HCLSoftware

HCL BigFix

Get More Done with BigFix

BigFix Relay Affiliation and Failover Strategies for Resiliency

Upcoming BigFix Webinars & Events

Join us for these upcoming webinars…

5-January

Get More Done with BigFix – BigFix Relay Affiliation and Failover Strategies for Resiliency

11-January

BigFix Briefing Room – What's New in January

19-January

Get More Done with BigFix – Virtual Segregation of Your BigFix Environment

Find more webinars at our website…

HCL BigFix

Your Presenters…

John Talbert

Director, BigFix Professional Services

Rhonda Studnick Kaiser

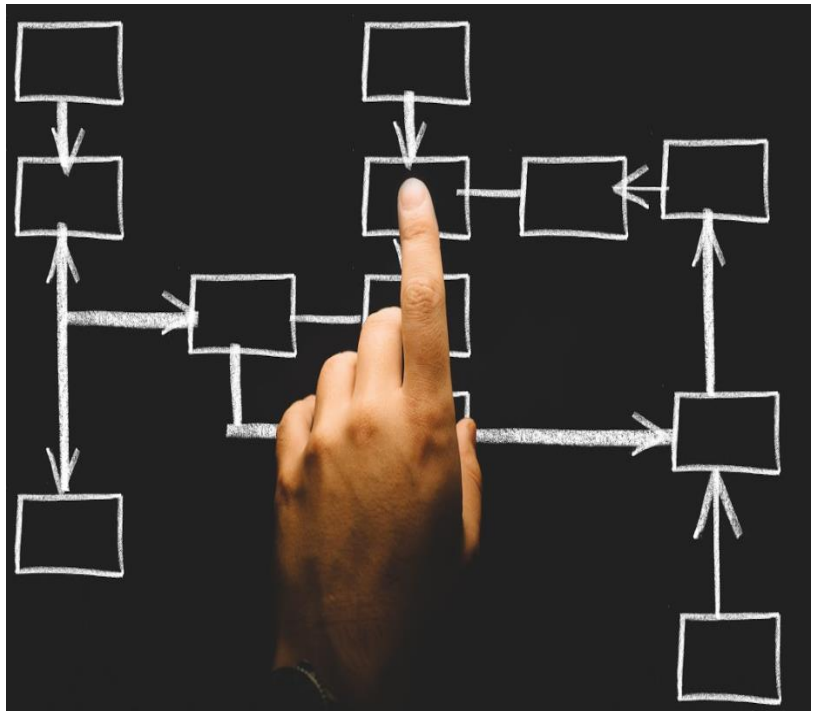
Director, BigFix Customer Experience

Jeff Schafer

BigFix Accelerated Value Program (AVP)

Types of Relays

- **Site Level Relays (aka Leaf Relays)**
- **Top Level Relays**



Relay Selection Methods

- **Manual**
- **Automatic**

Automatic Relay Selection Benefits

- Automates clients finding newly deployed relays and moving off decommissioned or “down” relays
- Reduces manhours configuring clients
- Reduces clients falling into “black hole” where they can’t find a relay (*if configured correctly)
- Helps ensure clients are always using the closest relay on the network, the least network hops away
- Provides automated failovers and failbacks when configured properly

Relay Network Requirements

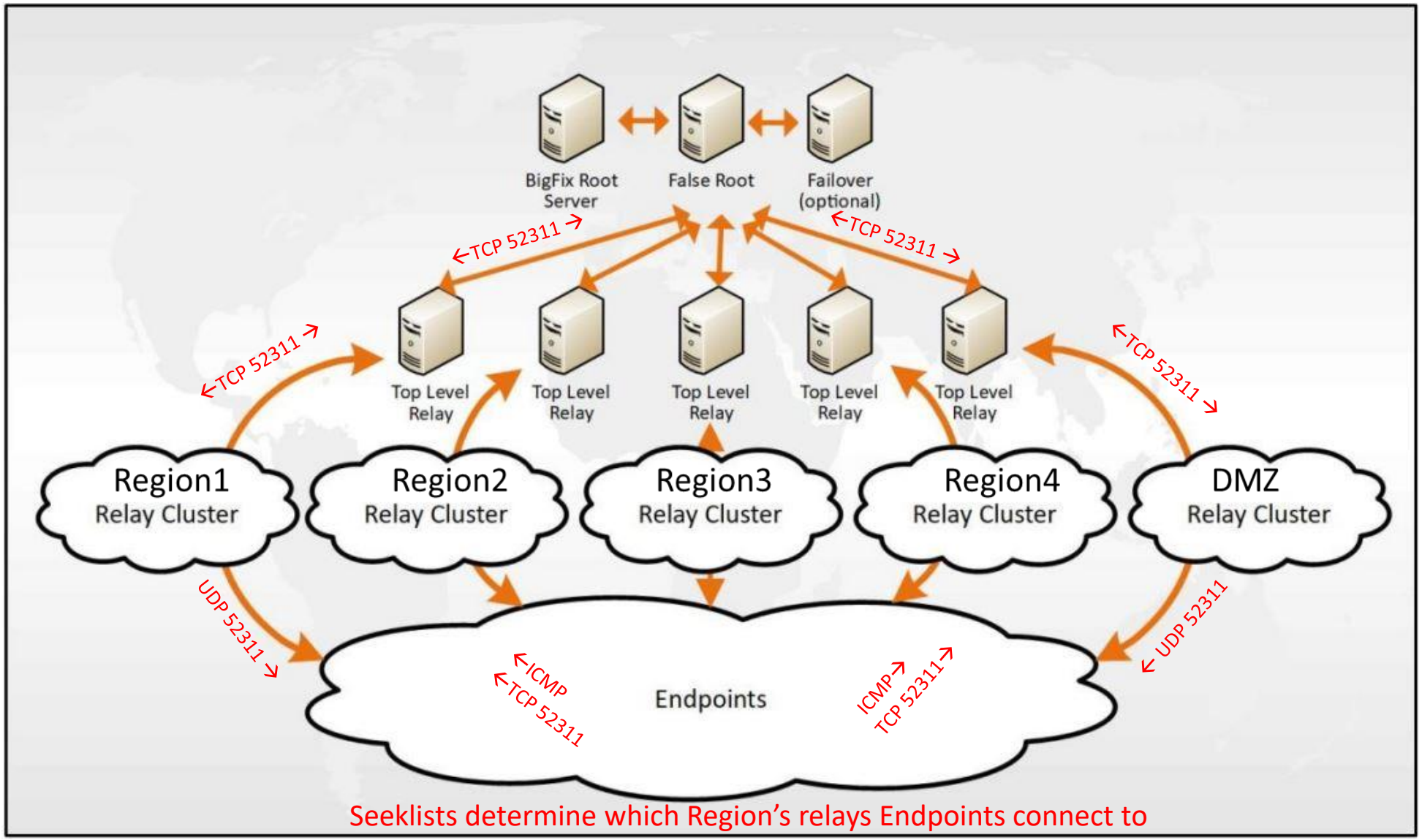
MANUAL RELAY SELECTION

- TCP 52311 **bi-directional** between relays (and relay(s) to core server)
- TCP 52311 **incoming** from endpoints
- UDP 52311 **from relay to endpoints** (notification port)

AUTOMATIC RELAY SELECTION

- TCP 52311 **bi-directional** between relays (and relay to core)
- TCP 52311 **incoming** from endpoints
- UDP 52311 **from relay to endpoints** (notification port)
- ICMP **from endpoints to all potential relays**

Automatic Relay Selection



HCL SOFTWARE

How Automatic Relay Selection Works

- **Clients use their seeklist setting and reference their local relays.dat file to find relay affiliation groups that match their seeklist.**
- **ICMP is sent out with a TTL of 0 to all relays in their first seeklist group**
  - **If a relay responds, client stops searching and registers with relay.**
  - **If multiple relays respond, client will choose relay that responds quicker (~load balancing~)**
- **If no relays respond with a TTL of 0, client then goes out 1 router hop and does ICMP ping again with a TTL of 1.**
  - **If a relay responds, client stops searching and registers with relay.**
  - **If multiple relays respond, client will choose relay that responds quicker (~load balancing~)**
- **This continues up to the max TTL setting value**
- **If no relay is found after reaching max TTL value, and a failover relay or relay list is specified, client will use that setting to choose a relay**
- **If no failover relay is reachable, client will use core server by default or whatever is specified in the admin tool/masthead setting**

Relays.dat

- Maintains a list of all relays in organization and their affiliation group
- Located in \BES Client__BESData\actionsite on all endpoints. Updated on all systems every time a new relay is added/removed or modified in organization
- Default affiliation group of all relays is the wildcard (*) “unaffiliated” group
- Default seek list on all endpoints is the wildcard (*) group
- Enabling Automatic Relay selection on clients with no seek list setting assigned to endpoints and no affiliation group assigned to relays = all relays in organization are “fair game” to be used by all endpoints (not optimal for larger organizations)
- Relays.dat parser utility can read contents of relays.dat (Google: “Relays.dat parser download”) https://bit.ly/33QrF1s
  - Usage: `ParseRelaysDotDat.exe Relays.dat (from directory where relays.dat or copy is located)`

```
Name: Relay1.company.org
Affiliation: Unaffiliated
Name: Relay1.company.org
Port: 52311
Priority: 0
Weight: 0
Name: Relay2.company.org
Affiliation: Region3
Name: Relay1.company.org
Port: 52311
Priority: 0
Weight: 0
```

Key Client Settings- Automatic Relay Selection

- **_BESRelay_Register_Affiliation_AdvertisementList (relays)**
- **__RelaySelect_Automatic (endpoint)**
- **_BESClient_Register_Affiliation_SeekList (endpoints)**
- **_BESClient_RelaySelect_MaximumTTLToPing (endpoints)**
- **_BESClient_RelaySelect_ResistFailureIntervalSeconds (endpoints)**
- **_BESClient_RelaySelect_IntervalSeconds (endpoints)**
- **_BESClient_RelaySelect_AlwaysOnIPListChange (endpoints)**
- **_BESClient_RelaySelect_FailoverRelayList (endpoints)**

  (OR)

- **_BESClient_RelaySelect_FailoverRelay (endpoints)**
- **NOTE: There are more settings but these are the main ones we like to see most customers utilize. The rest we like to work with customers on a case by case basis as needed.**

Set Affiliation Group Setting on Relays

- **Allow ICMP Incoming from Endpoints to Relays (used to determine closest relays)- CRITICAL STEP BEFORE CONTINING**
- **Assign Relay Affiliation group to relays first before enabling Automatic relay selection on clients**
  - **_BESRelay_Register_Affiliation_AdvertisementList**
  - *Example Value: Region1*

    ** Relays can be members of multiple affiliation groups using semicolon separated list*

    - *Example Value:* ***Region1;Region2;Region3;Region4;****
    - *Note:* ******* *is the default wildcard group all relays are members of including existing relays that have no affiliation group assigned. You can/should remove this * to make sure it's never used by a default endpoint set to automatic relay selection that has no seek list assigned.*

Set Seek List Setting on Endpoints

- **Assign Seek List to endpoints (after relays have already been assigned affiliation groups)**
  - **_BESClient_Register_Affiliation_SeekList**

    Example Value: ***Region1;Region2;Region3;DMZ***
    - *Clients can and usually do have multiple seek lists assigned using semicolon separated list)*
    - *If no relays found in first group, it moves to second, third etc…*
    - *Note.* ***Do not*** *assign seeklist setting on relays. Relay(s) to Relay(s) communication should always be manual*

Key Client Settings

- **Increase/decrease maximum hop count (TTL) as needed**
  - **_BESClient_RelaySelect_MaximumTTLToPing = #** (default 20)
  - TIP: Monitor “Distance To BES Relay” property to adjust TTL to lowest value needed after starting highLower value means clients give up sooner searching each affiliation group in their seek list = less network traffic but too low means they could miss a relay (ie external systems)
- **Keep clients from selecting new relays too quickly (ie during server maintenance)**
  - **_BESClient_RelaySelect_ResistFailureIntervalSeconds**
  - **Default: 600 seconds (10 minutes)**
  - **Consider raising to 1800 seconds (30 minutes) or longest time you can tolerate**
- **Set the time when client does a fresh automatic relay selection**
  - **_BESClient_RelaySelect_IntervalSeconds**
  - **Default: 21,600 seconds (6 hours)**
  - **Consider raising to 43,200 (12 hours)**
  - **Only consider lowering during periods when adding new relays to network**

Key Client Settings

- **Control what happens when client changes IP address/network**
  - **_BESClient_RelaySelect_AlwaysOnIPListChange**
  - **Default: 0 (disabled) = won't initiate a new relay selection/search**
  - **Consider setting to 1 if you have clients that may change IP addresses while powered on (ie laptops moving around or VM's changing networks frequently)**
  - **Clients will do a fresh automatic relay selection when IP address changes, when enabled**

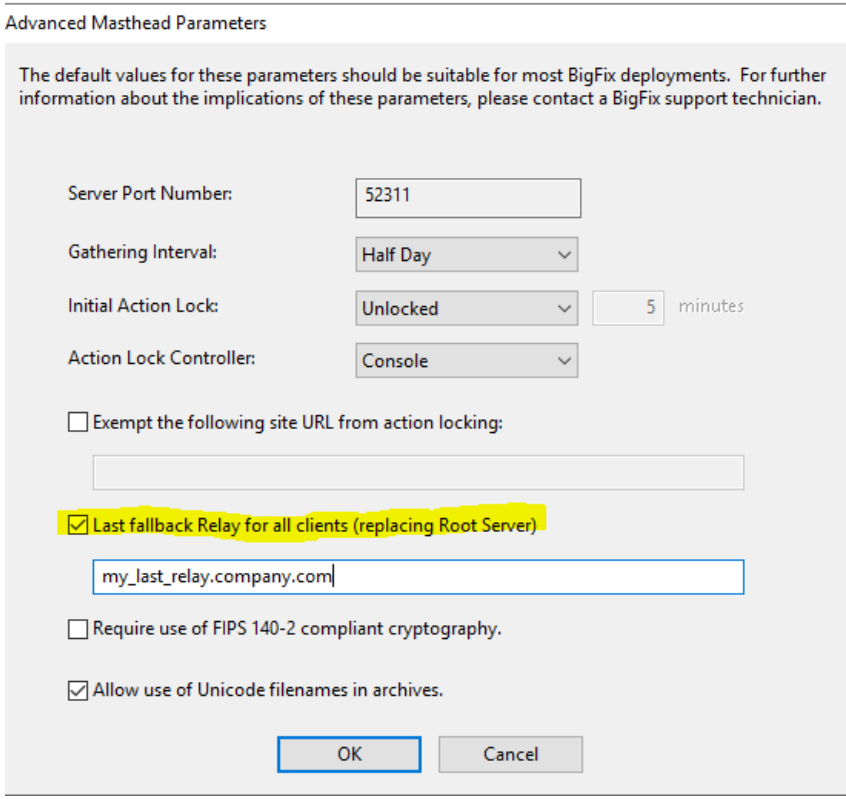
Handle Failovers when Automatic Selection Fails

- **Control what happens when client exhausts relay search using a list of failovers**
  - **_BESClient_RelaySelect_FailoverRelayList**
  - **Default: Doesn't exist**
  - **Value: list of IP's or FQDN (separated by commas) of relays to use (in order) when automatic selection can't find relay**
  - **Example: relayfailover1.company.org,144.32.22.3,relayfailover3.company.org**

  OR

- **Single relay failover option**
  - **_BESClient_RelaySelect_FailoverRelay**
  - **Default: Doesn't Exist**
  - **Value: A single relay IP or FQDN**

NOTE: Bigfix Administration tool (Edit Masthead option) has a setting to control last possible relay to use if the above fails to find relay. Otherwise, Core will be used. Recommend setting a relay every system in org can get to that's only used for that purpose.



HCL SOFTWARE

Key Client Settings

- **Enable Relay Automatic Selection on Clients**
  - **__RelaySelect_Automatic = 1** (Two underscores in front)
  - Remember: With no seek list assigned, clients will look for relays in the * affiliation group (default for relays)

  - **Enable Automatic relay selection only after first client registration (not on install)**
    - **REASON: Clients don't have a relays.dat file until they register once with a relay manually. No relays.dat = no relays to choose from = trying to connect to your core server on install.**
- **Set this last after getting relays configured**
- **Generally, the last item in action script after the settings that set all the other client settings first (ie seeklist, report intervals, failover list etc)**

Create Policy Fixlets to Assign settings

- **Example Actionscript for a relay: (to give you ideas)**

setting "Relay_Type"="REGION 1 SITE RELAY" on "{now}" for client

setting "_BESGather_Download_CacheLimitMB"="20480" on "{now}" for client

setting "_BESRelay_Register_Affiliation_AdvertisementList"="REGION1" on "{now}" for client

Example Actionscript for a client: (to give you ideas)

setting "_BESClient_Register_Affiliation_SeekList"="REGION1,DMZ" on "{now}" for client

setting "__RelaySelect_Automatic"="1" on "{now}" for client

setting "_BESClient_RelaySelect_FailoverRelayList"="192.168.10.3,relay2.myorg.net,relay5.myorg.net"

//Check the individual values of the settings.

if { not (exists settings "_BESClient_RelaySelect_IntervalSeconds" whose((exist value of it) and (value of it as string as integer >= 43200)) of client)} setting "_BESClient_RelaySelect_IntervalSeconds"="43200" on "{parameter "action issue date" of action}" for client

endif

if {not (exists settings "_BESClient_RelaySelect_MaximumTTLToPing" whose((exist value of it) and (value of it as string as integer >= 15)) of client)} setting "_BESClient_RelaySelect_MaximumTTLToPing"="15" on "{parameter "action issue date" of action}" for client

endif

Create Analysis Properties to Read all client and relay settings

It’s a good idea to create properties to read the values of the key settings

Allows you to monitor your policies/settings of endpoints/relays in console and web reports! (search first though to make sure they don’t already exist)

Example Property Relevance to pull Affiliation group setting from Relays

if (not exists relay service) then "N/A" else if exists settings whose(name of it = "**_BESRelay_Register_Affiliation_AdvertisementList**") of client then value of setting "**_BESRelay_Register_Affiliation_AdvertisementList**" of client else "None“

Example Property Relevance to pull Seek List setting from Non-Relays

if (exists relay service) then “N/A” else if exists settings whose(name of it = “**_BESClient_Register_Affiliation_SeekList**”) of client then value of setting “**_BESClient_Register_Affiliation_SeekList**” of client else “None”

Consider TCP Settings for Busy Relays

Lower TCP TimeWaitDelay from Default (120 seconds)

- **HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\ Services\TCPIP\Parameters**
- **REG_DWORD** value named **TcpTimedWaitDelay**. Set the value to **30** (seconds)
- Benefit: Helps avoid port exhaustion on busy relays

Adjust MaxUserPort

- default start port is 49152, and the default end port is 65535
- **netsh int ipv4 set dynamicport tcp start=1025 num=65535**
- Benefit: Helps avoid port exhaustion on busy relays

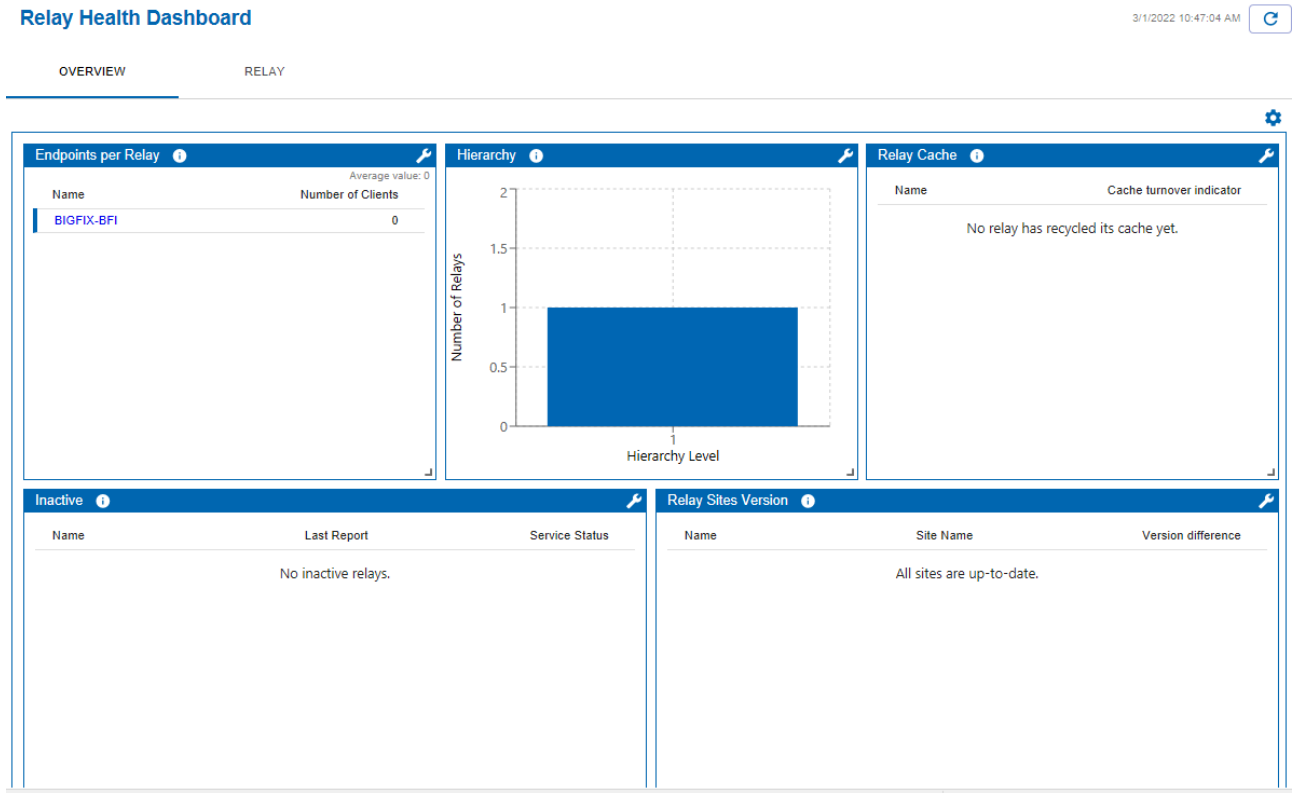
Consider Raising Client Report Interval from Default

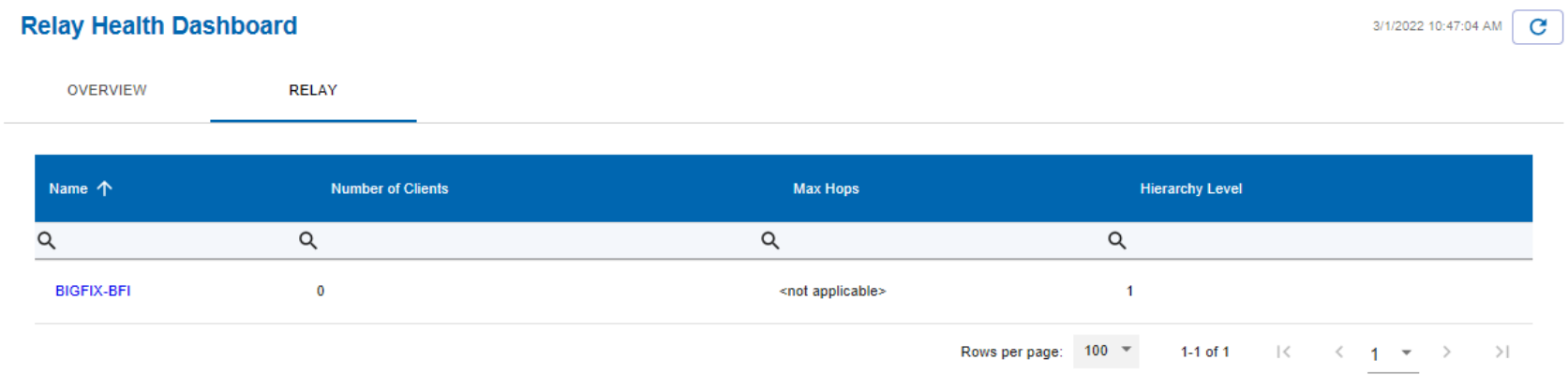
Raise to 5-15 minutes

- **_BESClient_Report_MinimumInterval**
- **Default: 60 (seconds)**

- Too low = constant filldb backlog on core server during normal operations
- Too high = less real-time visibility in console
- Find balance that keeps filldb from filling up during normal operations. Start higher, and lower until you find sweet spot. May take a few weeks to discover best value.
- Very important for larger customers with slower disk on core or customers that use remote SQL servers instead of co-installed on core.
- See our Bigfix Capacity Guide to tweak SQL and Bigfix core server settings for optimal Filldb Performance
- This setting is important regardless if you use automatic relay selection or not

Relay Health Dashboard





HCL SOFTWARE

Summary of Suggestions and Requirements

- CRITICAL STEP WITH AUTOMATIC RELAY SELECTION- Allow incoming ICMP to Relays (used by endpoints to determine closest relays)- WILL NOT WORK WITHOUT
- Keep it as simple as possible. Just because there's a client/relay setting available, doesn't mean you need to modify/add it!
- You've worked with network team to strategically plan out where and how many relays need to go where. Remember: Every customer network is different. What works good for one customer may not work good for another.
- Try to get site relays as close to endpoints as possible. Use existing hardware/VM's as much as possible to reduce cost. Don't let clients talk to Top Level Relays. TLR's job is to talk to site relays and core only.
- Use the Relay Health Dashboard in BES Support site to monitor relay info
- Plan out and create baseline open-ended policy fixlets for applying client and relay settings
- Relays can now handle up to 5000 endpoints from 10.0.5 and on but that doesn't mean you should do it. This is possible only if OS and relay configured properly to accept that many connections. Consider half that, especially if you want relays to handle fail-over clients if other relays are down.
- Don't forget relay cache sizes, filldb buffer sizes, client download cache sizes, client report intervals
- NEVER USE LOAD BALANCERS TO MOVE CLIENTS BETWEEN RELAYS!!! (they make things worse)
- Review our Peer Nest and Persistent Connection features for special situations/networks
- NEVER put a public facing relay in DMZ without enabling Relay Authentication! (Security Threat)
- CONSIDER Services contract or AVP to help you plan/design, especially if large customer.

Recommended Resources

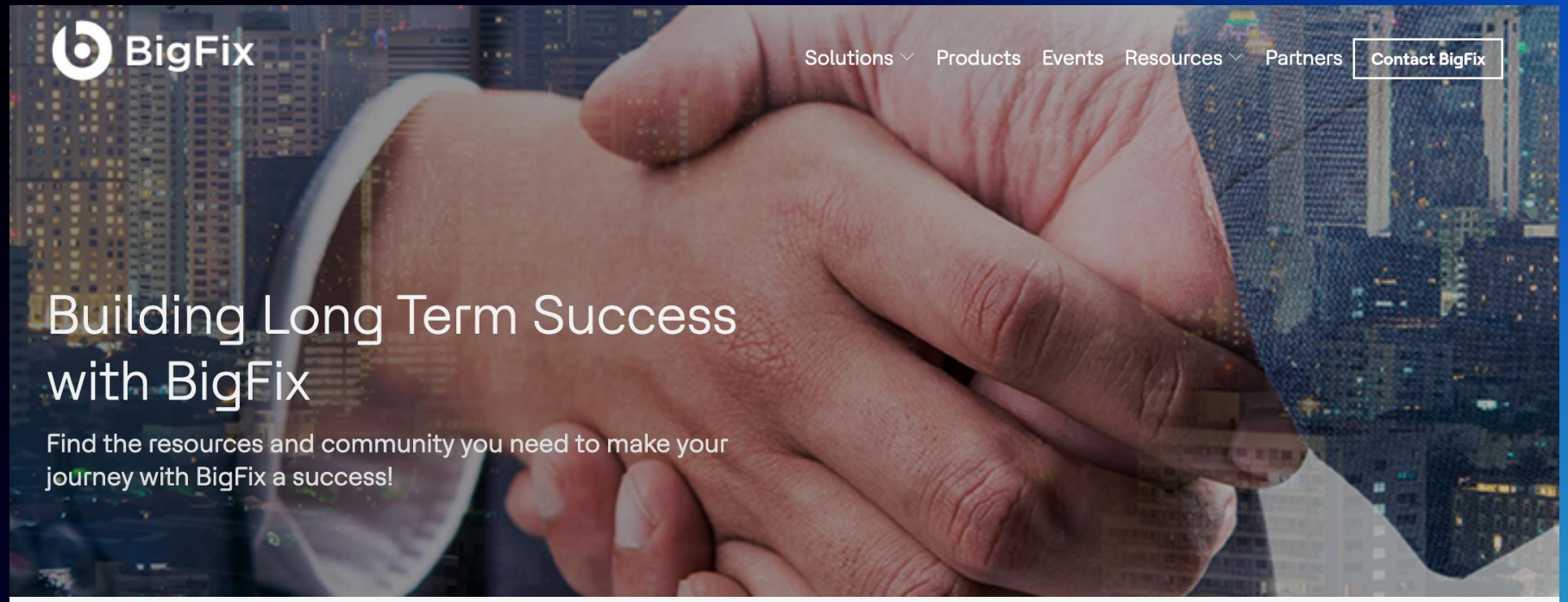
- **Bigfix Capacity Planning Guide: https://bigfix-mark.github.io/**
- **Relay affiliation basic documentation: https://bit.ly/2Wn882q**
- **Client / Server / Relay Settings: https://help.hcltechsw.com/bigfix/10.0/platform/Platform/Config/r_client_set.html**

Recommended Resources

BigFix Best Practice Series: Relay 101 – Getting Started with Relays –
https://www.buzzsprout.com/1248878/episodes/9673804

BigFix Best Practice Series: Relay 201 – Expanding the Relay Infrastructure Base –
https://www.buzzsprout.com/1248878/9868569

BigFix Best Practice Series: Relay 301 – Advanced Topics in Relay Infrastructure –
https://www.buzzsprout.com/1248878/9970613



Social Media

Engage with other BigFixers

- BigFix Forum
- BigFix Slack
- BigFix on LinkedIn
- BigFix on Twitter
- BigFix on Facebook
- BigFix on Reddit

Learning

Deepen your BigFix knowledge

- BigFix Training
- BigFix YouTube Channel
- BigFix Webinars
- BigFix Days User Conferences
- Endpoint Management Today Podcast
- BigFix Newsletter

Resources

Discover resources for BigFix – procedure documents, code, settings and more

- BigFix Documentation
- BigFix Wiki
- BigFix.me Community Content
- BigFix Developer Information Repository
- BigFix Data Sheets and White Papers
- BigFix Ideas Portal

Consultation and 1:1 Help

Engage with BigFix experts

- BigFix Support Portal
- BigFix Professional Services
- Client Advocacy

Thank You

HCLSoftware

hcltechsw.com