2.3.4.2.2 Data-In-Motion

Return to State of Data Taxonomy

Overview

Return to Top

Data-in-Motion, also referred to as **Data in Transit** or **Data in Flight**, is a Digital Asset transmitted between locations (i.e., between computers or computer components). Data-In-Motion also describes data within Random Access Memory (RAM).

Some examples of Data-in-Motion include data:

- Moving from one Endpoint to another Endpoint (e.g., endpoint device to a Web Application (Web App)
- Moving between Virtual Machines
- Moving between Cloud Services
- Traversing trusted private networks (i.e. Intranet) or untrusted networks (i.e., Internet).

Once the data arrives at its final destination, it is classified as Data-at-Rest.

Because Data-in-Motion is vulnerable to Man-in-the-Middle (MiTM) Attack attacks, it is often Encrypted to prevent interception. For example, the Internet Small Computer System Interface (iSCSI) transport layer incorporates Internet Protocol Security (IPsec), which can encrypt data as it is transferred between two devices to prevent a hacker with a sniffer from seeing the contents of that data. IPSec has been used extensively as a transit encryption protocol for Virtual Private Network (VPN) tunnels because it makes use of Cryptography Algorithms such as Triple DES (3DES) and Advanced Encryption Standard (AES). Encryption Platform software can also be integrated with existing Enterprise Resource Planning (ERP) systems to keep Data-in-Motion secure.

DIDO Specifics

Return to Top

Overview

Within DIDOs, the **Data-In-Motion** refers to data as it moves between the **Nodes** within a **Node Network**. The data moves on an infrastructure that includes Hypertext Transfer Protocol (HTTP), Hypertext Transport Protocol Secure (HTTPS), Gossip Protocol, etc. There are any number of Technical and de facto Standards that are relevant to the Transport of the data between the nodes.

Although the **Data-In-Motion** is dependent-upon and influenced-by the Transport Layer, this section is concerned about the data flowing on the **Transport Layer** rather than on the Physical Layer and the Protocols.

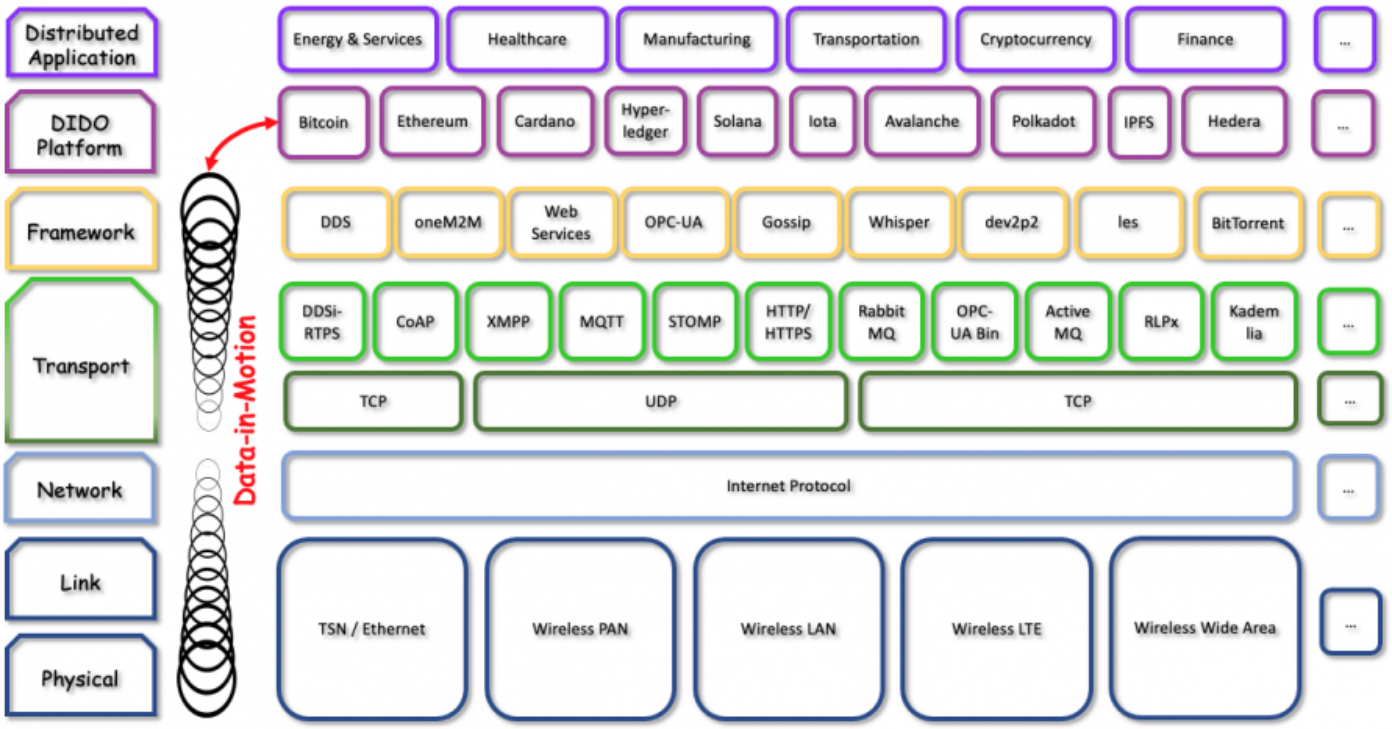

Figure 1: The layers in a DIDO and where the Data-In-Motion occurs.

> **Note:** Also see: 4.3.5.3 System Manageability Issues figure on the Industrial Internet Consortium's Connectivity Framework

IIOT Level[1)]	Description
Distributed Application	Distributed Application (ĐApp or DApp) are Applications that exist and run on a distributed Peer-to-Peer (P2P) network of Nodes. DApps are outside the control of a single Governing Body. DApps are created for a variety of purposes, including: • Energy • Services • Healthcare • Manufacturing • Supply Chain • Transportation • Cryptocurrency • Finance • Gaming • Social media
DIDO Platform	DIDO Platforms are not a single software entity such as in an Application, but rather a set of components integrated together. In a DIDO, the integration includes Software and Hardware (H/W) Components that are more than just the Software (SW) and Hardware (H/W) components integrated as aSoftware Stack or Solution Stack residing on a single Node but on a collection of Nodes called a Node Network. • Bitcoin • Cardano • Ethereum • Hyperledger • Solana • Iota • Avalanche • Polkadot • InterPlanetary File System (IPFS) • Hedera

IIOT Level[1)]	Description
Framework	A Framework is Software (SW) that is an Non-Functional Requirements abstraction of Reusable, generic, tailorable Components by the use of user-written, application-specific Plug In software. Frameworks provide a standardized mechanism for designing, building and deploying Applications. Framweworks leverage either a Software Stack or Solution Stack. The lowest level of the Framework covers the highest layer of the Open Systems Interconnection (OSI) Model, the applayer and includes: • Data Distribution Service (DDS) • OneM2M • Web Service • Open Platform Communication Unified Architecture (OPC-UA) • Gossip Protocol • Whisper • devp2p • Light Ethereum Subprotocol (LES) • BitTorrent

IIOT Level[1)]	Description
Transport	There are two levels to the **Transport Level**. The transport mechanisims for messages and for connectivity: • The **Message Transport Level** which is analogous to the the Open Systems Interconnection (OSI) Model Presentation Layer and includes: ◦ DDSI-RTPS ◦ Constrained Application Protocol (CoAP) ◦ eXtensible Messaging and Presence Protocol (XMPP) ◦ Message Queuing Telemetry Transport (MQTT) ◦ Simple (or Streaming) Text Oriented Message Protocol (STOMP) ◦ Hypertext Transfer Protocol (HTTP) ◦ Hypertext Transport Protocol Secure (HTTPS) ◦ RabbitMQ ◦ Open Platform Communication Unified Architecture (OPC-UA) ◦ Apache ActiveMQ ◦ RLPx ◦ Kademlia • The **Connectivity** which is analogous to the Open Systems Interconnection (OSI) Model netlayer and includes: ◦ Transmission Control Protocol (TCP) ◦ User Datagram Protocol (UDP) ◦ Port Numbers
Network	The **Network Level** is analogous to the Open Systems Interconnection (OSI) Model Session Layer providing data routing paths for network communication. Data is transferred in the form of packets via logical network paths in an ordered format controlled by the network layer. Logical connection setup, data forwarding, routing and delivery error reporting are the Network Layer's primary responsibilities. This layer includes: • Internet Protocol (IP) • Router

IIOT Level[1)]	Description
Link	The **Link Level** is analgous to the Open Systems Interconnection (OSI) Model Data Link Layer (DLL) controling connections between multiple computers. The session layer tracks the dialogs between computers, which are also called sessions. This layer establishes, controls and ends the sessions between local and remote applications.

Physical	The **Physical Level** is more or less a combination of the Open Systems Interconnection (OSI) Model Physical Layer and Transport Layer.
	• Time Sensitive Network (TSN)
	• Wireless Network - Pulse-Amplitude Modulation (PAM)
	• Wireless Network - Local Area Network (LAN)
	• Wireless Network - Long-Term Evolution (LTE)
	• Wide Area Network (WAN)
	• Network Cabling
	• Registered Jack-45 (RJ45) Connectors
	• Wifi Frequencies
	• Network Devices other than Routers :
	  ◦ Hub
	  ◦ Switch
	  ◦ Bridge
	  ◦ Gateway
	  ◦ Modem
	  ◦ Repeater
	  ◦ Network Appliance

Discussion

Most of the **DIDO Platforms** in combination with the **Frameworks** and **Application Layer** provide a Reusable, general mechanism for Dapps to interact with DIDO **Nodes** in a **Node Network**. However, this combination of **Platforms**, **Frameworks**, and **Applications** create **DIDO Silos** offering little to no Interoperability between the silos.

interface with a specific DIDO Platform, however, the **Frameworks** offer little in the way of Bridging between the **DIDO Platforms** making it hard for **Data-In-Motion** to cross from DIDO Platform silo to another DIDO Platform silo, or in other words to Innteroperate. The Interoperability issues between the

various platforms silos originate at at the lowest level of the **Framework** stack, or depending on the perspective, the upper level of the Open Systems Interconnection (OSI) Model Presentation Layer.

1)

The Industrial Internet Consortium **Levels** are similar, but differnt from the Open Systems Interconnection (OSI) Model **Layers**. However, there is some overlap between the **Levels** and **Layers** especially at the lower rungs. However, it is imprtant to differentiate between the two. Where possible in the discussion of the **Levels**, the corresponding **Layers** are provided.