Rendere i Linked Data più affidabili con una soluzione di failover facile da implementare: il metodo FIRM

Luca Nannipieri
INGV

WORKSHOP GARR 2021

NET MAKERS

IL METODO F.I.R.M.

- **F**ailover system **I**ntegrated in **R**outing **M**anagement

- http://firm.pi.ingv.it/

- https://doi.org/10.4401/ag-8050

  “MAKING LINKED DATA MORE RELIABLE WITH A FAILOVER SERVER SYSTEM: A CASE STUDY WITH SEISMOLOGICAL DATA AT INGV.” Nannipieri L, Cacciaguerra S, Mirenna S, Locati M, Marletta M, Gucciardi E.

In termini informatici e di rete, un **failover** è un meccanismo volto a passare da un sistema primario a un sistema ridondante in standby automaticamente in caso di malfunzionamento del sistema primario.

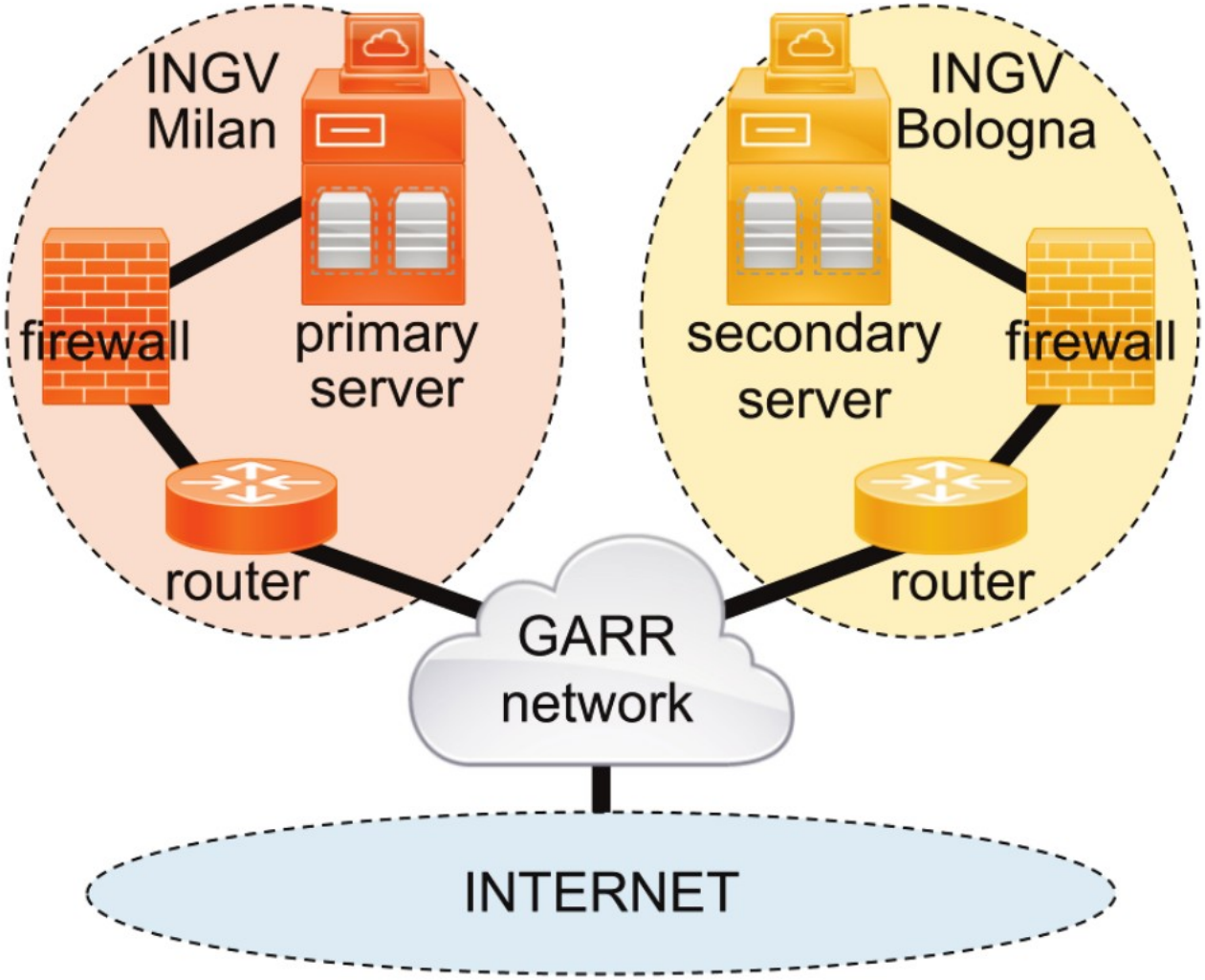
Al fine di implementare un servizio fruibile da rete su un sistema **fault tollerance**, il failover dovrebbe essere applicato a tre livelli di errore:

1) Errore di rete: il server diventa irraggiungibile per problemi dovuti a un dispositivo di rete locale o di rete wan. In questo caso, anche se il server funziona correttamente, il suoi servizi diventano irraggiungibili.

2) Errore del server: il server non funziona a causa di un errore del sistema operativo o di un arresto anomalo dell'hardware.

3) Errore di servizio: il server è attivo e funzionante ma i suoi servizi hanno smesso di funzionare, di solito a causa di problemi sul lato software.

Caso di studio INGV

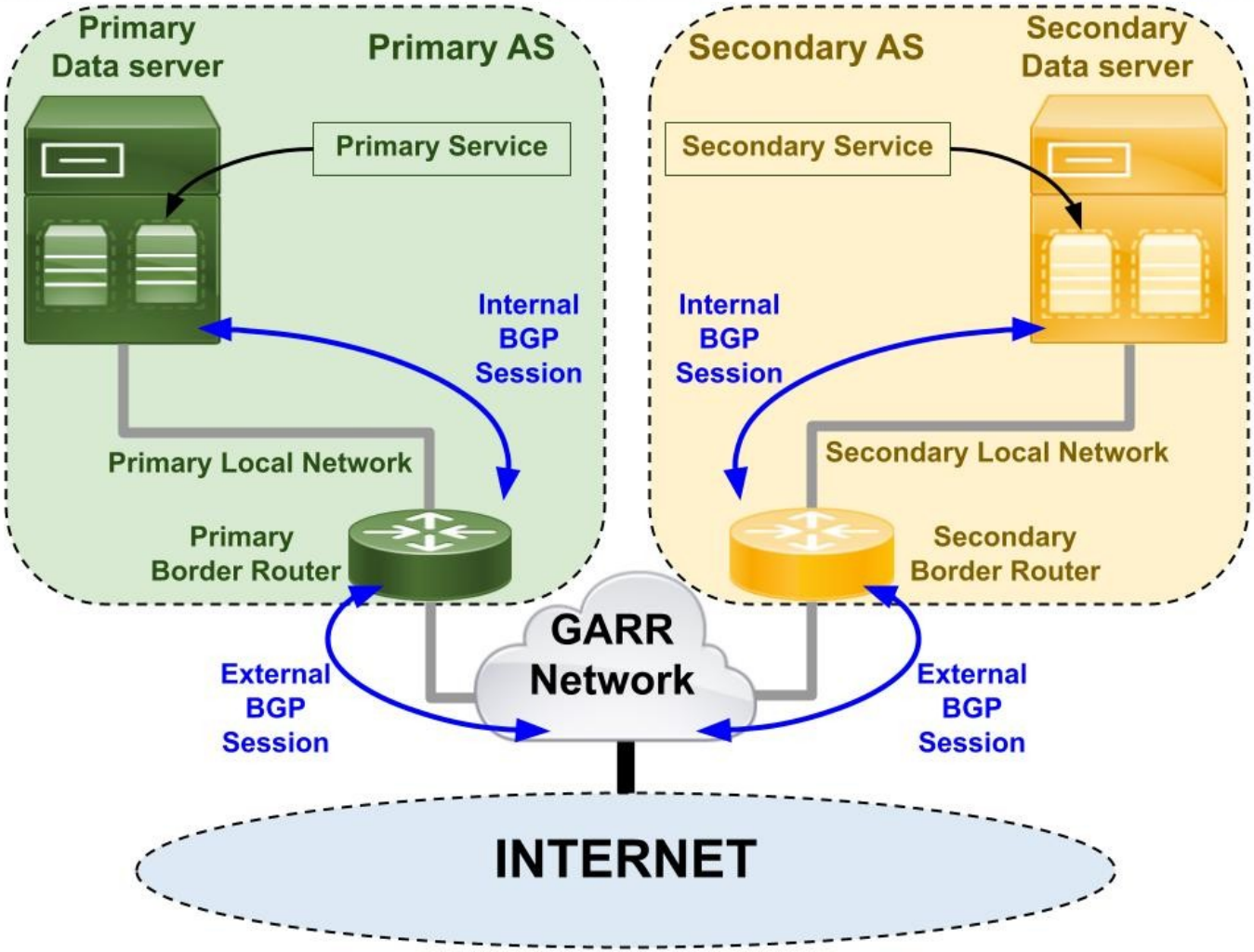


1) Ovviare alle interruzioni di rete con il protocollo BGP

L’idea di fondo è che il server primario a Milano e il secondario a Bologna annuncino con il protocollo BGP l’indirizzo IP del server primario dove risiede il servizio di cui si vuole aumentare l’affidabilità. Entrambi lo annunciano al router di bordo del proprio data center che lo riannuncia al GARR insieme alla propria rete. Il router del GARR avrà impostato un peso maggiore sull’annuncio del singolo indirizzo IP proveniente dalla sessione con il data center di Milano dove è il server primario in modo che quando è presente il protocollo BGP privilegia quella rotta.

SCHEMA DEL METODO

INGV MILANO



INGV BOLOGNA

CONFIGURAZIONE ROUTER GARR

```
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
```

CONFIGURAZIONE ROUTER INGV MILANO

```
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
```

CONFIGURAZIONE ROUTER INGV 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
```


WORKSHOP GARR 2021 NET MAKERS

CONFIGURAZIONE SERVER PRIMARIO E SECONDARIO

Sui server primario e secondario viene installato il software quagga (https://www.quagga.net/). Quagga è una suite software per implementare i protocolli di routing su server linux tra cui il BGP.

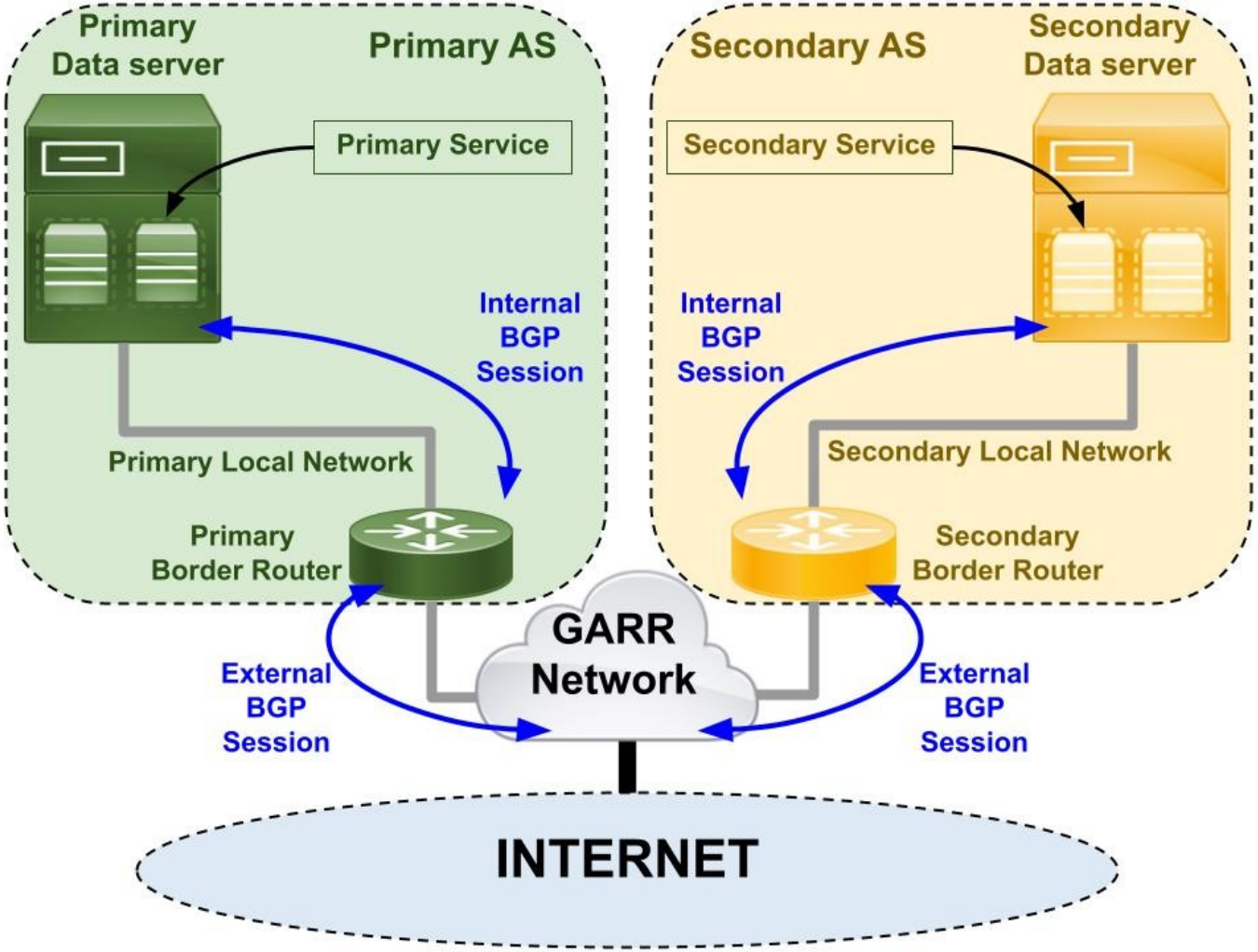
Di seguito la configurazione del BGP per il software quagga sui server:

```
!
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>
!
```

Attivazione del BGP come descritto interviene in caso di problemi di rete locali o sulla wan

INGV MILANO

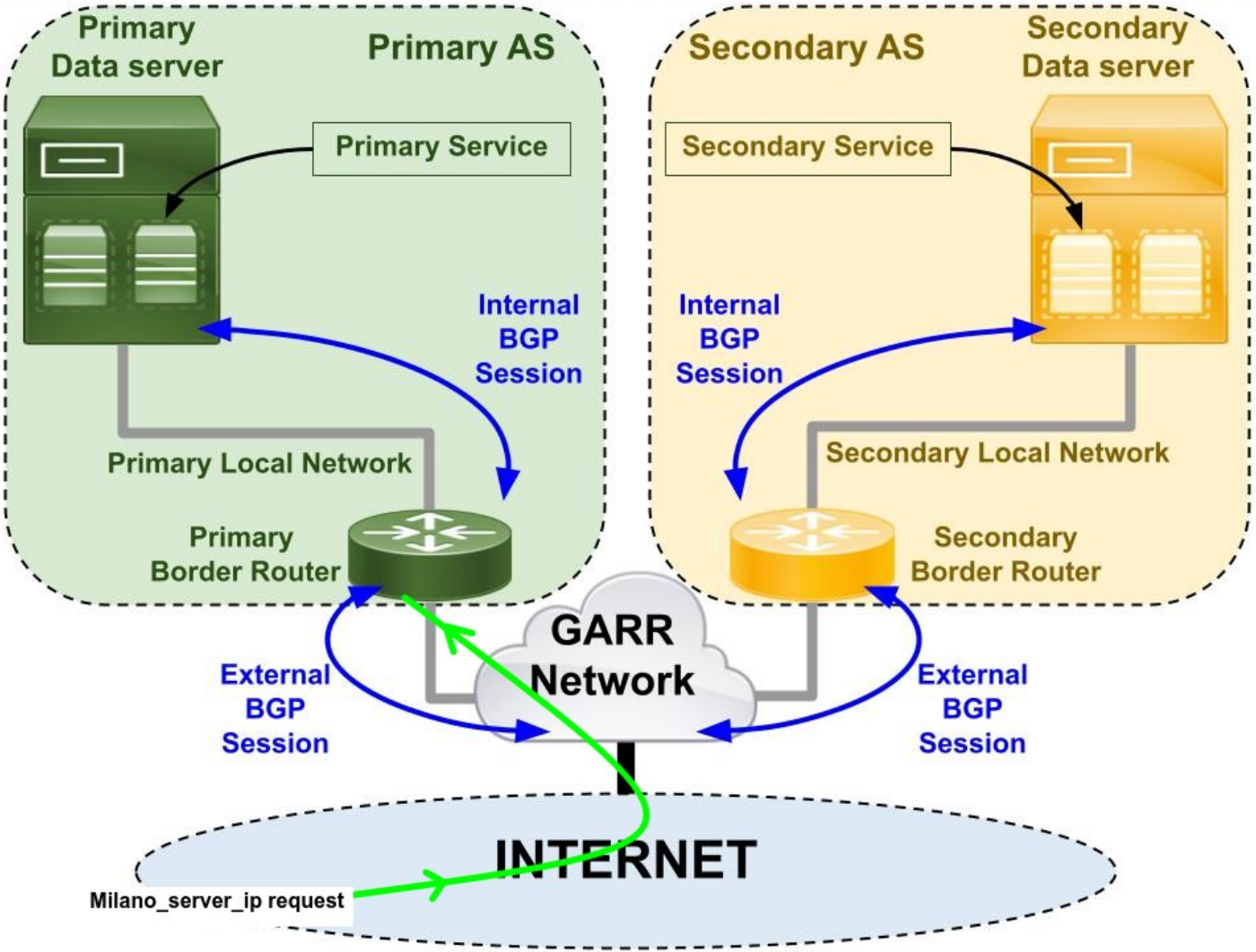
INGV BOLOGNA



Caso di funzionamento regolare

INGV MILANO

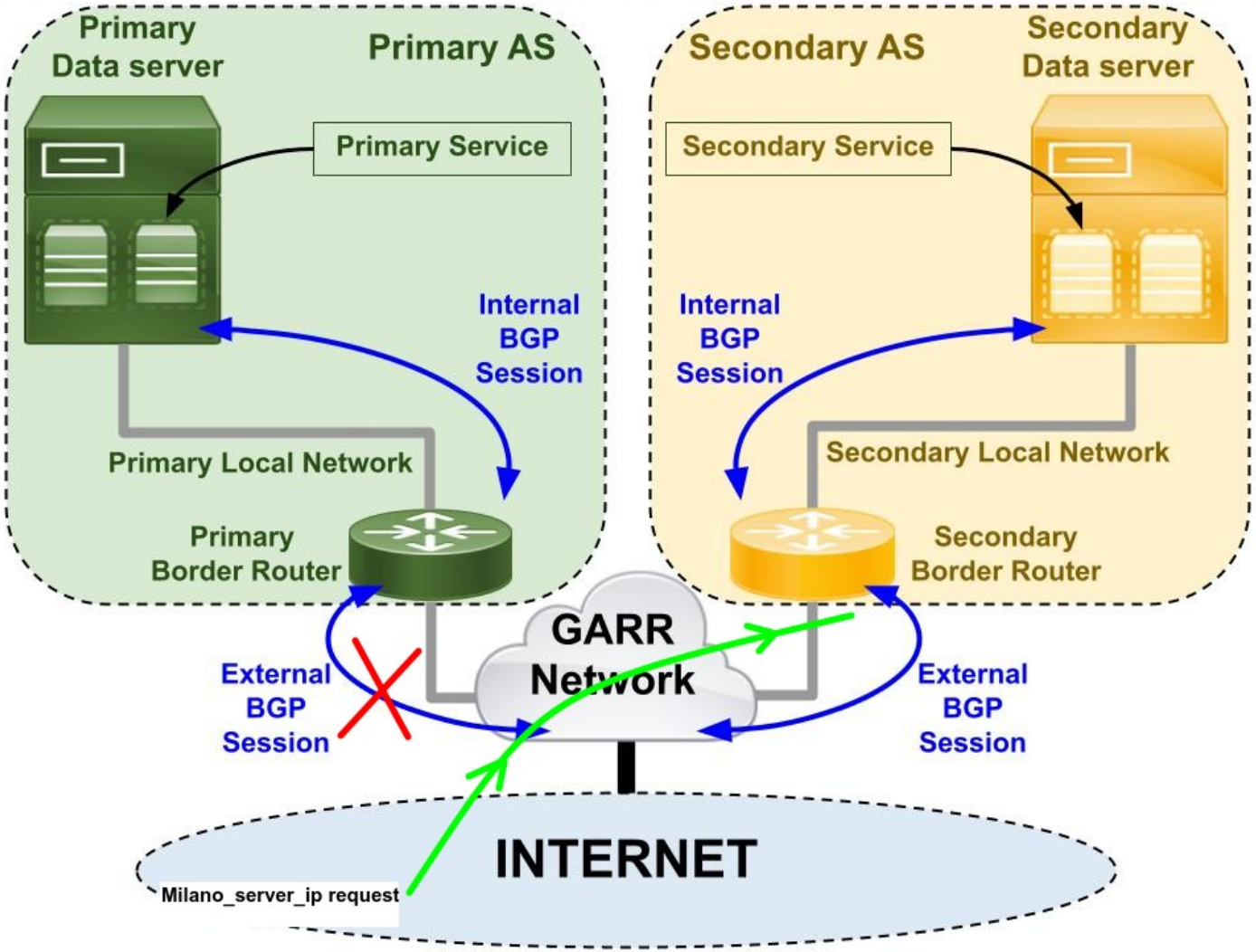
INGV BOLOGNA



Caso malfunzionamento link GARR

INGV MILANO

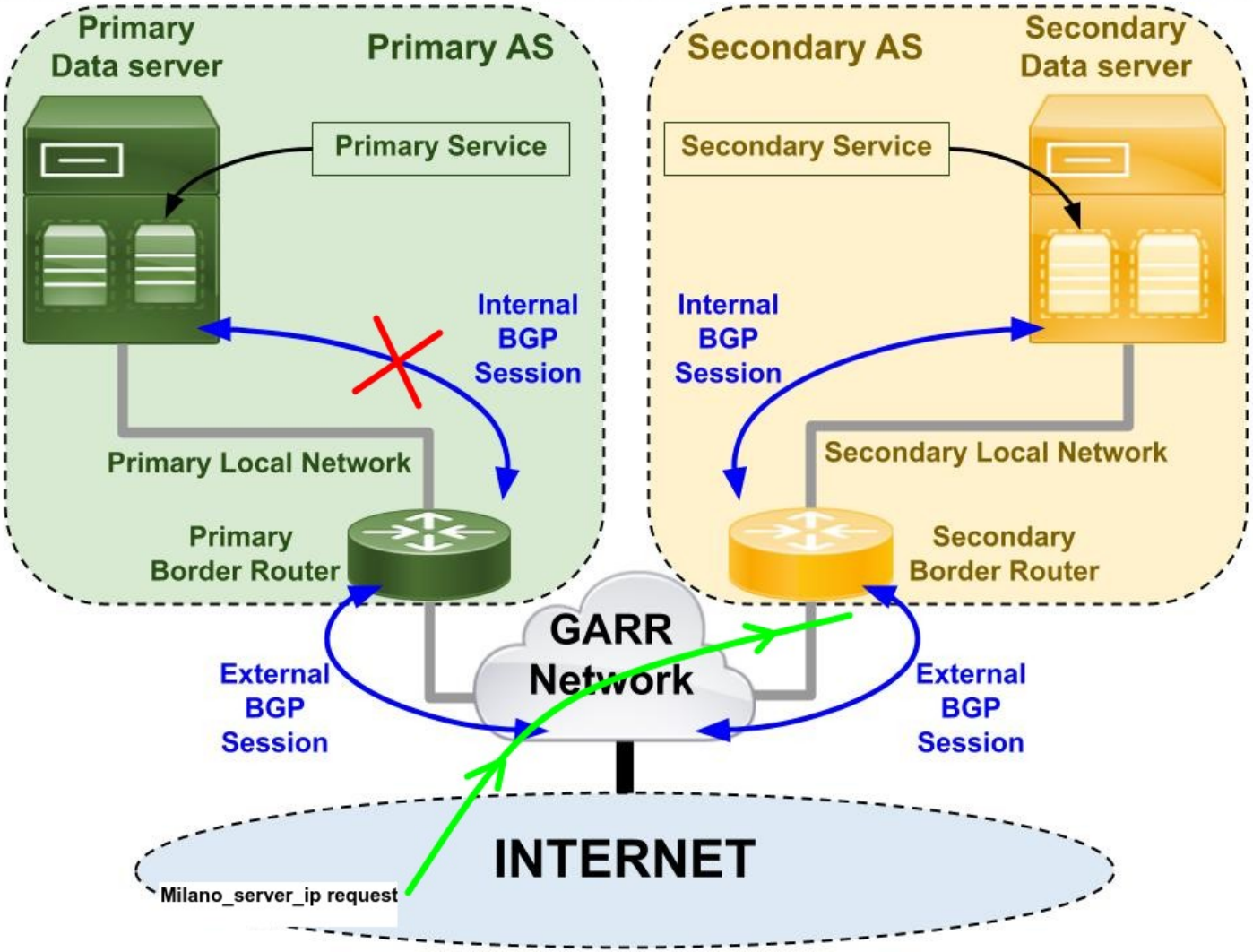
INGV BOLOGNA



Caso malfunzionamento link interno

INGV MILANO

INGV BOLOGNA



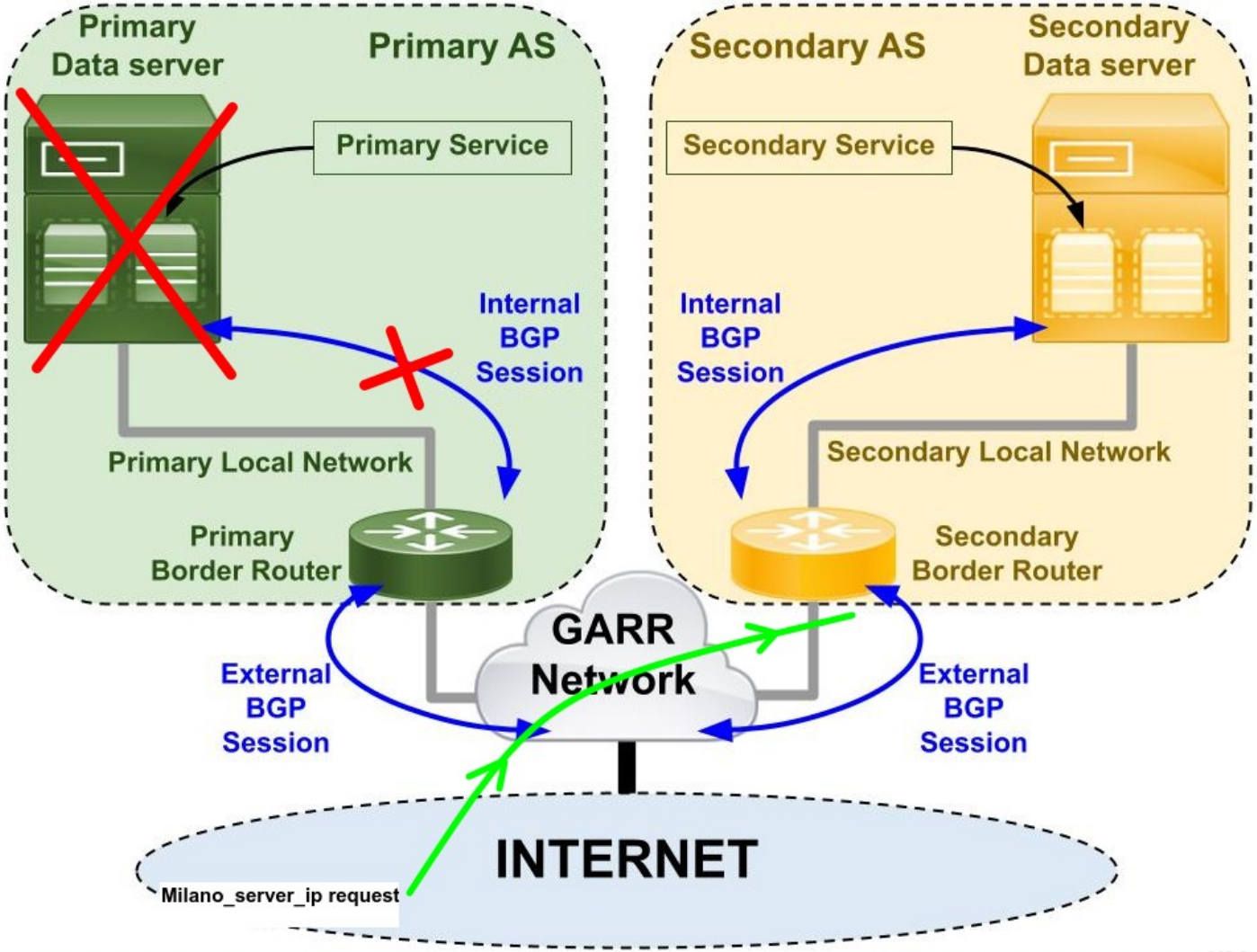
2) Ovviare agli errori del server (S.O. o Hardware)

Quando si verifica un problema al Sistema Operativo del server o al suo hardware, si ricade nella stessa situazione descritta precedentemente quando non funziona un link di rete interno. Il comportamento è lo stesso perché comunque viene a mancare l’annuncio del singolo IP del server primario di Milano e quindi il routing viene dirottato sul server seconfario di Bologna come mostra la successiva slide

Caso malfunzionamento server

INGV MILANO

INGV BOLOGNA



3) Ovviare agli errori dei soli servizi

Quando si verifica un problema ai soli servizi che interessano gli utenti, ma il server continua a funzionare per tutto il resto (annuncio BGP compreso) è necessario intervenire con un altro software da installare sul server primario. Si installa nagios sul server primario (https://www.nagios.org) e lo si configura per monitorare i servizi interessati.

Al momento che uno di questi servizi si interrompe si configura Nagios per eseguire uno script che arresta il servizio quagga e di conseguenza l'annuncio dell'indirizzo IP del server primario a milano

Esempio di monitoraggio del servizio web e definizione dell’evento “stop-quagga”

```
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
                }
```

Manutenzione server primario

Una volta configurato, il sistema può essere utilizzato anche per la manutenzione del server primario. Interrompendo il servizio quagga, il routing viene immediatamente dirottato sul server secondario a Bologna. Finita la manutenzione, riattivando il servizio quagga, il traffico verso l’IP del server primario ritorna su Milano.

Sincronizzazione dei server

La sincronizzazione del database tra i due server che sta alla base del servizio viene effettuata tramite un tunnel Ipsec, in questo modo il traffico può avvenire tra 2 ip privati risparmiando l’utilizzo di un ulteriore ip pubblico sul server slave.

Punti di forza e vincoli del metodo FIRM

- L’unico vincolo di questo metodo è che i 2 data center devono avere la connessione ad internet tramite lo stesso ISP (Internet Service Provider).
- I punti di forza sono la semplicità di implementazione (rispetto all’ annoso e complicato problema della continuità del servizio) e il basso costo. In definitiva si tratta di modificare di poco la configurazione BGP del router dell’ISP e dei router per i collegamenti dei data center, unita all’installazione di quagga sui server e l’installazione di nagios sul server primario.

- Forse il metodo FIRM è uno dei primi esempi di introduzione del protocollo BGP anche per il routing interno dei data center (implementazione nel 2018 e pubblicazione nel 2019). Effettivamente in questo modo si riesce a mutuare meglio gli eventi interni alla prorpia rete verso l’esterno se si utilizza lo stesso protocollo di routing dinamico.

GRAZIE PER L’ATTENZIONE