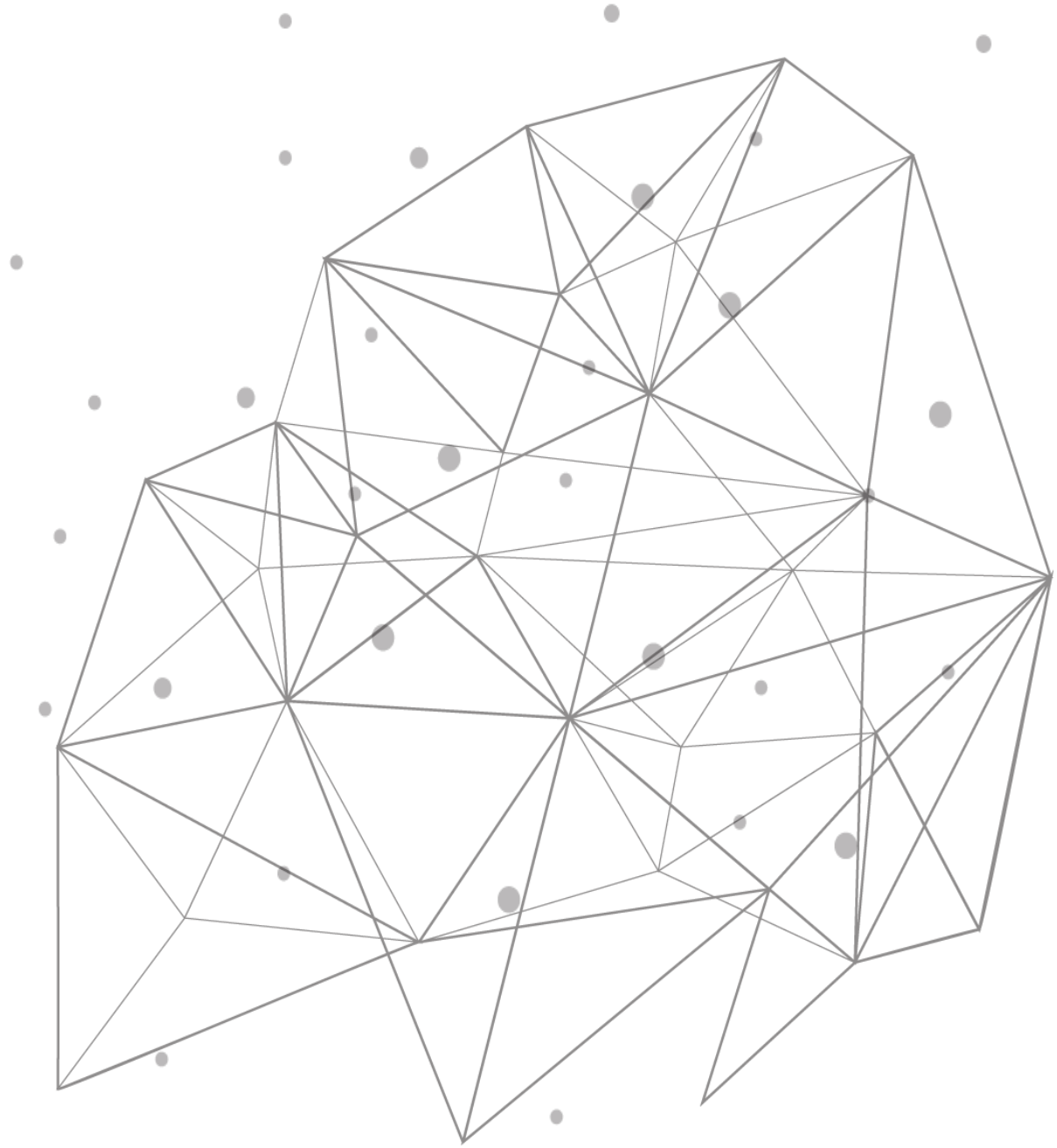
TCPWave Messaging: Fast, Reliable, and Secure with ActiveMQ



Introduction

In simple terms, a message broker is something that would deliver an application's message to another application, which needs it for further processing. But it isn't effortless as what the short description says. Not every message broker would fit well for the use-cases of different classes of applications within an enterprise infrastructure. Some of them would do the job but could have performance issues, operational limitations, etc. Hence, a message broker is the core of a message-oriented infrastructure where data has to flow seamlessly between different moving parts. Thus, finding the right message broker is critical to efficiently solve messaging issues in a variety of enterprise applications.

ActiveMQ is a top-rated messaging-oriented middleware service that is built on top of Java & offers standards-based application integration across various languages, including C#, Java, Python, Ruby, etc. ActiveMQ is probably the most flexible enterprise messaging system currently available in the market, with a history of over 17 years as of 2021. Over the years since its initial release, it has evolved to be a standards-compliant, scalable product that is efficient of handling a wide array of enterprise messaging use-cases. ActiveMQ is widely used by enterprises directly & behind the scenes in their SaaS offerings, commercial cloud offerings, etc.

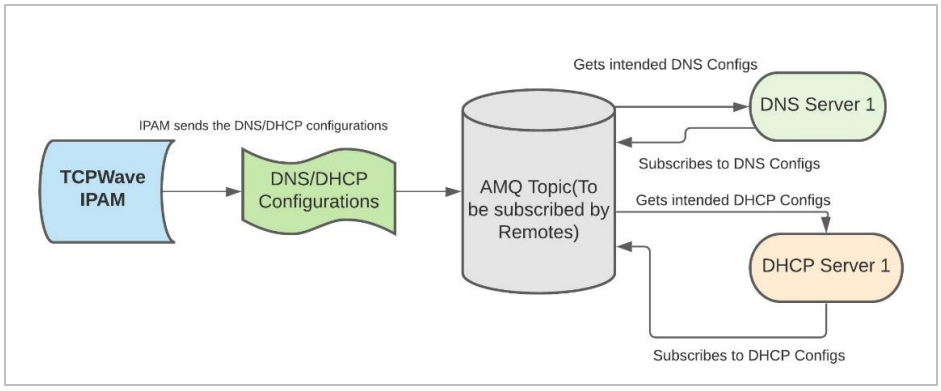
Why ActiveMQ for DDI?

Administrators rely on IPAM solutions to centrally manage their IP space from a single pane of glass. An administrator gets the flexibility to push configurations to a DNS or DHCP server directly from the IPAM's WEB-UI without manually touching each & every server. In this regard, effective communication between the IPAM manager & the connected DNS or DHCP servers becomes business-critical. Different vendors depend on different messaging systems & some of them even depend on obsolete technologies. TCPWave Engineering team has done extensive research & analysis before choosing ActiveMQ over other messaging systems currently available in the industry. ActiveMQ turned out to be the most efficient messaging middleware out of all other popular technologies that were considered. Key benefits of ActiveMQ are:

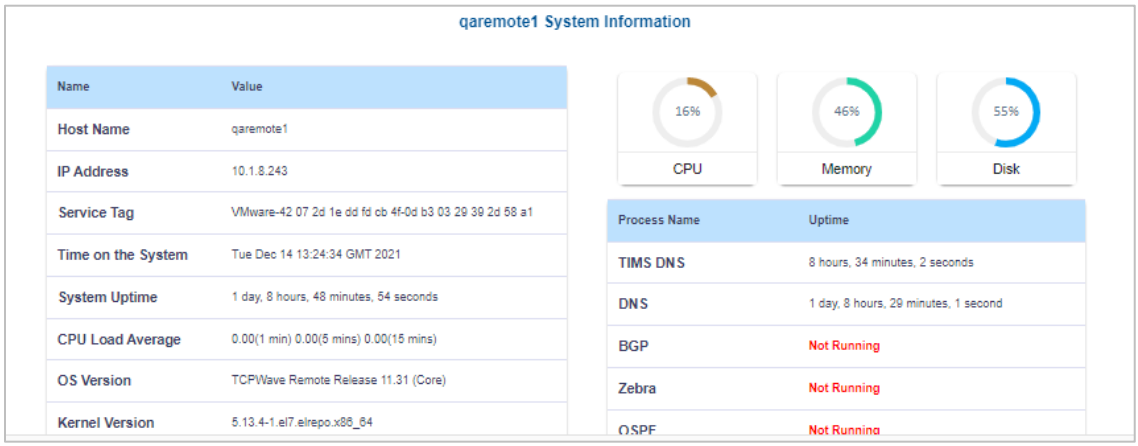
1. ActiveMQ can provide robust, reliable & one-time message delivery guarantees.
2. It supports many messaging protocols & has matured libraries to allow connectivity from different operating systems & programming languages. It seamlessly supports integration between an application written in Java & another one written in Python(Just an example) – which typically could be a challenge with most other messaging systems available.
3. Highly scalable.
4. Since broker clustering is supported, failover to another TCPWave IPAM in the event of failure becomes easy.
5. Supports multiple protocols like HTTP/HTTPS, STOMP, UDP, etc., for connectivity.
6. Provides reliable communication between IPAM & DNS or DHCP Servers.

How Does It Work?

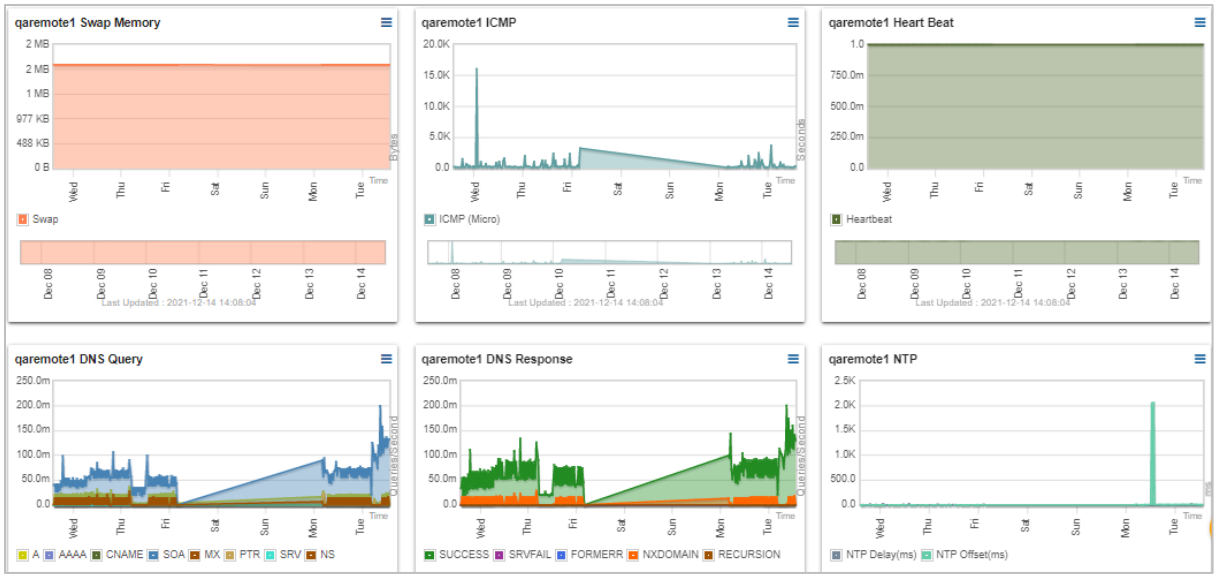
Here's a high-level overture of how the technology works behind pushing the DNS/DHCP Configurations from the IPAM to TCPWave DNS/DHCP Servers:



TCPWave uses a Secure Message Tunnel (SMT) for all communication between an IPAM manager & a DNS or DHCP server. TCPWave Secure Message Tunnel is robust, secure, fast & is a complete enterprise-grade messaging system built on top of ActiveMQ. As illustrated in the diagram above, TCPWave utilizes the subscribe messaging style for IPAM to Remotes communication. The TCPWave transport layer is secured with TCP over SSL & uses certificate-based authentication. When a DNS/DHCP server is provisioned, the TCPWave IPAM would add this Remote to its list of allowed AMQ client list & SMT communication from IPAM is restricted to that list of provisioned Remotes. This ensures that a non-intended intruder doesn't get a chance to talk to the IPAM over the TCPWave SMT. Once these Remote DNS or DHCP servers are connected to the IPAM, the IPAM could give complete visibility over its resource utilization, service usage trends, etc. as shown below:



All of these data are pulled via TCPWave Secure Message Tunnel & the stats seen above is just a high-level summary of a DNS Server's performance metrics. An administrator would get to drill down to the specifics further like this below:



Unlike many DDI vendors who still rely on single-threaded channels, there are over ten different types of transport layer channels in a TCPWave IPAM to Remote communication. This would mean that when an administrator creates new configurations that is expected to be pushed out to a number of DNS/DHCP servers, TCPWave IPAM's messaging channel takes just about a second to complete all those transactions. Since the SMT uses subscribe model of ActiveMQ, there's no need of continuous polling when a Remote is down for any reason – IPAM would sync the configuration for that specific Remote whenever it's back online. TCPWave SMT supports a failover mechanism, enabling it to connect to the second IPAM within the list if the primary IPAM goes down.

Conclusion

TCPWave T-Message Secure channel for IPAM to Remotes communication is one of the best-in-class transport channels built on top of the most efficient messaging system available in the industry. ActiveMQ is an open-source implementation of Java Message Service at its core. Java Message Service API is a messaging standard that allows application components based on the Java Platform Enterprise Edition (Java EE) to create, send, receive, and read messages.

TCPWave Secure Message Tunnel's efficiency stands out when compared to any other competitor's transport channel in terms of speed, flexibility, security, message encryption & error handling during message transmission. For a quick demo, contact the **TCPWave Sales Team**.