MICROSOFT EXHIBIT 1047
Microsoft v. Qomplx
IPR2026-00183

EXHIBIT I
U.S. Patent No. 11,539,663

As used herein and with respect to the ’663 Patent, the term “Accused Edge Pipeline Products” means:

(a) Microsoft products that incorporate, rely upon, interact with, or otherwise utilize Azure Monitor pipeline at edge (“Edge Pipeline”), including at least Azure Monitor;

(b) Any other systems, services, or products that utilize the libraries, applications, scripts, packages, or other modules that implement the functionality described below in a manner not materially different with respect to the claims charted below

(c) any other products that infringe the asserted claims for analogous reasons to those described below; and,

(d) Microsoft products that practice one of more claims of the ’663 Patent.

This claim chart for the ’663 Patent covers all Accused Edge Pipeline Products. The theory of infringement described below in connection with the Asserted Claims is analogous to the theory of infringement for all the Accused Edge Pipeline Products.

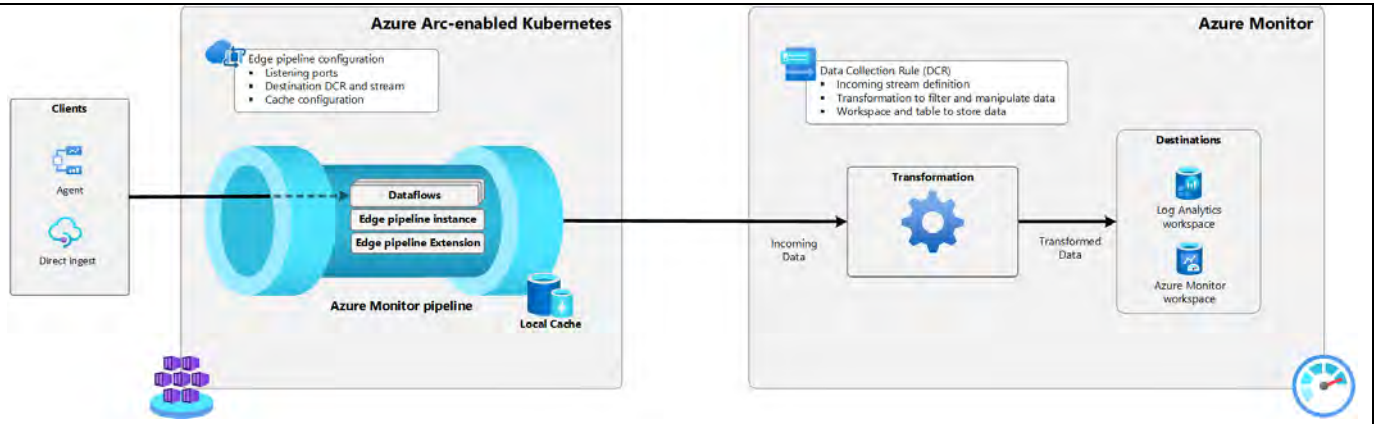
I. Claim 1

A system for ingestion of data into a cloud-based service from an external network, comprising:	The Accused Edge Pipeline Products are systems for ingestion of data into a cloud-based service from an external network. For example, Microsoft's documentation explains that the Azure Monitor Pipeline "extends the data collection capabilities of Azure Monitor to edge . . . environments. It enables at-scale collection, and routing of telemetry data before it's sent to the cloud."[1]
a midserver comprising at least a processor, a memory, and a plurality of programming instructions stored in the memory and operating on the processor, wherein the plurality of programming instructions, when operating on the processor, cause the processor to:	The Accused Edge Pipeline Products comprise a midserver comprising at least a processor, a memory, and a plurality of programming instructions stored in the memory and operating on the processor. For example, Microsoft's documentation explains that "Azure Monitor pipeline is a containerized solution that is deployed on an Arc-enabled Kubernetes cluster."[2] The following exemplary diagram, reproduced from Microsoft's literature, "shows the components of the pipeline":[3]

[1] Microsoft, *Configure Azure Monitor pipeline*, available at `https://learn.microsoft.com/en-us/azure/azure-monitor/essentials/edge-pipeline-configure?tabs=Portal` [hereinafter *Configure Azure Monitor Pipeline*].

[2] *Configure Azure Monitor Pipeline.*

[3] *Id.*

	 Microsoft's documentation further explains that "[t]he pipeline can cache data locally and sync with the cloud when connectivity is restored and route telemetry to Azure Monitor in cases where the network is segmented and data can't be sent directly to the cloud."[4] As another example, Microsoft's documentation explains that Azure Monitor pipeline "extends the data collection process to your own data center. It enables at-scale collection and routing of telemetry data before it's delivered to the cloud."[5]

[4] *Id.*

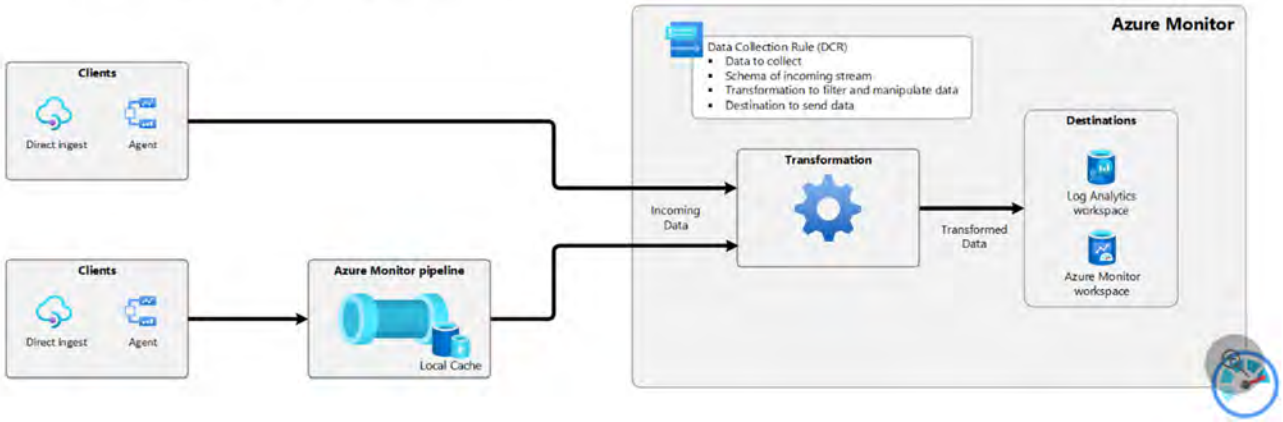
[5] Microsoft, *Data collection rules (DCRs) in Azure Monitor*, available at `https://learn.microsoft.com/en-us/azure/azure-monitor/essentials/data-collection-rule-overview` [hereinafter *DCRs in Azure Monitor*].

Azure Monitor pipeline

The Azure Monitor pipeline extends the data collection process to your own data center. It enables at-scale collection and routing of telemetry data before it's delivered to the cloud.

Specific use cases for Azure Monitor pipeline are:

- **Scalability**. The pipeline can handle large volumes of data from monitored resources that may be limited by other collection methods such as Azure Monitor agent.
- **Periodic connectivity**. Some environments may have unreliable connectivity to the cloud, or may have long unexpected periods without connection. The pipeline can cache data locally and sync with the cloud when connectivity is restored.
- **Layered network**. In some environments, the network is segmented and data can't be sent directly to the cloud. The pipeline can be used to collect data from monitored resources without cloud access and manage the connection to Azure Monitor in the cloud.



	As another example, Microsoft's documentation explains that Azure Arc-enabled Kubernetes works with "clusters running on your own on-premises data center."[6] Microsoft's documentation further explains that an "AKS cluster" comprises "underlying virtual machines (VMs) that run your applications."[7] Based on Qomplx's analysis to date, this element appears to be practiced literally, including as charted above. Any differences between the charted material and this claim element are expected to be insubstantial, in which case the element is expected to be practiced under the Doctrine of Equivalents. For example, the technology described above performs substantially the same function in substantially the same way to achieve substantially the same result as this claim element.
automatically install a virtual appliance software application,	The Accused Edge Pipeline Products automatically install a virtual appliance software application. For example, Microsoft's documentation explains that "[w]hen you use the Azure portal to enable and configure the pipeline, all required components are created," including on the Azure Arc cluster.[8] Microsoft documentation shows that services are created automatically:[9]

[6] Microsoft, *What is Azure Arc-enabled Kubernetes?*, available at `https://learn.microsoft.com/en-us/azure/azure-arc/kubernetes/overview` [hereinafter *What is Azure Arc-enabled Kubernetes?*].

[7] Microsoft, *Core concepts for Azure Kubernetes Service (AKS)*, available at `https://learn.microsoft.com/en-us/azure/aks/core-aks-concepts`

[8] *Configure Azure Monitor Pipeline.*

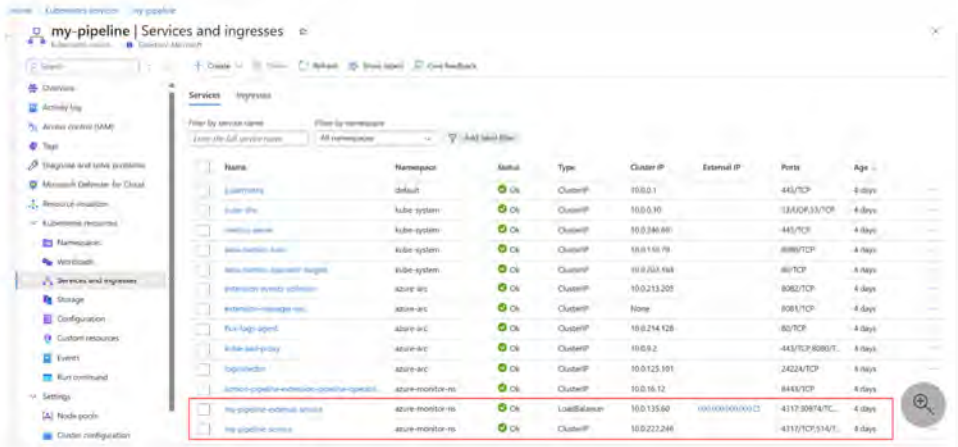
[9] *Id.*

Verify configuration

Verify pipeline components running in the cluster

In the Azure portal, navigate to the **Kubernetes services** menu and select your Arc-enabled Kubernetes cluster. Select **Services and ingresses** and ensure that you see the following services:

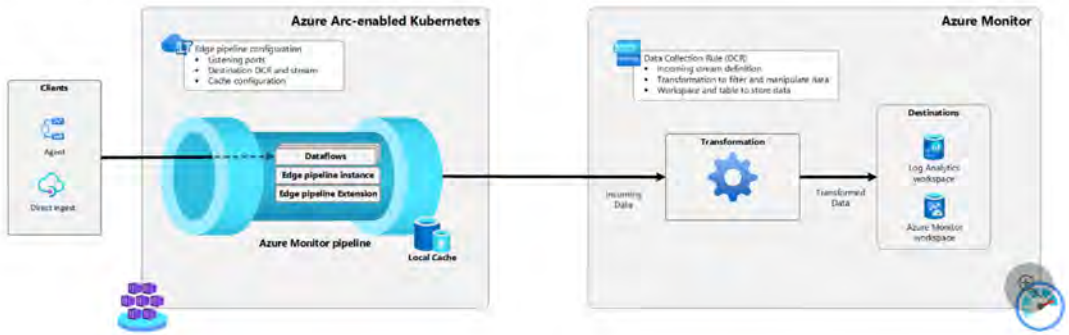
- <pipeline name>-external-service
- <pipeline name>-service



Click on the entry for **<pipeline name>-external-service** and note the IP address and port in the **Endpoints** column. This is the external IP address and port that your clients will send data to. See Retrieve ingress endpoint for retrieving this address from the client.

Kubernetes documentation explains that services are collection of "one or more pods in your cluster," each containing one or more containers.[10]

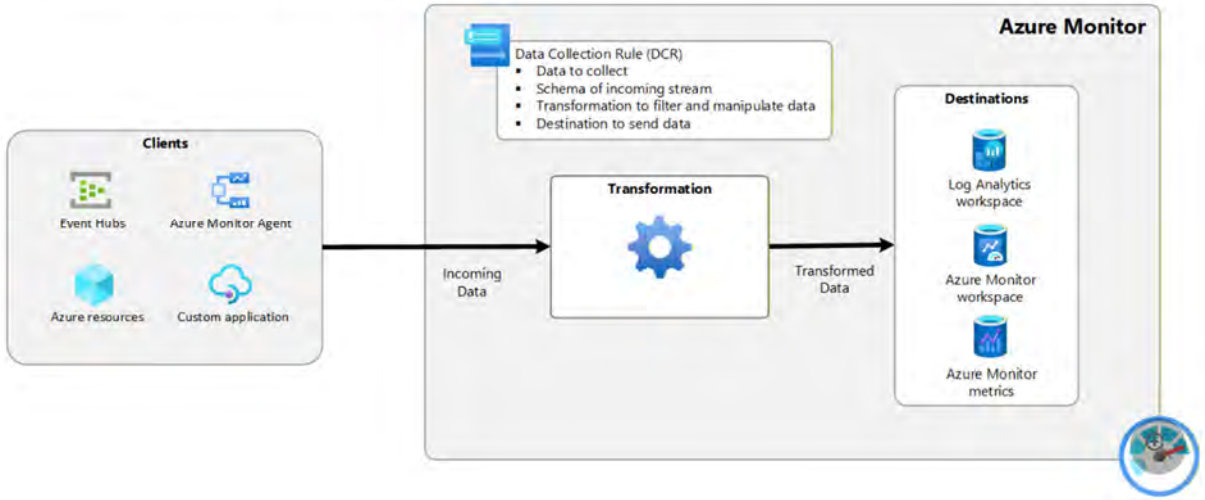
[10] Kubernetes.io, *Service*, available at `https://kubernetes.io/docs/concepts/services-networking/service/`.

the virtual appliance software application configured to automatically load a plurality of stored configurations on the midserver;	The virtual appliance software application is configured to automatically load a plurality of stored configurations on the midserver. For example, Microsoft's documentation explains that in the Azure Monitor Pipeline, the "pipeline configuration file defines the data flows and cache properties for the pipeline. The DCR defines the schema of the data being sent to the cloud, a transformation to filter or modify the data, and the destination where the data should be sent. Each data flow definition for the pipeline configuration specifies the DCR and stream within that DCR that will process that data in the cloud."[11] The pipeline configuration file defines the data flows and cache properties for the pipeline. The DCR defines the schema of the data being sent to the cloud, a transformation to filter or modify the data, and the destination where the data should be sent. Each data flow definition for the pipeline configuration specifies the DCR and stream within that DCR that will process that data in the cloud.  As another example, Azure Monitor distributes data collection rules to entities within the system, including to pipelines, to configure the flow of data.[12] Microsoft's documentation explains that "Azure Monitor creates and configures the DCR for you. . . . Data collection rules (DCRs) are stored in Azure so they can be centrally deployed and managed like any other Azure resource."[13] Microsoft's

[11] *Configure Azure Monitor Pipeline.*

[12] *DCRs in Azure Monitor*.

[13] *Id*.

	documentation goes on to explain that the "data collection process supported by DCRs provides a common processing path for incoming data. Each data collection scenario is defined in a DCR. The DCR provides instructions for how Azure Monitor should process the data it receives."[14] **Data collection process** The data collection process supported by DCRs provides a common processing path for incoming data. Each data collection scenario is defined in a DCR. The DCR provides instructions for how Azure Monitor should process the data it receives. Depending on the scenario, DCRs specify all or some of the following: • Data to collect and send to Azure Monitor. • Schema of the incoming data. • Transformations to apply to the data before it's stored. • Destination where the data should be sent. 

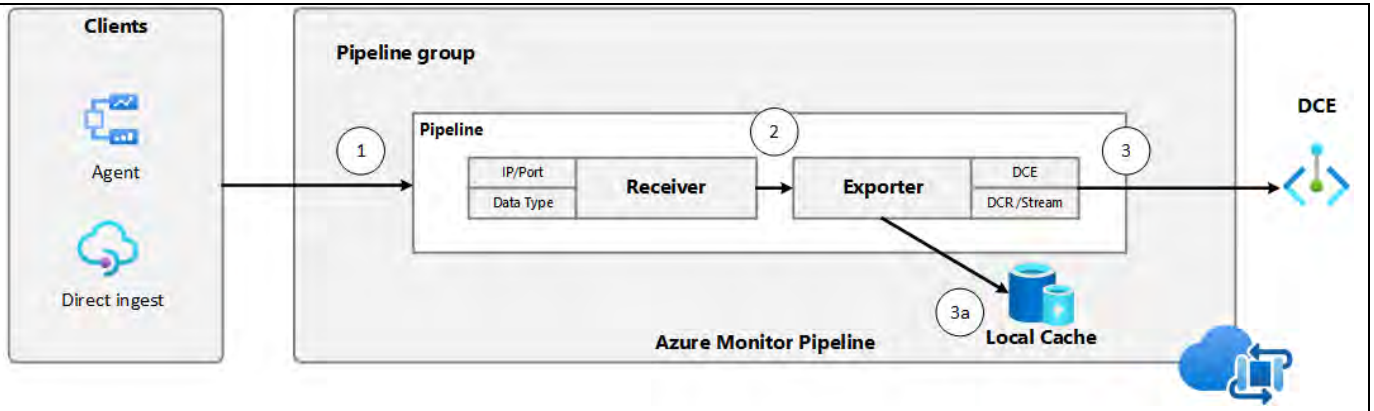
[14] *DCRs in Azure Monitor*.

	Microsoft's documentation explains that the "components and configurations" that "are required to enable the Azure Monitor pipeline" are "created for [users]" that use the Azure portal.[15] Based on Qomplx's analysis to date, this element appears to be practiced literally, including as charted above. Any differences between the charted material and this claim element are expected to be insubstantial, in which case the element is expected to be practiced under the Doctrine of Equivalents. For example, the technology described above performs substantially the same function in substantially the same way to achieve substantially the same result as this claim element.
establish a secure network connection to an external network; receive data over a local network from a plurality of computing devices;	The Accused Edge Pipeline Products establish a secure network connection to an external network and receive data over a local network from a plurality of computing devices. For example, Microsoft documentation explains that in Azure Monitor pipeline, "[o]ne or more data flows listen for incoming data from clients, and the pipeline extension forwards the data to the cloud."[16] Azure Monitor pipeline sends data to a DCE, which is the "[e]ndpoint where the data is sent to Azure Monitor in the cloud. The pipeline configuration includes a property for the URL of the DCE so the pipeline instance knows where to send the data."[17]

[15] *Configure Azure Monitor Pipeline.*

[16] *Id.*

[17] *Id.*

	 Microsoft's documentation explains that all site-to-cloud communication is performed over a secure network connection.[18]

[18] Microsoft, *Double encryption*, available at `https://learn.microsoft.com/en-us/azure/security/fundamentals/double-encryption#data-in-transit`.

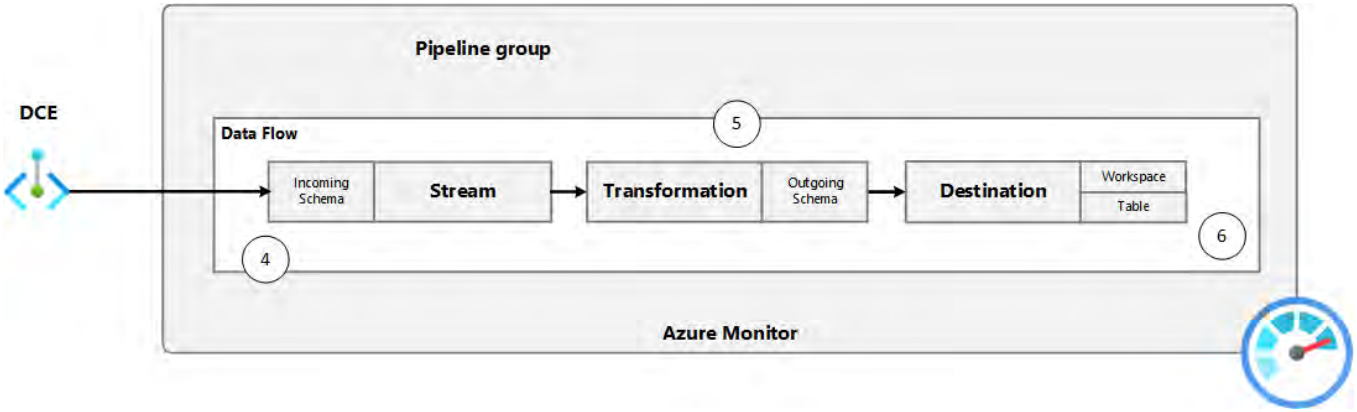
	Data in transit Microsoft's approach to enabling two layers of encryption for data in transit is: • **Transit encryption using Transport Layer Security (TLS) 1.2 to protect data when it's traveling between the cloud services and you.** All traffic leaving a datacenter is encrypted in transit, even if the traffic destination is another domain controller in the same region. TLS 1.2 is the default security protocol used. TLS provides strong authentication, message privacy, and integrity (enabling detection of message tampering, interception, and forgery), interoperability, algorithm flexibility, and ease of deployment and use. • **Additional layer of encryption provided at the infrastructure layer.** Whenever Azure customer traffic moves between datacenters-- outside physical boundaries not controlled by Microsoft or on behalf of Microsoft-- a data-link layer encryption method using the IEEE 802.1AE MAC Security Standards (also known as MACsec) is applied from point-to-point across the underlying network hardware. The packets are encrypted and decrypted on the devices before being sent, preventing physical "man-in-the-middle" or snooping/wiretapping attacks. Because this technology is integrated on the network hardware itself, it provides line rate encryption on the network hardware with no measurable link latency increase. This MACsec encryption is on by default for all Azure traffic traveling within a region or between regions, and no action is required on customers' part to enable. Based on Qomplx's analysis to date, this element appears to be practiced literally, including as charted above. Any differences between the charted material and this claim element are expected to be insubstantial, in which case the element is expected to be practiced under the Doctrine of Equivalents. For example, the technology described above performs substantially the same function in substantially the same way to achieve substantially the same result as this claim element.
apply a plurality of transformations to at least a portion of the received data; and	The Accused Edge Pipeline Products apply a plurality of transformations to at least a portion of the received data. For example, the data collection rules can "a transformation to filter or modify the data."[19]

[19] *Configure Azure Monitor Pipeline.*

Configuration	Description
Data collection rule (DCR)	Configuration file that defines how the data is received by Azure Monitor and where it's sent. The DCR can also include a transformation to filter or modify the data before it's sent to the destination.

Microsoft's documentation goes on to explain that the transformations "may filter data, remove or add columns, or completely change its schema."[20]

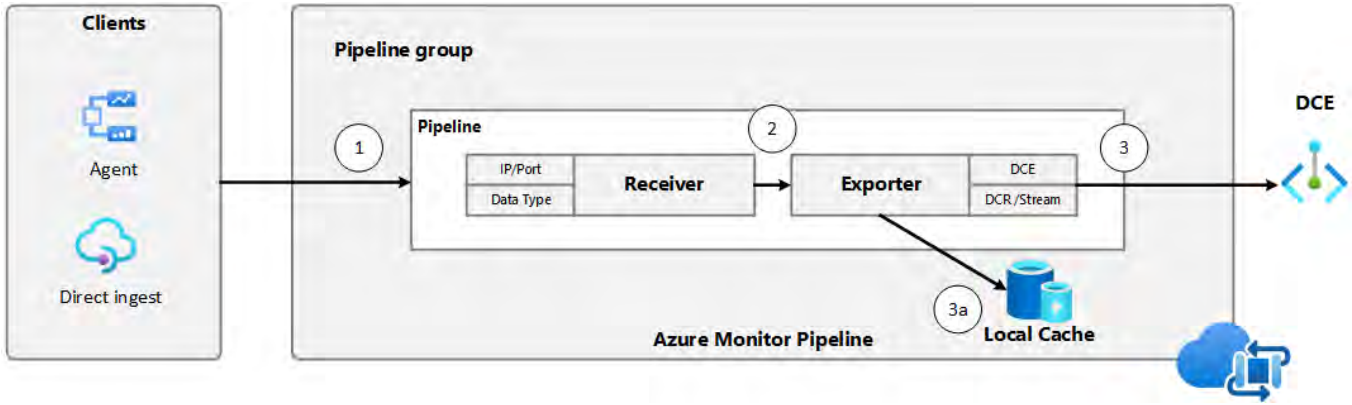
5.	Azure Monitor applies a transformation to the data.	The DCR includes a transformation that filters or modifies the data before it's sent to the destination. The transformation may filter data, remove or add columns, or completely change its schema. The output of the transformation must match the schema of the destination table.



As another example, Azure Monitor pipeline can store data in a local cache if it cannot connect to the DCE.[21]

[20] *Id.*

[21] *Id.*

	3a. Exporter stores data in the local cache if it can't connect to the DCE. — Persistent volume for the cache and configuration of the local cache is enabled in the pipeline configuration.  Based on Qomplx's analysis to date, this element appears to be practiced literally, including as charted above. Any differences between the charted material and this claim element are expected to be insubstantial, in which case the element is expected to be practiced under the Doctrine of Equivalents. For example, the technology described above performs substantially the same function in substantially the same way to achieve substantially the same result as this claim element.
retransmit the received data over the secure connection as a single data stream.	The Accused Edge Pipeline Products retransmit the received data over the secure connection as a single data stream. For example, as explained above, the Microsoft's documentation explains that "the pipeline extension forwards the data to the cloud."[22] Azure Monitor pipeline sends data to a DCE, which is the "[e]ndpoint

[22] *Id.*

	where the data is sent to Azure Monitor in the cloud. The pipeline configuration includes a property for the URL of the DCE so the pipeline instance knows where to send the data."[23] Based on Qomplx's analysis to date, this element appears to be practiced literally, including as charted above. Any differences between the charted material and this claim element are expected to be insubstantial, in which case the element is expected to be practiced under the Doctrine of Equivalents. For example, the technology described above performs substantially the same function in substantially the same way to achieve substantially the same result as this claim element.

[23] *Id.*