MICROSOFT EXHIBIT 1090
Microsoft v. Qomplx
IPR2026-00182

EXHIBIT F
U.S. Patent No. 12,301,627

As used herein and with respect to the ’627 Patent, the term “Accused ’627 Fusion Products” means:

(a) Microsoft products that incorporate, rely upon, interact with, or otherwise utilize Microsoft Fusion (“Fusion”), including at least Microsoft Sentinel (“Sentinel”) and Microsoft Defender;

(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 ’627 Patent.

This claim chart for the ’627 Patent covers all Accused ’627 Fusion Products. The theory of infringement described below in connection with the Asserted Claims is analogous to the theory of infringement for all the Accused ’627 Fusion Products.

I. Claim 1

A computer system comprising: a hardware memory, wherein the computer system is configured to execute software instructions stored on nontransitory machine-readable storage media comprising software instructions that:	The Accused ’627 Fusion Products include a computer system comprising a hardware memory, wherein the computer system is configured to execute software instructions stored on nontransitory machine-readable storage media comprising software instructions that perform as discussed below. For example, Microsoft Sentinel “is a cloud-native Security Information and Event Management (SIEM) and unified security platform . . . . built on a modern data lake.”[1] The Sentinel data lake “is a fully managed, cloud-native data lake purpose-built for security operations. It unifies, retains, and analyzes security data at scale - providing the foundation for advanced analytics, AI-driven insights, and agentic defense.”[2] Microsoft’s documentation explains that Sentinel data lake is priced based on the amount of data processed, the amount of data analyzed, compute hours, and the amount of data stored.[3]

[1] Microsoft, *What is Microsoft Sentinel?*, available at `https://learn.microsoft.com/en-us/azure/sentinel/sentinel-overview` [hereinafter *What is Microsoft Sentinel?*].

[2] *Id*.

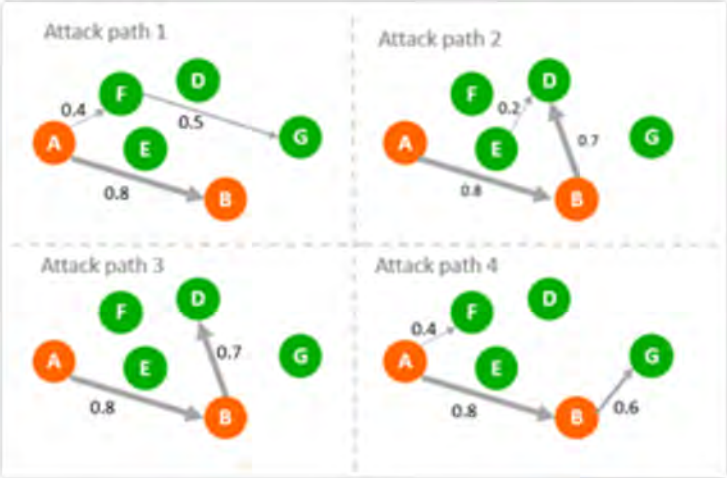
[3] Microsoft, *Microsoft Sentinel pricing*, available at `https://www.microsoft.com/en-us/security/pricing/microsoft-sentinel` [hereinafter *Sentinel Pricing*].

	Sentinel Microsoft Sentinel data lake enables security teams to ingest, retain, and analyze massive volumes of security data cost-effectively. With separate compute and storage meters, the data lake allows defenders to flexibly run advanced data insights, machine learning, and forensic investigations from a single point.[4] Learn more. SKU / Meter type / Price Data lake ingestion / Data Processed (GB) / $0.05 USD Data processing / Data Processed (GB) / $0.1 USD Data lake query / Data Analyzed (GB) / $0.005 USD Advanced Data Insights / 1 Compute Hour / $0.15 USD Data lake storage / Data Stored (GB/Month) / $0.026 USD
store in the hardware memory a representation of a first graph, wherein the representation of the first graph comprises representations of a first plurality of nodes corresponding to a first plurality of entities and further comprises representations of a first plurality of edges,	The software instructions of the Accused '627 Fusion Products store in the hardware memory a representation of a first graph, wherein the representation of the first graph comprises representations of a first plurality of nodes corresponding to a first plurality of entities and further comprises representations of a first plurality of edges. For example, Microsoft's documentation explains that Fusion "builds and continually updates a hyperconnected graph on large scale data sets" in which "nodes represent the entities and the activities, and the edges represent the relationships between the nodes. . . . The entities can be IP addresses, accounts, Cloud resources, virtual machines, etc."[4]

[4] Microsoft, *Behind the Scenes: The ML Approach for Detecting Advanced Multistage Attacks with Sentinel Fusion*, available at `https://techcommunity.microsoft.com/blog/microsoftsentinelblog/behind-the-scenes-the-ml-approach-for-detecting-advanced-multistage-attacks-with/3239236` [hereinafter *Fusion Behind the Scenes*].

	Graph forming: Fusion builds and continually updates a hyperconnected graph on large scale data sets, typically millions of anomalous signals in a customer workspace. In the graph, the nodes represent the entities and the activities, and the edges represent the relationships between the nodes. The activities are the alerts and anomalies from different sources. The entities can be IP addresses, accounts, Cloud resources, virtual machines, etc. *Figure 1: Graph formed from a Microsoft Sentinel workspace*
wherein the first graph is a directed graph,	The first graph of the Accused '627 Fusion Products is a directed graph. For example, Microsoft's documentation shows directionality of edges in Fusion's graph visualizations:[5]

[5] *Id.*

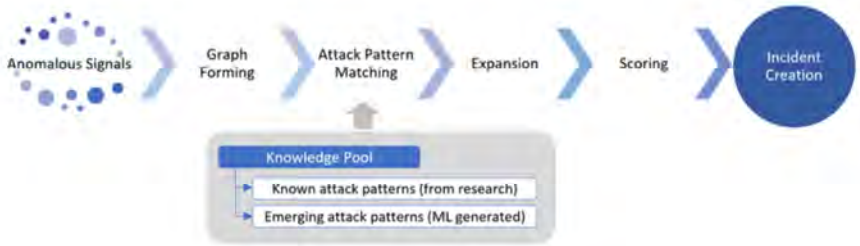
	 *Figure 4: Expansion - probabilistic random walk* *Figure 5: Expansion - aggregate weights and apply threshold* Based on Qomplx's analysis to date, this element appears to be practiced literally, including as charted above. Any differences between the charted material and this claim element are expected 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.

wherein the first plurality of entities comprises a plurality of accounts and a plurality of resources, and wherein each edge of the first plurality of edges corresponds to a respective relationship between a respective pair of entities;	The first plurality of entities of the Accused '627 Fusion Products comprises a plurality of accounts and a plurality of resources, and each edge of the first plurality of edges of the Accused Fusion Products corresponds to a respective relationship between a respective pair of entities of the first plurality of entities. For example, as explained above, nodes of Fusion's graph representation "represent the entities and the activities, and the edges represent the relationships between the nodes. . . . The entities can be IP addresses, accounts, Cloud resources, virtual machines, etc."[6] **Graph forming**: Fusion builds and continually updates a hyperconnected graph on large scale data sets, typically millions of anomalous signals in a customer workspace. In the graph, the nodes represent the entities and the activities, and the edges represent the relationships between the nodes. The activities are the alerts and anomalies from different sources. The entities can be IP addresses, accounts, Cloud resources, virtual machines, etc. *Figure 1: Graph formed from a Microsoft Sentinel workspace*

[6] *Id.*

	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.
receive streaming data comprising time-stamped data about events relating to one or more entities of the first plurality of entities, based on a first portion of the streaming data, identify a first entity that does not correspond to any of the first plurality of nodes, wherein the first entity is not of the first plurality of entities, based on a second portion of the streaming data, wherein the second portion is not identical to the first portion, identify a first relationship between a pair of entities of the first plurality of entities that does not correspond to	The software instructions of the Accused ’627 Fusion Products receive streaming data comprising time-stamped data about events relating to one or more entities of the first plurality of entities, based on a first portion of the streaming data, identify a first entity that does not correspond to any of the first plurality of nodes, wherein the first entity is not of the first plurality of entities, and based on a second portion of the streaming data, wherein the second portion is not identical to the first portion, identify a first relationship between a pair of entities of the first plurality of entities that does not correspond to any of the first plurality of edges. For example, Microsoft’s documentation explains that “Fusion correlates signals from multiple clouds, on-premise, and at the Edge for your entire enterprise, including anomalies, alerts from Microsoft products, as well as alerts from scheduled analytics rules . . . helping you to automatically detect sophisticated, multistage attacks”:[7]

[7] *Id.*

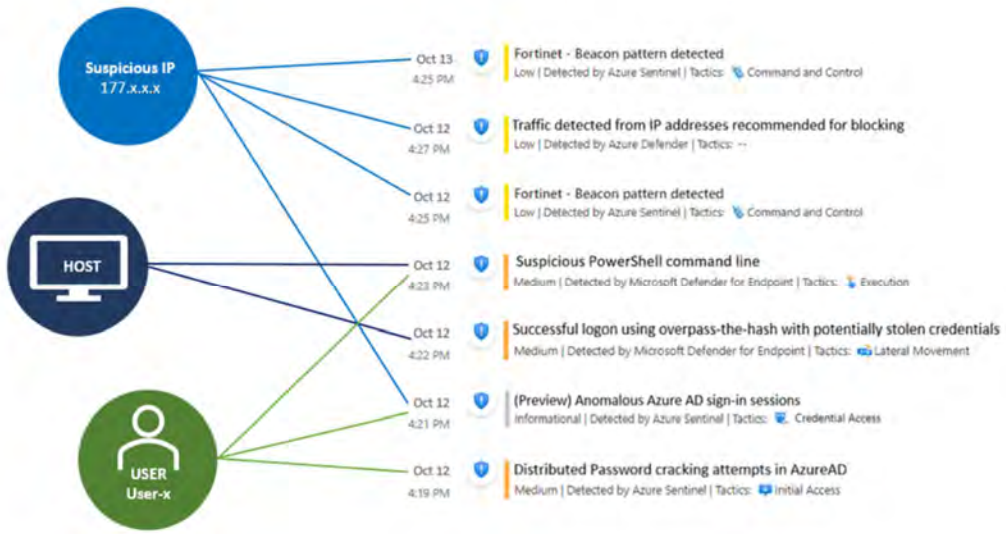
any of the first plurality of edges,	From Anomalous Signals to High Fidelity Incidents: How Fusion Works End to End Fusion operates a series of patented machine learning algorithms to look for advanced attacks from millions of anomalous signals. The process includes graph forming, attack pattern matching, expansion, scoring, and incident creation.  **Anomalous signals:** Fusion correlates signals from multiple clouds, on-premise, and at the Edge for your entire enterprise, including anomalies, alerts from Microsoft products, as well as alerts from scheduled analytics rules - both built-in and those created by your security analysts — helping you to automatically detect sophisticated, multistage attacks. As another example, Microsoft's documentation explains that Fusion is "a correlation engine based on scalable machine learning algorithms, to automatically detect multistage attacks by identifying combinations of anomalous behaviors and suspicious activities that are observed at various stages of the attack chain. Based on these discoveries, Microsoft Sentinel generates incidents that would otherwise be difficult to catch."[8] Microsoft's documentation further explains that Fusion "can help you find the emerging and unknown threats in your environment by applying extended ML analysis and by

[8] Microsoft, *Configure multistage attack detection (Fusion) rules in Microsoft Sentinel*, available at `https://learn.microsoft.com/en-us/azure/sentinel/configure-fusion-rules` [hereinafter *Configure Fusion Rules*].

	correlating a broader scope of anomalous signals."[9]
	
	As another example, Microsoft's documentation explains that Fusion collects data from multiple sources, including "[o]ut-of-the-box anomaly detections," "[a]lerts from Microsoft services," and "[a]lerts from scheduled analytics rules."[10]
	
	Fusion for emerging threats supports data collection and analysis from the following sources:
	
	• Out-of-the-box anomaly detections
	
	• Alerts from Microsoft services:
	  ○ Microsoft Entra ID Protection
	  ○ Microsoft Defender for Cloud
	  ○ Microsoft Defender for IoT
	  ○ Microsoft Defender XDR
	  ○ Microsoft Defender for Cloud Apps
	  ○ Microsoft Defender for Endpoint
	  ○ Microsoft Defender for Identity
	  ○ Microsoft Defender for Office 365
	
	• Alerts from scheduled analytics rules. Analytics rules must contain kill-chain (tactics) and entity mapping information in order to be used by Fusion.
	
	As another example, Microsoft's documentation provides the following example of "a possible attack" detected by Fusion that "started with initial access from the Cloud to end point execution, and then moved on to consistent beaconing from an internal IP address to a suspicious external one in roughly 24

[9] Microsoft, *Advanced multistage attack detection in Microsoft Sentinel*, available at `https://learn.microsoft.com/en-us/azure/sentinel/fusion` [hereinafter *Advanced Multistage Attack Detection*].

[10] *Id.*

	hours":[11] **An example** The example below shows a possible attack started with initial access from the Cloud to end point execution, and then moved on to consistent beaconing from an internal IP address to a suspicious external one in roughly 24 hours. The Fusion ML algorithms detected this attack by correlating anomaly (Anomalous Azure AD sign-in sessions), as well as alerts from custom scheduled rules, Azure Defender, and Microsoft Defender for Endpoint.  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.

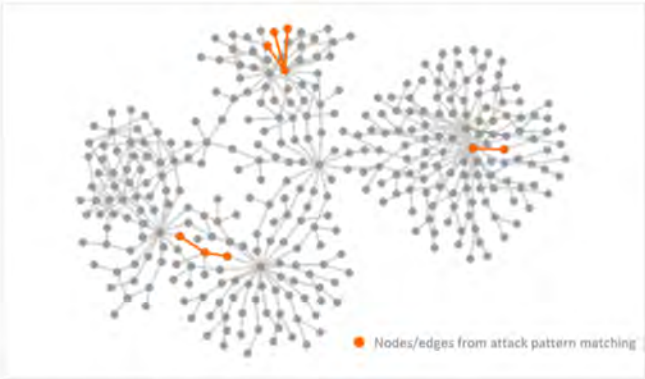
[11] Microsoft, *Detecting Emerging Threats with Microsoft Sentinel Fusion*, available at `https://techcommunity.microsoft.com/blog/microsoftsentinelblog/detecting-emerging-threats-with-microsoft-sentinel-fusion/2923835` [hereinafter *Detecting Emerging Threats*].

	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.
modify, in the hardware memory, the representation of the first graph to generate a modified representation of the first graph, wherein the modified representation of the first graph comprises a representation of a first node corresponding to the first entity and a representation of a first edge corresponding to the first relationship, wherein the first node is not of the first plurality of nodes and the first edge is not of the first plurality of edges, identify, based on the modified representation of the first graph, an attack path that could be involved in an attack involving the first entity,	The software instructions of the Accused ’627 Fusion Products modify, in the hardware memory, the representation of the first graph to generate a modified representation of the first graph, wherein the modified representation of the first graph comprises a representation of a first node corresponding to the first entity and a representation of a first edge corresponding to the first relationship, wherein the first node is not of the first plurality of nodes and the first edge is not of the first plurality of edges, and identify, based on the modified representation of the first graph, an attack path that could be involved in an attack involving the first entity, wherein identifying the attack path comprises identifying a second entity that can be reached using the first entity, wherein the second entity corresponds to a second node, and the second node is related by one or more edges to the first node corresponding to the first entity in the modified representation of the first graph; and, identifying a third entity that can be reached using the second entity, wherein the third entity corresponds to a third node, and the third node is related by one or more edges to the second node in the modified representation of the first graph. For example, Microsoft’s documentation explains that Fusion “builds and continually updates a hyperconnected graph on large scale data sets, typically millions of anomalous signals in a customer workspace.”[12] Fusion uses the “hyperconnected graph” to perform “[a]ttack pattern matching” that identifies “subgraphs representing possible attacks,” based on attack patterns “consist[ing] of activities (nodes), entities (nodes), and their relationships (edges).”[13] For example, in the below example reproduced from Microsoft’s documentation, Fusion identifies “4 nodes and 3 edges in the top subgraph”:[14]

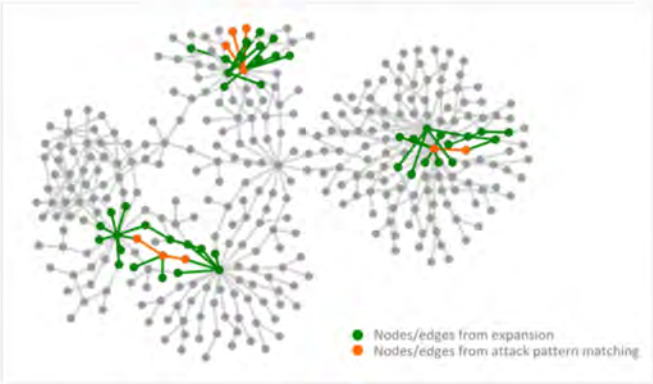
[12] *Fusion Behind the Scenes.*

[13] *Id.*

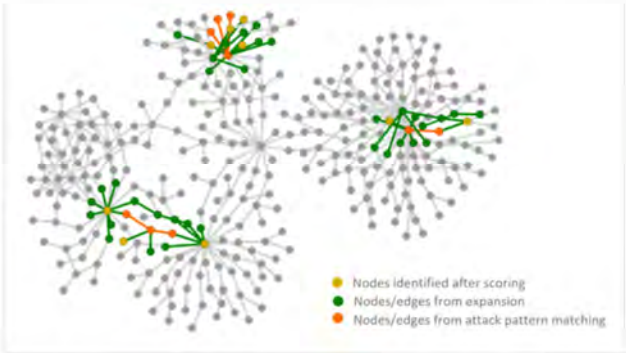
[14] *Id.*

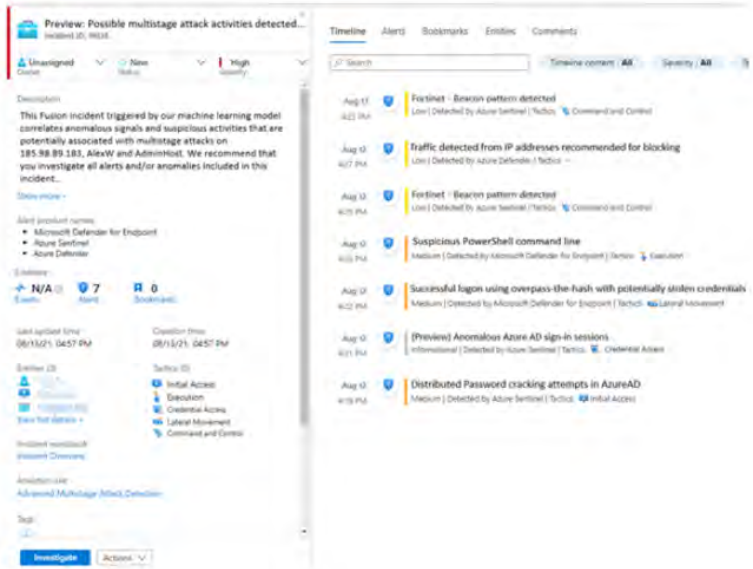
wherein identifying the attack path comprises: identifying a second entity that can be reached using the first entity, wherein the second entity corresponds to a second node, and the second node is related by one or more edges to the first node corresponding to the first entity in the modified representation of the first graph; and, identifying a third entity that can be reached using the second entity, wherein the third entity corresponds to a third node, and the third node is related by one or more edges to the second node in the modified representation of the first graph; and	**Attack pattern matching**: Fusion keeps a large set of attack patterns in a knowledge pool, including known attack patterns and ML generated emerging attack patterns. The known attack patterns are derived from past true positive incidents and security research. We will deep dive into how ML generates the emerging attack patterns in the next section of the blog. An attack pattern consists of activities (nodes), entities (nodes), and their relationships (edges). In this step, Fusion constantly takes attack patterns from the knowledge pool and identifies matches in the hyperconnected graph. Those identified matches are called subgraphs. This step reduces the millions of anomalous signals to a smaller set of subgraphs representing possible attacks. In the example below, three attack patterns are matched in the graph. There are 4 nodes and 3 edges in the top subgraph.  *Figure 2: Simplified graph shows nodes and edges from attack pattern matching* Microsoft's documentation further explains that Fusion "expands the matched attack patterns to discover additional activities and entities that are relevant."[15]

[15] *Id.*

	Expansion: During the expansion phase, Fusion expands the matched attack patterns to discover additional activities and entities that are relevant.  *Figure 3: Simplified graph shows nodes and edges from attack pattern matching and expansion* Fusion then performs "[s]coring and incident creation" to "trigger[] incidents that include[] the most relevant alerts, anomalies, and entities" by "calculat[ing] the killchain reachability of an attack and identify the nodes that have highest relevance in a real attack."[16]

[16] *Id.*

	Scoring and incident creation: Once the subgraphs representing possible attacks are identified, Fusion applies a round of scoring and triggers incidents that includes the most relevant alerts, anomalies, and entities to further reduce alert volume and speedup investigation. In this step, Fusion uses k-nearest neighbors (KNN) to calculate the killchain reachability of an attack and identify the nodes that have highest relevance in a real attack. In the example below, all the colored nodes (orange, yellow, green) are relevant to an attack. After the scoring round, Fusion only surfaces the nodes that have the highest relevance (orange and yellow colored nodes) in an incident. This way the security analysts only need to investigate a focused set of the most relevant activities and entities to quickly understand an attack.  *Figure 6: Simplified graph shows nodes and edges from attack pattern matching, expansion and scoring* 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.
generate a report comprising an identification of the first entity and at least one of the	The software instructions of the Accused '627 Fusion Products generate a report comprising an identification of the first entity and at least one of the second entity and the third entity.

second entity and the third entity.	For example, Microsoft's documentation reproduces a screenshot of an example "Possible multistage attack activities detected" report, reproduced below, which includes an identification of various entities:[17] The example in *Figure 7* shows a possible attack that started with initial access from the Cloud to endpoint execution, and then moved on to consistent beaconing from an internal IP address to a suspicious external IP address, and possible Command and Control in roughly 24 hours. The Fusion ML algorithms detected this attack by correlating an anomaly (Anomalous Azure AD sign-in sessions), as well as alerts from custom scheduled rules, Azure Defender, and Microsoft Defender for Endpoint.  *Figure 7: Fusion incident in Microsoft Sentinel workspace*

[17] *Id.*

II. Claim 5

The computer system of claim 1	See above for an analysis of Claim 1.

wherein the modified representation of the first graph comprises a representation of a node corresponding to the first entity, wherein the first entity is identified based on active reconnaissance results.	The modified representation of the first graph of Claim 1 comprises a representation of a node corresponding to the first entity, wherein the first entity is identified based on active reconnaissance results.
	
	As explained above, Fusion "supports data collection and analysis" from multiple sources, including, among others, Microsoft Defender for Endpoint.[18]
	
	Fusion for emerging threats supports data collection and analysis from the following sources:
	
	• Out-of-the-box anomaly detections
	
	• Alerts from Microsoft services:
	  ○ Microsoft Entra ID Protection
	  ○ Microsoft Defender for Cloud
	  ○ Microsoft Defender for IoT
	  ○ Microsoft Defender XDR
	  ○ Microsoft Defender for Cloud Apps
	  ○ Microsoft Defender for Endpoint
	  ○ Microsoft Defender for Identity
	  ○ Microsoft Defender for Office 365
	
	• Alerts from scheduled analytics rules. Analytics rules must contain kill-chain (tactics) and entity mapping information in order to be used by Fusion.

[18] *Advanced Multistage Attack Detection.*

	Defender for Endpoint, for example, includes "Standard discovery," which is a "device discovery capability that helps you find unmanaged devices connected to your corporate network," using "active probing to discover additional information about observed devices to enrich existing device information."[19] For example, Standard discovery "uses various PowerShell scripts to actively probe devices in the network. Those PowerShell scripts are Microsoft signed and are executed from the following location: `C:\ProgramData\Microsoft\Windows Defender Advanced Threat Protection\Downloads\*.ps`. For example, `C:\ProgramData\Microsoft\Windows Defender Advanced Threat Protection\Downloads\UnicastScannerV1.1.0.ps1`."[20] 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.

[19] Microsoft, *Device discovery overview*, available at `https://learn.microsoft.com/en-us/defender-endpoint/device-discovery` [hereinafter *Device Discovery*].

[20] Microsoft, *Configure device discovery in Defender for Endpoint*, available at `https://learn.microsoft.com/en-us/defender-endpoint/configure-device-discovery`.

III. Claim 6

The computer system of claim 1	See above for an analysis of Claim 1.
wherein the modified representation of the first graph comprises a representation of a node corresponding to the first entity, wherein the first entity is identified based on passive reconnaissance results.	The modified representation of the first graph of Claim 1 comprises a representation a node corresponding to the first entity, wherein the first entity is identified based on passive reconnaissance results. For example, as noted above, Fusion collects and analyzes data from, among other sources, Microsoft Defender for Endpoint.[21] Defender for Endpoint includes "Basic discovery," which is a "device discovery capability that helps you find unmanaged devices connected to your corporate network" by "passively collect[ing] events in your network and extract[ing] device information from them," "us[ing] the `SenseNDR.exe` binary for passive network data collection."[22] Microsoft's documentation further explains that "no network traffic is initiated. Endpoints extract data from all network traffic seen by an onboarded device."[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.

[21] *Advanced Multistage Attack Detection.*

[22] *Id.*

[23] *Device Discovery.*

IV. Claim 11

11. The computer system of claim 1,	See above for an analysis of Claim 1.
wherein the computer system is further configured to execute software instructions stored on nontransitory machine-readable storage media comprising software instructions that determine whether an event, of the events relating to one or more entities of the first plurality of entities, is anomalous, wherein determining whether the event is anomalous comprises: determining that the event relates to the first entity,	The computer system of Claim 1 is further configured to execute software instructions stored on nontransitory machine-readable storage media comprising software instructions that determine whether an event, of the one or more events relating to one or more entities of the first plurality of entities, is anomalous, wherein determining whether the event is anomalous comprises determining that the event relates to the first entity, determining at least one behavior pattern associated with the first entity, and comparing the event to the at least one behavior pattern. For example, as explained above, Fusion “identif[ies] combinations of anomalous behaviors and suspicious activities that are observed at various stages of the kill chain.”[24] Fusion “supports data collection and analysis” from a variety of sources, including Sentinel’s “[o]ut-of-the-box anomaly detections.”[25] Such out-of-the-box anomaly detections include, for example, “UEBA anomalies” which “detect[] anomalies based on each entity's baseline historical behavior across various environments. Each entity's baseline behavior is set according to its own historical activities, those of its peers, and those of the organization as a whole. Anomalies can be triggered by the correlation of different attributes such as action type, geo-location, device, resource, ISP, and more.”[26]

[24] *Advanced Multistage Attack Detection*.

[25] *Id*.

[26] Microsoft, *Use customizable anomalies to detect threats in Microsoft Sentinel*, available at `https://learn.microsoft.com/en-us/azure/sentinel/soc-ml-anomalies` [hereinafter *Customizable Anomalies*].

determining at least one behavior pattern associated with the first entity, and comparing the event to the at least one behavior pattern.	**UEBA anomalies** Some of Microsoft Sentinel's anomaly detections come from its User and Entity Behavior Analytics (UEBA) engine, which detects anomalies based on each entity's baseline historical behavior across various environments. Each entity's baseline behavior is set according to its own historical activities, those of its peers, and those of the organization as a whole. Anomalies can be triggered by the correlation of different attributes such as action type, geo-location, device, resource, ISP, and more. As another example, and as noted above, Fusion "supports data collection and analysis" from multiple sources, including, among others Microsoft Defender for Cloud Apps.[27]

[27] *Advanced Multistage Attack Detection.*

	Fusion for emerging threats supports data collection and analysis from the following sources: • Out-of-the-box anomaly detections • Alerts from Microsoft services: ○ Microsoft Entra ID Protection ○ Microsoft Defender for Cloud ○ Microsoft Defender for IoT ○ Microsoft Defender XDR ○ Microsoft Defender for Cloud Apps ○ Microsoft Defender for Endpoint ○ Microsoft Defender for Identity ○ Microsoft Defender for Office 365 • Alerts from scheduled analytics rules. Analytics rules must contain kill-chain (tactics) and entity mapping information in order to be used by Fusion. Microsoft documentation explains that Defender for Cloud Apps uses "multiple detection methods, including anomaly, behavioral analytics (UEBA), and rule-based activity detections, to provide a broad view of how your users use apps in your environment."[28] For example, Defender for Cloud Apps uses "user and entity behavioral analytics (UEBA) and machine learning (ML)" to "target[] numerous behavioral anomalies across your users and the machines and devices connected to your network" using

[28] Microsoft, *Tutorial: Detect suspicious user activity with behavioral analytics (UEBA)*, available at `https://learn.microsoft.com/en-us/defender-cloud-apps/tutorial-suspicious-activity` [hereinafter *Detect Suspicious User Activity*].

	a "heuristic anomaly detection engine that profiles your environment and triggers alerts with respect to a baseline that was learned on your organization's activity."[29] **Create Defender for Cloud Apps anomaly detection policies** The Microsoft Defender for Cloud Apps anomaly detection policies provide out-of-the-box user and entity behavioral analytics (UEBA) and machine learning (ML) so that you're ready from the outset to run advanced threat detection across your cloud environment. Because they're automatically enabled, the new anomaly detection policies immediately start the process of detecting and collating results, targeting numerous behavioral anomalies across your users and the machines and devices connected to your network. In addition, the policies expose more data from the Defender for Cloud Apps detection engine, to help you speed up the investigation process and contain ongoing threats. The anomaly detection policies are automatically enabled, but Defender for Cloud Apps has an initial learning period of seven days during which not all anomaly detection alerts are raised. After that, as data is collected from your configured API connectors, each session is compared to the activity, when users were active, IP addresses, devices, and so on, detected over the past month and the risk score of these activities. Be aware that it may take several hours for data to be available from API connectors. These detections are part of the heuristic anomaly detection engine that profiles your environment and triggers alerts with respect to a baseline that was learned on your organization's activity. These detections also use machine-learning algorithms designed to profile the users and sign in pattern to reduce false positives. 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

[29] Microsoft, *Create Defender for Cloud Apps anomaly detection policies*, available at `https://learn.microsoft.com/en-us/defender-cloud-apps/anomaly-detection-policy` [hereinafter *Anomaly Detection Policies*].

	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.

V. Claim 12

The computer system of claim 11,	See above for an analysis of Claim 11.
wherein comparing the event to the at least one behavior pattern comprises using a threshold.	Comparing the event to the at least one behavior pattern of Claim 11 comprises using a threshold. For example, Microsoft's documentation explains that Fusion's anomaly detections can be "tuned."[30] "[T]hresholds and parameters can be configured and fine-tuned through the . . . analytics rule user interface."[31] Anomalies can be powerful tools, but they are notoriously noisy. They typically require a lot of tedious tuning for specific environments, or complex post-processing. Customizable anomaly templates are tuned by Microsoft Sentinel's data science team to provide out-of-the-box value. If you need to tune them further, the process is simple and requires no knowledge of machine learning. The thresholds and parameters for many of the anomalies can be configured and fine-tuned through the already familiar analytics rule user interface. The performance of the original threshold and parameters can be compared to the new ones within the interface and further tuned as necessary during a testing, or flighting, phase. Once the anomaly meets the performance objectives, the anomaly with the new threshold or parameters can be promoted to production with the click of a button. Microsoft Sentinel customizable anomalies enable you to get the benefit of anomaly detection without the hard work. As another example, and as noted above, Microsoft's documentation indicates that Fusion collects and analyzes data from, among other sources, Microsoft Defender for Cloud Apps.[32] Defender for Cloud Apps allows users to set "dynamic thresholds":[33]

[30] *Customizable Anomalies.*

[31] *Id.*

[32] *Advanced Multistage Attack Detection.*

[33] *Detect Suspicious User Activity.*

	Next, you want to tune your policies. The following policies can be fine-tuned by setting filters, dynamic thresholds (UEBA) to help train their detection models, and suppressions to reduce common false positive detections: • Anomaly detection • Cloud discovery anomaly detection • Rule-based activity detection As another example, Defender for Cloud Apps uses a "sensitivity slider" to "determine[] the level of suppressions applied to anomalous behavior before triggering an impossible travel alert," offering "Low," "Medium," and "High" sensitivity levels:[34] 3. **Tune sensitivity of impossible travel** Configure the sensitivity slider that determines the level of suppressions applied to anomalous behavior before triggering an impossible travel alert. For example, organizations interested in high fidelity should consider increasing the sensitivity level. On the other hand, if your organization has many users that travel, consider lowering the sensitivity level to suppress activities from a user's common locations learned from previous activities. You can choose from the following sensitivity levels: • **Low**: System, tenant, and user suppressions • **Medium**: System and user suppressions • **High**: Only system suppressions As another example, Defender for Cloud Apps offers the ability to tune "the volume of activity required before the detection raises an alert":[35]

[34] *Id.*

[35] *Id.*

	Phase 4: Tune rule-based detection (activity) policies Rule-based detection policies give you the ability to complement anomaly detection policies with organization-specific requirements. We recommend creating rules-based policies using one of our Activity policy templates (go to **Control** > **Templates** and set the **Type** filter to **Activity policy**) and then configuring them to detect behaviors that aren't normal for your environment. For example, for some organization that don't have any presence in a particular country/region, it may make sense to create a policy that detects the anomalous activities from that country/region and alert on them. For others, who have large branches in that country/region, activities from that country/region would be normal and it wouldn't make sense to detect such activities. 1. **Tune activity volume** Choose the volume of activity required before the detection raises an alert. Using our country/region example, if you have no presence in a country/region, even a single activity is significant and warrants an alert. However, a single sign-in failure could be human error and only of interest if there are many failures in a short period. 2. **Tune activity filters** Set the filters you require to detect the type of activity you want to alert on. For example, to detect activity from a country/region, use the **Location** parameter. 3. **Tune alerts** To prevent alert fatigue, set the **daily alert limit**. 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.

VI. Claim 14

The computer system of claim 1,	See above for an analysis of Claim 1.
wherein the computer system comprises a plurality of physical computing machines.	The computer system of Claim 1 comprises a plurality of physical computing machines. For example, as explained above, Microsoft Sentinel is a cloud software platform built on, among other things, "a modern data lake."[36] The Sentinel data lake "is a fully managed, cloud-native data lake" that "unifies, retains, and analyzes security data."[37] Microsoft's documentation explains that Sentinel data lake usage is priced based on the amount of data processed, the amount of data analyzed, compute hours, and the amount of data stored.[38] **Sentinel** Microsoft Sentinel data lake enables security teams to ingest, retain, and analyze massive volumes of security data cost-effectively. With separate compute and storage meters, the data lake allows defenders to flexibly run advanced data insights, machine learning, and forensic investigations from a single point.[4] Learn more. SKU / Meter type / Price Data lake ingestion / Data Processed (GB) / $0.05 USD Data processing / Data Processed (GB) / $0.1 USD Data lake query / Data Analyzed (GB) / $0.005 USD Advanced Data Insights / 1 Compute Hour / $0.15 USD Data lake storage / Data Stored (GB/Month) / $0.026 USD

[36] *What is Microsoft Sentinel?*.

[37] *Id*.

[38] *Sentinel Pricing*.

	One of skill in the art would thus understand that Fusion comprises a plurality of physical computing machines. 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.

VII. Claim 16

The computer system of claim 1,	See above for an analysis of Claim 1.
wherein at least one entity of the first plurality of entities is at least one of a user, a place, a device, a resource, a group, or a service.	At least one entity of the first plurality of entities of Claim 1 is at least one of a user, a place, a device, a resource, a group, or a service. For example, as explained above, nodes in Fusion “represent . . . entities” such as “IP addresses, accounts, Cloud resources, virtual machines, etc.”[39] As another example, Microsoft’s documentation provides a list of the types of entities identified by Sentinel:[40]

[39] *Fusion Behind the Scenes*.

[40] Microsoft, *Entities in Microsoft Sentinel*, available at https://learn.microsoft.com/en-us/azure/sentinel/entities.

	The following types of entities are currently identified in Microsoft Sentinel: • Account • Host • IP address • URL • Azure resource • Cloud application • DNS resolution • File • File hash • Malware • Process • Registry key • Registry value • Security group • Mailbox • Mail cluster • Mail message • Submission mail 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.

VIII. Claim 17

The computer system of claim 1,	See above for an analysis of Claim 1.
wherein the representation of the first graph comprises a representation of at least one node that does not correspond to an entity.	The representation of the first graph of Claim 1 comprises a representation of at least one node that does not correspond to an entity. For example, as noted above, nodes in Fusion's graph representation can represent "activities" in addition to "entities."[41] "The activities are the alerts and anomalies from different sources."[42] **Graph forming**: Fusion builds and continually updates a hyperconnected graph on large scale data sets, typically millions of anomalous signals in a customer workspace. In the graph, the nodes represent the entities and the activities, and the edges represent the relationships between the nodes. The activities are the alerts and anomalies from different sources. The entities can be IP addresses, accounts, Cloud resources, virtual machines, etc. 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.

[41] *Fusion Behind the Scenes.*

[42] *Id.*

IX. Claim 18

A computer system comprising: a hardware memory, wherein the computer system is configured to execute software instructions stored on nontransitory machine-readable storage media comprising software instructions that:	The Accused '627 Fusion Products include a computer system comprising a hardware memory, wherein the computer system is configured to execute software instructions stored on nontransitory machine-readable storage media comprising software instructions that perform as discussed below. For example, as discussed above, Microsoft Sentinel "is a cloud-native Security Information and Event Management (SIEM) and unified security platform . . . . built on a modern data lake."[43] The Sentinel data lake "unifies, retains, and analyzes security data at scale - providing the foundation for advanced analytics, AI-driven insights, and agentic defense,"[44] and usage of the Sentinel data lake is priced based on the amount of data processed, the amount of data analyzed, compute hours, and the amount of data stored.[45]
store in the hardware memory a representation of a first graph, wherein the representation of the first graph comprises representations of a first plurality of nodes corresponding to a first plurality of entities and further comprises	The software instructions of the Accused '627 Fusion Products store in the hardware memory a representation of a first graph, wherein the representation of the first graph comprises representations of a first plurality of nodes corresponding to a first plurality of entities and further comprises representations of a first plurality of edges, wherein the first graph is a directed graph. For example, Microsoft's documentation explains that Fusion "builds and continually updates a hyperconnected graph on large scale data sets" in which "nodes represent the entities and the activities, and the edges represent the relationships between the nodes. . . . The entities can be IP addresses, accounts, Cloud resources, virtual machines, etc":[46]

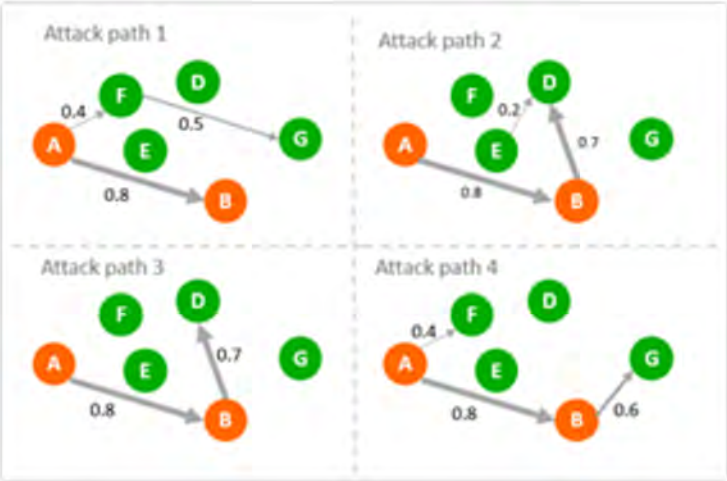
[43] *What is Microsoft Sentinel?*.

[44] *Id*.

[45] *Sentinel Pricing*.

[46] *Fusion Behind the Scenes*.

representations of a first plurality of edges,	**Graph forming**: Fusion builds and continually updates a hyperconnected graph on large scale data sets, typically millions of anomalous signals in a customer workspace. In the graph, the nodes represent the entities and the activities, and the edges represent the relationships between the nodes. The activities are the alerts and anomalies from different sources. The entities can be IP addresses, accounts, Cloud resources, virtual machines, etc. *Figure 1: Graph formed from a Microsoft Sentinel workspace* 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.
wherein the first graph is a directed graph,	The first graph of the Accused '627 Fusion Products is a directed graph.

	For example, Microsoft's documentation shows directionality of edges in Fusion's graph visualization:[47]  *Figure 4: Expansion - probabilistic random walk*

[47] *Id.*

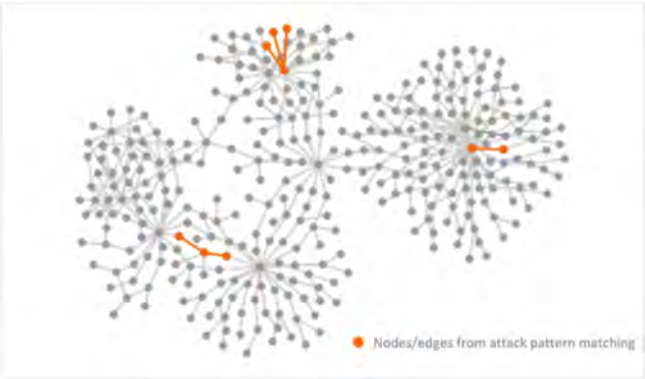
	 Figure 5: Expansion - aggregate weights and apply threshold Based on Qomplx's analysis to date, this element appears to be practiced literally, including as charted above. Any differences between the charted material and this claim element are expected to be insubstantial, in which case the element is 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.
wherein the first plurality of entities comprises a plurality of accounts and a plurality of resources, and wherein each edge of the first plurality of edges corresponds to a respective relationship between a respective pair of entities;	The first plurality of entities of the Accused '627 Fusion Products comprises a plurality of accounts and a plurality of resources, and each edge of the first plurality of edges of the Accused Fusion Products corresponds to a respective relationship between a respective pair of entities of the first plurality of entities. For example, as explained above, nodes of Fusion's graph representation "represent the entities and the activities, and the edges represent the relationships between the nodes. . . . The entities can be IP addresses, accounts, Cloud resources, virtual machines, etc."[48] 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.

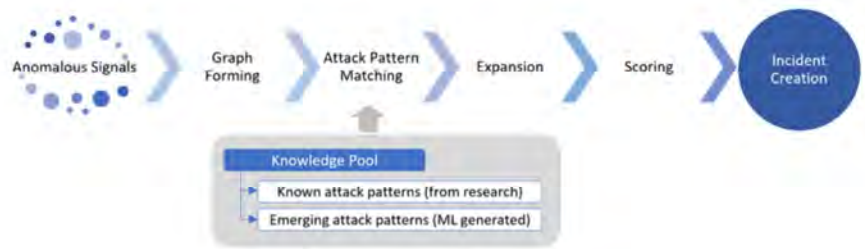
[48] *Id.*

	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.
identify, based on the representation of the first graph, a first plurality of attack paths comprising a first entity of the first plurality of entities, wherein each attack path of the first plurality of attack paths targets a second entity of the first plurality of entities,	The software instructions of the Accused ’627 Fusion Products identify, based on the representation of the first graph, a first plurality of attack paths comprising a first entity of the first plurality of entities, wherein each attack path of the first plurality of attack paths targets a second entity of the first plurality of entities. For example, as explained above, Microsoft’s documentation explains that Fusion “builds and continually updates a hyperconnected graph” of nodes and edges in a customer environment.[49] Fusion uses the “hyperconnected graph” to perform “[a]ttack pattern matching” and identify “subgraphs representing possible attacks,” based on attack patterns “consist[ing] of activities (nodes), entities (nodes), and their relationships (edges).”[50]

[49] *Id.*

[50] *Id.*

	Attack pattern matching: Fusion keeps a large set of attack patterns in a knowledge pool, including known attack patterns and ML generated emerging attack patterns. The known attack patterns are derived from past true positive incidents and security research. We will deep dive into how ML generates the emerging attack patterns in the next section of the blog. An attack pattern consists of activities (nodes), entities (nodes), and their relationships (edges). In this step, Fusion constantly takes attack patterns from the knowledge pool and identifies matches in the hyperconnected graph. Those identified matches are called subgraphs. This step reduces the millions of anomalous signals to a smaller set of subgraphs representing possible attacks. In the example below, three attack patterns are matched in the graph. There are 4 nodes and 3 edges in the top subgraph.  *Figure 2: Simplified graph shows nodes and edges from attack pattern matching* 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.
receive streaming data comprising time-stamped data about events relating to one or	The software instructions of the Accused '627 Fusion Products receive streaming data comprising time-stamped data about events relating to one or more entities of the first plurality of entities, based on a first portion of the streaming data, identify a third entity that does not correspond to any of the first plurality of nodes, wherein the third entity is not of the first plurality of entities, and based on a second portion of

more entities of the first plurality of entities, based on a first portion of the streaming data, identify a third entity that does not correspond to any of the first plurality of nodes, wherein the third entity is not of the first plurality of entities, based on a second portion of the streaming data, wherein the second portion is not identical to the first portion, identify a first relationship between a pair of entities of the first plurality of entities that does not correspond to any of the first plurality of edges,	the streaming data, wherein the second portion is not identical to the first portion, identify a first relationship between a pair of entities of the first plurality of entities that does not correspond to any of the first plurality of edges. For example, Microsoft's documentation explains that Fusion collects and analyzes "signals from multiple clouds, on-premise, and at the Edge for your entire enterprise, including anomalies, alerts from Microsoft products, as well as alerts from scheduled analytics rules."[51] From Anomalous Signals to High Fidelity Incidents: How Fusion Works End to End Fusion operates a series of patented machine learning algorithms to look for advanced attacks from millions of anomalous signals. The process includes graph forming, attack pattern matching, expansion, scoring, and incident creation.  **Anomalous signals:** Fusion correlates signals from multiple clouds, on-premise, and at the Edge for your entire enterprise, including anomalies, alerts from Microsoft products, as well as alerts from scheduled analytics rules - both built-in and those created by your security analysts — helping you to automatically detect sophisticated, multistage attacks. Fusion uses this streaming data to "continually update[]" its "hyperconnected graph" of nodes and edges

[51] *Fusion Behind the Scenes*.

	in the customer environment.[52] As another example, Microsoft's documentation explains that Fusion is "identif[ies] combinations of anomalous behaviors and suspicious activities that are observed at various stages of the attack chain. Based on these discoveries, Microsoft Sentinel generates incidents that would otherwise be difficult to catch."[53] As another example, Microsoft's documentation explains that Fusion collects data from multiple sources, including "[o]ut-of-the-box anomaly detections," "[a]lerts from Microsoft services," and "[a]lerts from scheduled analytics rules":[54]

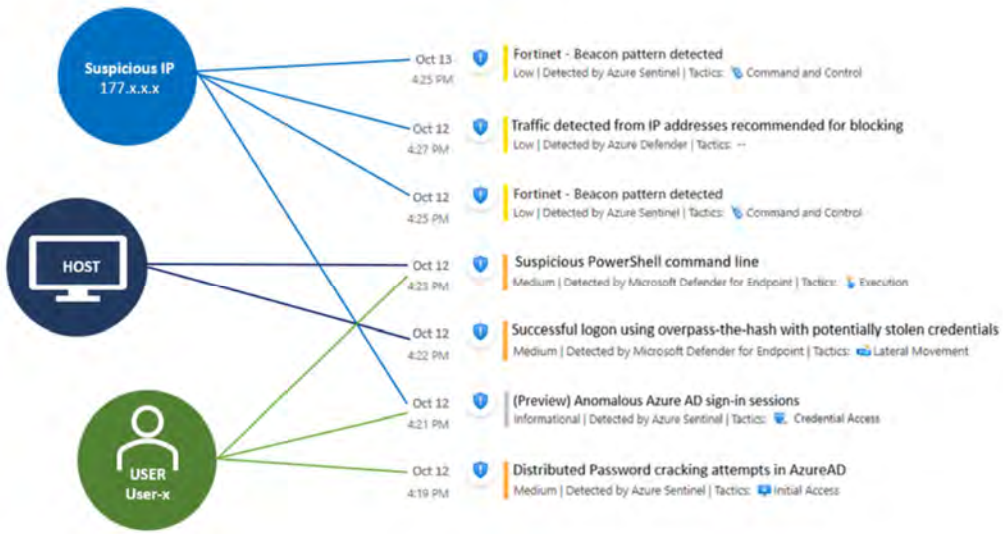
[52] *Id*.

[53] *Configure Fusion Rules*.

[54] *Advanced Multistage Attack Detection*.

	Fusion for emerging threats supports data collection and analysis from the following sources: • Out-of-the-box anomaly detections • Alerts from Microsoft services: ○ Microsoft Entra ID Protection ○ Microsoft Defender for Cloud ○ Microsoft Defender for IoT ○ Microsoft Defender XDR ○ Microsoft Defender for Cloud Apps ○ Microsoft Defender for Endpoint ○ Microsoft Defender for Identity ○ Microsoft Defender for Office 365 • Alerts from scheduled analytics rules. Analytics rules must contain kill-chain (tactics) and entity mapping information in order to be used by Fusion. As another example, Microsoft's documentation provides an example, reproduced below, of "a possible attack" detected by Fusion that "started with initial access from the Cloud to end point execution, and then moved on to consistent beaconing from an internal IP address to a suspicious external one in roughly 24 hours":[55]

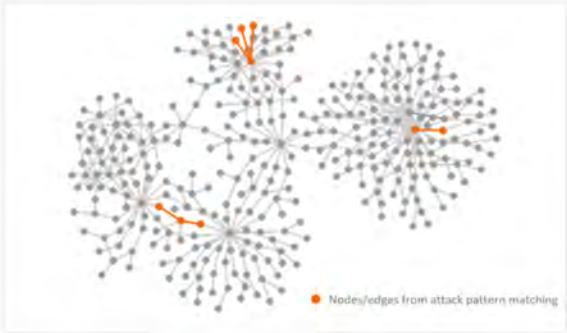
[55] *Detecting Emerging Threats*.

	An example The example below shows a possible attack started with initial access from the Cloud to end point execution, and then moved on to consistent beaconing from an internal IP address to a suspicious external one in roughly 24 hours. The Fusion ML algorithms detected this attack by correlating anomaly (Anomalous Azure AD sign-in sessions), as well as alerts from custom scheduled rules, Azure Defender, and Microsoft Defender for Endpoint.  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.
modify, in the hardware memory, the representation of the first graph to generate a modified representation of the	The Accused Fusion Products execute software instructions that modify, in the hardware memory, the representation of the first graph to generate a modified representation of the first graph, wherein the modified representation of the first graph comprises a representation of a node corresponding to the third entity and an edge corresponding to the first relationship, wherein the node is not of the first plurality of

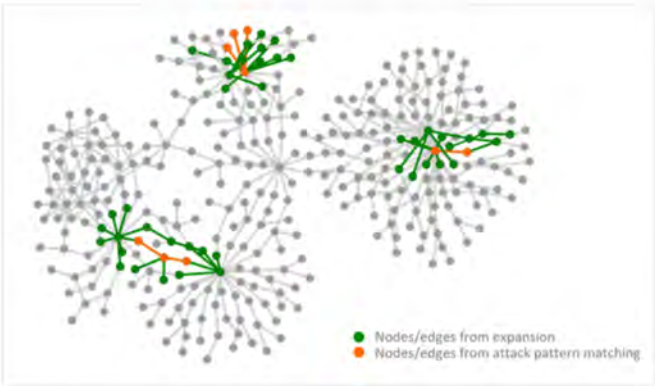
first graph, wherein the modified representation of the first graph comprises a representation of a node corresponding to the third entity and an edge corresponding to the first relationship, wherein the node is not of the first plurality of nodes and the edge is not of the first plurality of edges, and identify, based on the modified representation of the first graph, a second plurality of attack paths comprising the first entity, wherein each attack path of the second plurality of attack paths targets the second entity, and wherein an attack path of the second plurality of attack paths comprises the third entity.	nodes and the edge is not of the first plurality of edges, and identify, based on the modified representation of the first graph, a second plurality of attack paths comprising the first entity, wherein each attack path of the second plurality of attack paths targets the second entity, and wherein an attack path of the second plurality of attack paths comprises the third entity. For example, as explained above, Fusion uses its "hyperconnected graph" to perform "[a]ttack pattern matching" to identify "subgraphs representing possible attacks," based on attack patterns "consist[ing] of activities (nodes), entities (nodes), and their relationships (edges)."[56] As Fusion updates the graph, it discovers new attack patterns, including involving recently discovered entities. In addition, "Fusion constantly takes attack patterns from the knowledge pool and identifies matches in the hyperconnected graph."[57]

[56] *Fusion Behind the Scenes*.

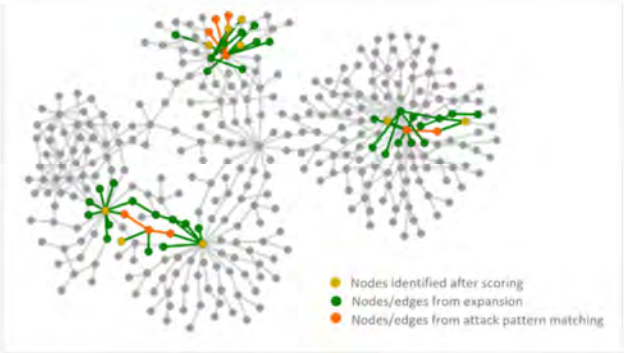
[57] *Id*.

	Attack pattern matching: Fusion keeps a large set of attack patterns in a knowledge pool, including known attack patterns and ML generated emerging attack patterns. The known attack patterns are derived from past true positive incidents and security research. We will deep dive into how ML generates the emerging attack patterns in the next section of the blog. An attack pattern consists of activities (nodes), entities (nodes), and their relationships (edges). In this step, Fusion constantly takes attack patterns from the knowledge pool and identifies matches in the hyperconnected graph. Those identified matches are called subgraphs. This step reduces the millions of anomalous signals to a smaller set of subgraphs representing possible attacks. In the example below, three attack patterns are matched in the graph. There are 4 nodes and 3 edges in the top subgraph.  *Figure 2: Simplified graph shows nodes and edges from attack pattern matching* Fusion "expands the matched attack patterns to discover additional activities and entities that are relevant" by, among other things, "determin[ing] the relevance of the nodes by taking information including time range, kill chain intent, severity, [and] entity type into consideration."[58]

[58] *Id.*

	Expansion: During the expansion phase, Fusion expands the matched attack patterns to discover additional activities and entities that are relevant.  *Figure 3: Simplified graph shows nodes and edges from attack pattern matching and expansion* Fusion then performs "[s]coring and incident creation" to "trigger[] incidents that include[] the most relevant alerts, anomalies, and entities" by "calculat[ing] the killchain reachability of an attack and identify the nodes that have highest relevance in a real attack."[59]

[59] *Id.*

	Scoring and incident creation: Once the subgraphs representing possible attacks are identified, Fusion applies a round of scoring and triggers incidents that includes the most relevant alerts, anomalies, and entities to further reduce alert volume and speedup investigation. In this step, Fusion uses k-nearest neighbors (KNN) to calculate the killchain reachability of an attack and identify the nodes that have highest relevance in a real attack. In the example below, all the colored nodes (orange, yellow, green) are relevant to an attack. After the scoring round, Fusion only surfaces the nodes that have the highest relevance (orange and yellow colored nodes) in an incident. This way the security analysts only need to investigate a focused set of the most relevant activities and entities to quickly understand an attack.  *Figure 6: Simplified graph shows nodes and edges from attack pattern matching, expansion and scoring* 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.

X. Claim 22

The computer system of claim 18,	See above for an analysis of Claim 18.
wherein a portion of the representation of the first graph is derived from reconnaissance data received by the computer system from a plurality of computer systems,	A portion of the representation of the first graph of Claim 18 is derived from reconnaissance data received by the computer system from a plurality of computer systems. For example, as explained above, Fusion collects and analyzes data from a variety of sources, including, among others, Microsoft Defender for Endpoint.[60] Defender for Endpoint uses "device discovery" to "find unmanaged devices connected to your corporate network," "us[ing] onboarded endpoints, in your network, to collect, probe, or scan your network to discover unmanaged devices."[61] Microsoft's documentation explains that "[t]he device discovery capability allows you to discover "[e]nterprise endpoints" like "workstations, servers, and mobile devices," "network devices like routers and switches," and "IoT devices like printers and cameras."[62] 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.
wherein a first computer system of the plurality of computer systems performs passive reconnaissance, and	A first computer system of the plurality of computer systems performs passive reconnaissance.

[60] *Advanced Multistage Attack Detection.*

[61] *Device Discovery.*

[62] *Id.*

	For example, as discussed above, Defender for Endpoint includes "Basic discovery," which "passively collect[s] events in your network and extract[s] device information from them" using "the `SenseNDR.exe` binary for passive network data collection."[63] 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.
wherein a second computer system of the plurality of computer systems performs active reconnaissance.	A second computer system of the plurality of computer systems performs active reconnaissance. For example, Defender for Endpoint includes "Network device discovery," in which a "designated Microsoft Defender for Endpoint device is used on each network segment to perform periodic authenticated scans of preconfigured network devices" such as "switches, routers, WLAN controllers, firewalls, and VPN gateways."[64] Microsoft's documentation explains that "[t]hese types of devices require an agentless approach where a remote scan obtains the necessary information from the devices. Depending on the network topology and characteristics, a single device or a few devices onboarded to Microsoft Defender for Endpoint performs authenticated scans of network devices using SNMP (read-only)."[65] 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.

[63] *Id.*

[64] Microsoft, *Network device discovery and vulnerability management*, available at `https://learn.microsoft.com/en-us/defender-endpoint/network-devices` [hereinafter *Network Device Discovery*].

[65] *Id.*

	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.

XI. Claim 24

The computer system of claim 18,	See above for an analysis of Claim 18.
further configured to execute software instructions stored on nontransitory machine-readable storage media comprising software instructions that analyze an attack path of the plurality of attack paths using a graph analysis algorithm.	The computer system of Claim 18 is further configured to execute software instructions stored on nontransitory machine-readable storage media comprising software instructions that analyze an attack path of the plurality of attack paths using a graph analysis algorithm. For example, Microsoft's documentation explains that Fusion runs a "probabilistic random walk" to "determine viable attack paths in the graph from the matched patterns. The model runs multiple times to simulate different attack paths." • ***Run probabilistic random walk***: a probabilistic kill chain model is then applied to determine viable attack paths in the graph from the matched patterns. The model runs multiple times to simulate different attack paths. In the example below, A and B represent the nodes in a matched attack pattern and D, E, F, G represent the relevant activities and entities. In the real world, the subgraphs and attack paths are much more complicated and can be time consuming for security analysts to manually complete the process. *Figure 4: Expansion - probabilistic random walk* 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.