Pekko Akka Compatibility

Apache Pekko 1.0.3-M1 contains some experimental changes to make it possible for Pekko nodes to interact with Akka nodes (remote or cluster mode).

Apache Pekko nodes use URLs like `pekko://host:port` while Akka uses `akka://host:port`. Both Akka and Pekko will by default ignore nodes that have the wrong URL scheme. Akka will not likely be changed to accept Pekko URLs but we are changing Pekko so that it can accept Akka URLs.

The changes are in PR765 and PR1112.

Apache Pekko 1.0.3-M1 is a milestone release and is only meant for testing. Do not attempt to use this Pekko release in production. Ideally, users who want to add Pekko nodes to an Akka cluster with the aim of replacing the Akka nodes with Pekko nodes, should try this milestone release in a test cluster. We would really like to hear from users who do this to tell us how they get on.

It is strongly recommended that you read the Pekko Migration Guide. Any Akka ecosystem libs that you use, you will need Pekko equivalents or to migrate the lib yourselves. Migrating Akka code to Pekko is usually very straightforward.

2 new configuration properties are supported and need to be set if you want Akka compatibility.

- `pekko.remote.protocol-name`: This defaults to `pekko`. If you want to interact with Akka nodes, change this to `akka`
- `pekko.remote.accept-protocol-names`: This is an array and defaults to `["pekko"]`. If you want to interact with Akka nodes, change this to `["akka", "pekko"]`.

If you do not have the Akka jars on the classpath for your Pekko nodes, then you should set:

- `akka.version`: Set the version to match version of Akka you are using in your Akka nodes
- `pekko.remote.akka.version`: This property is also supported as an alias for `akka.version`

If you have custom config compatibility checks then you may need to modify them to allow Akka and Pekko configs. These checks are done when a new node tries to join a cluster (Rolling Updates).

After you have successfully started Pekko nodes and have got them to interact with your Akka nodes, you should hopefully be in a position to then stop your Akka nodes so that you are only left with Pekko nodes. You can then be able to replace the Pekko nodes with new Pekko nodes that use `pekko.remote.protocol-name="pekko"`.

Once all nodes are running with `pekko.remote.protocol-name="pekko"`, you can then start replacing all the nodes with new nodes that also set `pekko.remote.accept-protocol-names=["pekko"]`.

To reiterate, you will need to do 3 phases to fully replace your Akka cluster with a Pekko cluster:

- Join Pekko nodes (that impersonate Akka nodes) to your Akka cluster and then start to remove the Akka nodes
- When you have no Akka nodes, start replacing the Pekko nodes with ones that do not impersonate Akka nodes but that still agree to interact with Akka nodes (because some Pekko nodes are still impersonating Akka nodes - ie ones that still have `pekko.remote.protocol-name="akka"`) when all nodes are using `pekko.remote.protocol-name="pekko"`, then you can replace all the nodes with nodes that no longer accept "pekko" as a protocol name (ie `pekko.remote.accept-protocol-names=["pekko"]`).