Bologna>
 bgp router-id <ip_of_master_or_slave_server>
 network xxx.xxx.xxx.111/32
 neighbor <ip_router_Milano_or_Bologna> remote-as <Private_AS_NUMBER_INGV_Milano_or_Bologna>
!
```

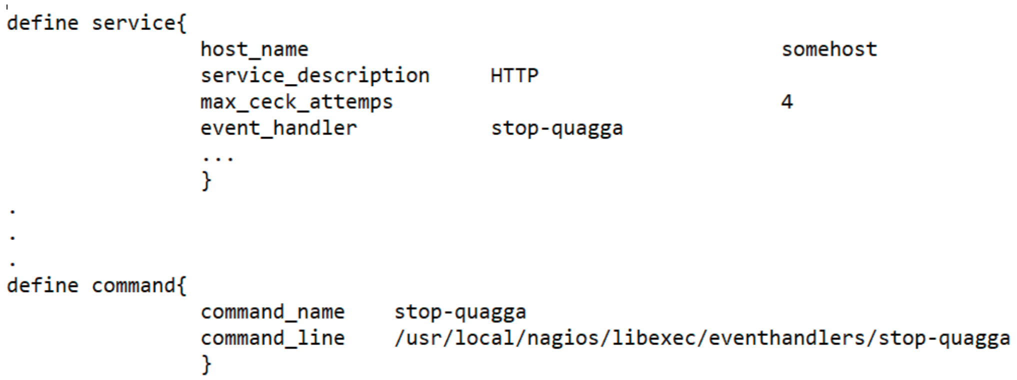
FIGURE 11. BGP configuration with Quagga on server.

dress to the INGV router. The configuration on primary and secondary server are similar. In this way, if the primary server becomes offline for any problem, automatically, the announcement of the *<Milano_server_ip>/32* goes down and the BGP switches the routing path for this IP address towards Bologna router and, then, to secondary server. If the primary server returns online, the announcement of the *<Milano_server_ip>/32* goes up and the GARR border router resets the preference for dynamic routing towards Milan router. The configuration of the primary/secondary architecture and of the internal BGP session, in particular, prevents that the crashes of the primary server makes unavailable the web service avoiding a Server Failure. Further, if the primary server returns online, the failover system redirects all the traffic towards Milan.

6. SOLVING A SERVICE FAILURE WITH CONSTANT MONITORING

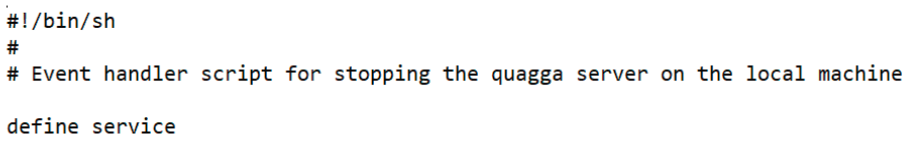
If the web service becomes unavailable, but the primary server works perfectly and the Milan network has no failure, it is necessary to find a solution able to force the routing path towards the replicated service. In order to tackle this case scenario, it is necessary to constantly monitor the web service on the primary server and to force the interruption of the announcement if a service failure occurs. In other words, this means to force the shutdown of the Quagga daemon that stops correctly the internal BGP session.

A solution is to install Nagios Core (https://www.nagios.org) on the primary server, a free and open source computer-software application that monitors systems, networks and infrastructure. Nagios offers monitoring and alerting services for servers, switches, applications and services. This software promotes the use of *Event handlers*: optional and highly customizable system commands (scripts or executables) that run whenever a host or service state change occurs. If the monitored web service becomes unavailable or gives an answer that does not fulfill the applied custom rules, it is possible to run a script to stop the Quagga daemon. It is necessary to modify the configuration for monitoring a server (Figure 8) and to specify the handler to run if a fault occurs. In Figure 9, the "*stop-quagga*" event handler is shown. It is important to highlight that "*stop-quagga*" stops, in the right way, Quagga allowing daemon to close the BGP session correctly. In this case, the BGP on the GARR router does not need to wait any timeout setup because a message of end of session is sent. This means that the routing path is immediately switched towards the secondary server. In case of crash of the web service, the monitoring on the primary server and implementation of an event handler facilitate the web service continuity on the replicated secondary server.



```
define service{
                host_name                                       somehost
                service_description     HTTP
                max_ceck_attemps                                4
                event_handler           stop-quagga
                ...
                }
.
.
.
define command{
                command_name    stop-quagga
                command_line    /usr/local/nagios/libexec/eventhandlers/stop-quagga
                }
```

FIGURE 12. The configuration of an Event handler.



```
#!/bin/sh
#
# Event handler script for stopping the quagga server on the local machine

define service
```

FIGURE 13. Example of script to stop Quagga.

7. CONCLUSIONS

INGV is a research institute geographically distributed in various Italian cities, and it is in charge of the continuous monitoring of natural seismic, volcanic and tsunamigenic phenomena in close cooperation with the National Civil Protection. In addition, INGV carry out scientific research, outreach and educational activities, at regional, national and international level. These complex, multiparametric networks, together with related research activities, generate a massive volume of seismic, volcanic and tsunamigenic data entering the *Research Data Life Cycle*. In a complex research environment heavily based on Linked Data, any interlinked data service provider must guarantee a reliable and operational activity. Our solution is based on the routing of the IP packets from the requests coming from internal and external users of our services, operating on the mechanisms that rule/control the Internet. In this sense, our system reacts to the different levels of failure acting on the mechanism of Internet without interacting with any upper levels that do not consider the structure of the Wide Area Networks themselves. In order to understand the power of our approach, we could compare with the routing service in the European Integrated Data Archive (EIDA). The routing service assists users to locate the data on server by means of a services of higher level, which can offer the user an integrated view of the whole EIDA, hiding the complexity of its internal structure and the external infrastructure of telecommunications. This routing service works at application level, abstracting from the lower levels and could not exploit the full performance and responsiveness of network, our approach works mainly at the network level and we configure the best network route according to criteria of better performance of the network connections, better site with computing power or preference to a site for reasons of institutional policies.. To test a new solution for improving the reliability of a data provider node, we designed and implemented an innovative failover system on a series of data repositories and related services hosted in the "emidius" server, the Italian Archive of Historical Earthquake Data (ASMI) and the European Archive of Historical Earthquake Data (AHEAD) among others. The service continuity was implemented by replicating the "emidius" server situated at the INGV data centre in Milan, at the INGV data centre in Bologna and a failover system was set up between the two departments. The content on the two servers is kept aligned using a series of standard protocols and applications that keeps synchronized both the filesystem and the data on the DBMS (Database Management System). The solution was made possible thanks to the geographically distributed nature of INGV that relies on interconnected data centres via the GARR network, whose personnel actively participated in experimenting the described innovative failover system. In detail, the solution is based on the combination of properly configured network devices (i.e. BGP), routing software (i.e. Quagga) and open source monitoring software (i.e. Nagios) in a primary/secondary architecture. The configuration of the BGP protocol on routers allows to very quickly divert the routing path from the primary to the replicated secondary server in case of any kind of network failure occurs. The BGP exchanges routing and reachability information among border routers on different Autonomous Systems on the Internet by means of a session based mechanism. The configurations of Quagga allow intercepting when the "emidius" server crashes (i.e. server failure) causing its unreachability, breaking the BGP session. The implementation of a monitoring handler in Nagios, stops the BGP session if a service fault occurs. Furthermore, this failover system could also be used for the primary server maintenance, as the administrator can manually stop the Quagga service, diverting the users to the secondary server, and safely operate on the primary server, all without any service interruption. Thanks to the innovative combination of these three components, the BGP, Quagga and Nagios, and thanks to the cooperation of the Consortium GARR, the adopted solution can be easily adopted by other data repositories and related services running in a similar environment. In fact, even if the analysed case study was based on a single server, with a single IP address, and running only a series of lightweight data repositories and web services, the described failover system can be easily adopted by an unlimited number of servers, independently from the type of repositories or services they run.

Acknowledgements. This paper was partially supported by the Agreement B2 2016-2017 and 2018 between the Italian Civil Protection Department (DPC) and Istituto Nazionale di Geofisica e Vulcanologia (INGV). Nevertheless, the views and conclusions reported in the paper are the sole responsibility of its authors, and should not be interpreted as necessarily representing official policies, either expressed or implied, of the Dipartimento della Protezione Civile.

REFERENCES

Berners-Lee T. (2006). Linked Data, https://www.w3.org/DesignIssues/LinkedData.html

Bizer C., T. Heath and T. Berners-Lee (2009). Linked Data - The Story So Far. Int. J. Sem. Web Info. Sys. (IJSWIS), 5(3), doi: https://doi.org/10.4018/jswis.2009081901

Clinton J., W. Hanka, S. Mazza, H. Pederson, R. Sleeman, K. Stammler, A. Strollo and T. Van Eck (2014). EIDA: The European distributed data archive for seismology, Proceeding of the 2nd European Conference on Earthquake Engineering and Seismology, Istanbul, Turkey, 24–29 August 2014, Paper Number 3322.

EPOS-IP WP6 & WP7 teams (2015). EPOS ICS-TCS Integration Guidelines - Handbook for TCS integration: Level-2, doi: https://doi.org/10.5281/zenodo.34666

Gasparini P., G. Manfredi, D. Asprone (Eds.) (2014). Resilience and Sustainability in Relation to Natural Disasters: A Challenge for Future Cities. SpringerBriefs in Earth Sciences, doi: https://doi.org/10.1007/978-3-319-04316-6

Lecarpentier D., P. Wittenburg, W. Elbers, A. Michelini, R. Kanso, P. Coveney, R. Baxter (2013). EUDAT: A New Cross-Disciplinary Data Infrastructure for Science. Int. J. Digital Curat., 8(1), doi: https://doi.org/10.2218/ijdc.v8i1.260

Locati M., A. Rovida, P. Albini and M. Stucchi (2014). The AHEAD Portal: A Gateway to European Historical Earthquake Data. Seism. Resource Letters, 85(3):727-734, doi: https://doi.org/10.1785/0220130113

Lucini B. (2014). Organizational Response to Emergencies: Italian Civil Protection and Civil Defence Service. In: Lucini B. (ed.), Disaster Res. Sociol Perspect., 55-80, doi: https://doi.org/10.1007/978-1-4020-6552-1_4

Michelini A., L. Margheriti, M. Cattaneo, G. Cecere, G. D'Anna, A. Delladio, M. Moretti, S. Pintore, A. Amato, A. Basili, A. Bono, P. Casale, P. Danecek, M. Demartin, L. Faenza, V. Lauciani, A.G. Mandiello, A. Marchetti, C. Marcocci, S. Mazza, F. M. Mele, A. Nardi, C. Nostro, M. Pignone, M. Quintiliani, S. Rao, L. Scognamiglio, G. Selvaggi (2016). The Italian National Seismic Network and the earthquake and tsunami monitoring and surveillance systems. Advan. Geosci, 43, 31-38, doi: https://doi.org/10.5194/adgeo-43-31-2016

Michelini A., G. Wotawa, D. Arnold-Arias and the ARISTOTLE Team (2017). ARISTOTLE (All Risk Integrated System TOwards The hoListic Early-warning). Geophys. Res. Abs, 19, EGU2017-11355.

Pennisi A. (2014). A brief introduction to the Italian Civil Protection System. Report 01, University of Catania and Freie Universität Berlin, 19.

Pintore S, F. Bernardi, A. Bono, P. Danecek, L. Faenza, M. Fares, V. Lauciani, F. P. Lucente, C. Marcocci, D. Pietrangeli, M. Quintiliani, S. Mazza, A. Michelini (2016). INGV data lifecycle management system performances during Mw 6.0 2016 Amatrice earthquake sequence. Ann. Geophys., 59, Fast Track 5, doi: https://doi.org/10.4401/ag-7218

Quinteros J. (2017). Routing Service: A data centre federation for the seismological community. GFZ German Research Centre for Geosciences, doi: https://doi.org/10.5880/gfz.2.4.2017.001

Valensise G., A. Amato, P. Montone and D. Pantosti (2003). Earthquakes in Italy: Past, present and future. Episodes, J. Int. Geosci., 26(3):245-249.

*CORRESPONDING AUTHOR: Luca NANNIPIERI,
Istituto Nazionale di Geofisica e Vulcanologia
Sezione di Pisa, Pisa, Italy;
email: luca.nannipieri@ingv.it