Unified Patent Court
Einheitliches Patentgericht
Juridiction unifiée du brevet

Mannheim Local Division
UPC_CFI_660/2024
(CCfR: UPC_CFI_134/2025)

Decision

of the Court of First Instance of the Unified Patent Court

Local Division Mannheim

delivered on 19 December 2025

CLAIMANT:

Centripetal Limited,
Galway Technology Centre, Mervue Business Park, 7XPF+6C Galway, Irland

represented by: Armin Kühne

DEFENDANT:

Palo Alto Networks, Inc.
3000 Tannery Way, Santa Clara, 95054 CA, USA

represented by: Henrik Lehment

PATENT AT ISSUE:

European Patent No EP 3 652 914

PANEL/DIVISION:

Panel of the Local Division in Mannheim

DECIDING JUDGES:

This decision is delivered by the presiding judge Tochtermann, the legally qualified judge Sender as judge-rapporteur, the legally qualified judge Härmand and the technically qualified judge Kitchen.

LANGUAGE OF THE PROCEEDINGS: English

SUBJECT OF THE PROCEEDINGS: Infringement action and Counterclaim for revocation

DATE OF THE ORAL HEARING: 17 November 2025

SUMMARY OF THE FACTS:

1. Claimant sues Defendant for direct and indirect infringement of the German and French part of EP 3 652 914 B1 (cf. Exhibit C7, the patent-in-suit), relating to methods and systems to accelerate a cyberanalysis workflow. Claimant, a developer of network security hardware and software, is the registered and sole proprietor of the patent-in-suit, which is – *inter alia* – in force in Germany and France. The date of publication and mention of the grant of the patent-in-suit is 9 November 2022. It was filed on 10 July 2018, claiming the priority of two US patent applications of 10 July 2017 and 9 July 2018. The opt-out pertaining to the patent-in-suit was withdrawn from the register on 5 November 2024.

2. Claims 1, 2, 10 and 14 of the patent-in-suit, on which the alleged infringement is based on in combination in accordance with Claimant's unconditionally amended main request under R. 30.1 RoP, read as follows in the language of the patent:

   1. A method comprising:

      receiving a plurality of event logs, wherein each event log of the plurality of event logs is associated with an event that corresponds to one or more threat detection rules, wherein each event log corresponds to one or more actions that were applied, based on the one or more threat detection rules, to one or more packets;
      determining, by a computing device (130) and based on applying at least one algorithm to each event log, a reportability likelihood for each of the plurality of event logs, wherein the reportability
      likelihood indicates a likelihood that a given event log is associated with an event that is reportable to an authority;
      sorting an event queue of the plurality of event logs based on the reportability likelihood of each of the plurality of event logs; and
      transmitting, by the computing device and to an analysis system (140), the plurality of event logs sorted in the event queue.

   2. The method of claim 1, wherein the reportability likelihood is a combined reportability likelihood, and wherein determining, by the computing device, the reportability likelihood for each event log comprises:

      determining, by the computing device, a first reportability likelihood for each event log based on a static algorithm;

determining, by the computing device, a second reportability likelihood for each event log based on a machine-learned algorithm provided by a machine learning system (170); and
determining, by the computing device, the combined reportability likelihood for each event log based on the first reportability likelihood and the second reportability likelihood.

10. The method of any one of claims 1-9, further comprising:

receiving a plurality of packets;
determining, based on threat intelligence data, a plurality of potential threat communications events;
generating, based on the plurality of potential threat communications events, the plurality of event logs; and
storing the plurality of event logs to the event queue.

14. A system comprising:

a computing device (130) configured to perform the steps of any one of claims 1-12, and an analysis system (140) configured to receive the plurality of event logs.

3. Defendant, a US company, is a global provider of cybersecurity products, software and services, marketing its products – *inter alia* – in Germany and France, especially via the websites *www.paloaltonetworks.de* and *www.paloaltonetworks.fr*.

4. In Claimant's opinion, Defendant's cybersecurity cloud platforms **Cortex XSIAM** (Extended Security Intelligence and Analytics Management), **Cortex XDR** (version 3.4 or later), and their **SmartScore function**, implemented through Cortex XSIAM and/or Cortex XDR with **Palo Alto Next Generation Firewalls** (abbreviated: NGFW) both as hardware and as virtualised firewalls, and the NGFWs in each case working with the **operating system PAN-OS** (version 10.1 or later), marketed by Defendant to its customers as a comprehensive network, are falling within the scope of the combined claims 1, 2, 10 and 14 of the patent-in-suit and are - viewed individually - means relating to an essential element of the scope of said combined claims (attacked embodiments).

5. Cortex XDR is the technological foundation for Cortex XSIAM. The SmartScore is a functionality of both Cortex XDR and Cortex XSIAM with no relevant technological difference in its implementation.

6. A NGFW, running the operating system PAN-OS (version 10.1 or later), in a network that

is protected by Cortex XSIAM receives packets and applies security policies to them containing packet filtering rules, actions to be applied to matching traffic and log settings, if configured accordingly. If packets match the criteria of one or more rules of a security policy, the NGFW will take the prescribed action and may, if configured in this way by a user, forward this information to Cortex XSIAM as a so-called ***Threat Log***.

7. Each Threat Log can display the action applied to the relevant packets due to the match with the rule criteria in its field "Action" (cf. Exhibit C12). The NGFW is in that case configured as a sensor to Cortex XSIAM and sends Threat logs to it for automated processing through a so-called data lake (cf. Exhibits C13 and C14).

8. Cortex XSIAM is a cloud-based Security Operations Centre (SOC) platform developed and operated by Defendant, which is accessible over a web-based interface that can be used through web browser software (cf. Exhibit C15). It is designed to integrate and automate security operations by combining various security technologies, including Extended Detection and Response (XDR) and Security Information and Event Management (SIEM).

9. It ingests – *inter alia* – Threat logs it received from NGFW. Cortex XSIAM can analyse and correlate ingested traffic information including Threat Logs collected from NGFWs. Furthermore, Cortex XSIAM can apply detection rules to ingested traffic information. If the criteria of a detection rule are met, Cortex XSIAM can generate an ***Alert*** for that traffic information (cf. Exhibit C19). Cortex XSIAM can combine alerts relating to the same event to so-called ***Incidents*** which refer to – *inter alia* – the potential attack that triggered the creation of NGFW Threat logs. An Incident is the data format in which a human cyberanalysts can investigate a threat and, as the case may be, report it for remedial measures (cf. Exhibit C20).

10. Cortex XSIAM afterwards applies the functionality ***SmartScore*** (cf. Exhibit C17) to the Incidents, providing values between 0 and 100, to automatically assess the risk of the potential attack relating to the respective Incident by applying (first) an algorithm including statistical and context-driven features that have been pre-programmed and (second) a machine-learned algorithm. SmartScore then combines the two results to generate a "unified risk score" to address so-called *alert fatigue* (cf. Exhibit C18). The

scored Incidents are forwarded to an ***Incident Response Surface*** of Cortex XSIAM for further assessment and remedial actions. The listed Incidents can be filtered/sorted by SmartScore on a user's request.

11. For further details relating to the technical characteristic of the attacked embodiments, reference is particularly made to exhibits C10 to C24 and C42 to C45 submitted by Claimant and exhibits HL8, HL9 and HL13 submitted by Defendant.

12. Defendant challenges the validity of the patent-in-suit with its Counterclaim for revocation as originally filed by relying on insufficient disclosure and on lack of novelty in relation to the US patent application US 2015/0213358 A1 (US'358, HLCC10/HAWK). Inventive step is challenged by US'358 (HLCC10/HAWK) in combination with common general knowledge, or either US patent No. 7,418,733 B2 (US'733, HLCC11/CONNARY) or *ArcSight Console User's Guide* (HLCC13), respectively, in combination with common general knowledge or an article by Conor Fellin and Micheal Haney ("*Preventing Mistraining of Anomaly-Based IDSs through Ensemble Systems*", HLCC12/FELLIN).

13. With its reply to the statement of defence regarding the Counterclaim for revocation and its Statement of defence regarding the Application to amend the patent Defendant additionally challenges novelty of the main request in relation to US patent No. US 9,516,053 B1 (US'053, HLCC20/MUDDU) and inventive step in light of HAWK (HLCC10) combined with either the European Patent application No. EP 3 018 879 A1 (EP'879, HLCC17/Palantir), additionally referring to an article by Rahul Saigal (*"How to Track Firewall Activity with the Windows Firewall Log"*/HLCC18), or FELLIN (HLCC12), which is deemed inadmissible by Claimant.

REQUESTS OF THE PARTIES

14. Claimant requests,

I. to order the Defendant to refrain from making, offering, placing on the market, using importing or storing for those purposes, in the territory of the Federal Republic of Germany and the Republic of France,

systems comprising a computing device and an analysis system, the computing device being configured to perform the steps of:

receiving a plurality of packets;

determining, based on threat intelligence data, a plurality of potential threat communications events;

generating, based on the plurality of potential threat communications events, a plurality of event logs; wherein each event log of the plurality of event logs is associated with an event that corresponds to one or more threat detection rules, wherein each event log corresponds to one or more actions that were applied, based on the one or more threat detection rules, to one or more packets;

determining, based on applying at least one algorithm to each event log, a reportability likelihood for each of the plurality of event logs, wherein the reportability likelihood indicates a likelihood that a given event log is associated with an event that is reportable to an authority; and wherein the reportability likelihood is a combined reportability likelihood, and wherein determining, by the computing device, the reportability likelihood for each event log comprises: determining, by the computing device, a first reportability likelihood for each event log based on a static algorithm; determining, by the computing device, a second reportability likelihood for each event log based on a machine-learned algorithm provided by a machine learning system; and determining, by the computing device, the combined reportability likelihood for each event log based on the first reportability likelihood and the second reportability likelihood;

sorting an event queue of the plurality of event logs based on the reportability likelihood of each of the plurality of event logs; and
transmitting, by the computing device and to the analysis system, the plurality of event logs sorted in the event queue;
the analysis system being configured to receive the plurality of event logs.

(Direct Infringement of Claim Combination 14, 1, 2, 10)

II. to order the Defendant to refrain from importing, storing, offering or placing on the market, in the territory of the Federal Republic of Germany and the Republic of France, to any person other than a party entitled to exploit the patented invention, means, namely Next Generation Firewalls with PAN OS (version 10.1 or later), the cybersecurity cloud platform Cortex XDR (version 3.4 or later), and the cybersecurity cloud platform Cortex XSIAM, which are suitable for operating a system as described under I.

(Indirect Infringement of Claim Combination 14, 1, 2, 10)

III. to order the Defendant to communicate to Plaintiff, within thirty days upon service of the notification and, where applicable, translation pursuant to R. 118 (1) s. 1 RoP, the extent to which the Defendant has committed the acts described under I and II. above since 10 December 2022 and render accounts by stating in a uniform, orderly schedule, in writing and in electronic form, by producing supporting documents such as invoices, alternatively delivery notes or receipts, for the extent to which it has committed the acts described under I. and II. above since 10 December 2022, in each case stating

1. the distribution channels of the infringing method and the distribution channels of the products obtained by the use of the infringing method, including the names and addresses of suppliers and other previous owners and the names and addresses of professional buyers;

2. the identity of any third party involved in the use of the infringing method and in the distribution of the products obtained by the use of the infringing method;

3. the quantity of products delivered, received or ordered, the prices paid for the products in question and the points of sale for which the products were intended;

4. the advertising carried out, broken down by advertising medium, its distribution, the distribution period and the distribution area; including the evidence for these advertising activities; and

5. the costs, with individual cost factors and profits realised listed separately,

whereas the relevant documentation, such as orders, order confirmations, invoices and copies of other purchase and sales documents is to be submitted, whereby confidential information outside of the scope of the requested information may be redacted;

IV. to order the Defendant to, within thirty days upon service of the notification and, where applicable, translation pursuant to R. 118 (1) s. 1 RoP,

1. recall the infringing products referred to under I. and II. by informing third parties from whom the infringing products are to be recalled that this Court has found that the products infringe European Patent EP 3 652 914, whereby the Defendant must give these third parties a binding undertaking to reimburse the costs incurred, to bear the packaging and transport costs incurred, to reimburse the customs and storage costs associated with the return of the products and to accept the products again;
2. permanently remove the infringing products referred to under I. from the distribution channels with reference to the fact that this court has found that the products infringe European Patent EP 3 652 914, in particular by requesting third parties who are commercial customers but not end costumers to cancel all orders regarding the infringing products referred to under I. and II.;
3. destroy, at its expense, the infringing products in its possession referred to under I. and II.;
4. to provide the Court and the Plaintiff with written evidence of the measures taken within 30 days of service of the judgement.

V. to establish that the Defendant is liable to compensate Plaintiff for any and all damage which the Plaintiff has suffered and will suffer in the future as a result of the infringing acts referred to under I and II. above committed since 10 December 2022.

VI. to order the Defendant to bear the Plaintiff's reasonable and proportionate legal costs and other expenses in connection with the present proceedings.

VII. to permit the Plaintiff, at Defendant's expense, to announce and publish the decision in whole or in part in public media, in particular on the Internet.

VIII. to order the Defendant to pay to Plaintiff an interim award of damages in the amount of EUR 100,000. This amount is to be amended if infringement continues.

IX. to order that for each violation of any of the orders under I. through IV., the Defendant shall pay a penalty payment in the amount of

- up to EUR 100,000 for each day of violation of the order I.;
- up to EUR 100,000 for each day of violation of the order II.;
- up to EUR 50,000 for each day of violation of the order III.;
- up to EUR 50,000 for each day of violation of the order IV.;

X. to make these orders directly enforceable and permit, in the event that a security is ordered, the Plaintiff to provide such security also in the form of a bank or savings bank guarantee and determine the amount of the security separately for the individual enforceable parts of the judgement.

15. Defendant requests with respect to the infringement action:

- to dismiss the action
- order Claimant to pay the costs

COUNTERCLAIM FOR REVOCATION

16. With regard to their Counterclaim for revocation (UPC_CFI_134/2025), Defendant requests:

revocation of the European patent EP 3 652 914 B1 in its entirety with effect in the Federal Republic of Germany and the Republic of France.

17. Claimant, having filed an Application to amend the patent, requests:

A. As main request,

to **dismiss** the counterclaim for revocation of EP 3 652 914 B1 to the extent of the new main request, i.e. in the combination of claims 1, 2 and 10 as a method claim and in additional combination with claim 14 as a system claim (main request);

B. As a subsidiary request,

to **dismiss** the counterclaim for revocation of EP 3 652 914 B1 to the extent of the auxiliary requests in the form of the claim amendments 1 to 12 as well as permutations of claim amendments 1 to 12, in accordance with the conditional order delineated in the application to amend the patent under section C.

18. Defendant requests to dismiss Claimant's requests to amend the patent.

POINTS AT ISSUE

19. The parties are in dispute about different aspects of the case at hand.

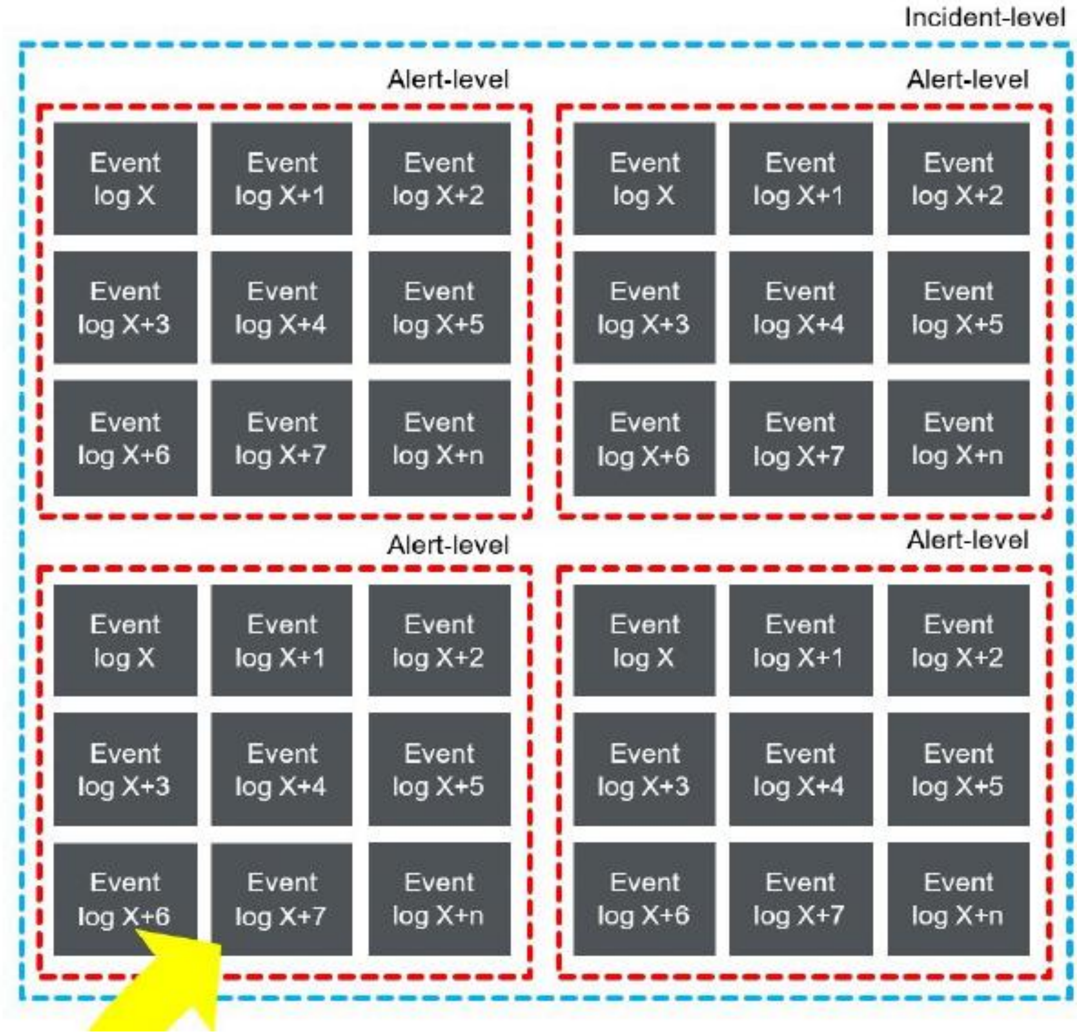
INFRINGEMENT

20. In general, Defendant states, that the different products of the attacked embodiment were not pre-configured in the specific way on which Claimant bases its alleged infringement. For example, the NGFW products, as shipped to customers, were on default not equipped with preinstalled rules that send Threat logs to Cortex XSIAM. They had to be specifically configured for such a purpose.

21. More specifically, Defendant holds that the attacked embodiment would not implement **feature 1.3.2** because it utilised all sorts of events irrespective of whether or not actions were performed and irrespective of whether any applied actions were based on threat detection rules. The term *'each'* defined that the computing device had to be configured in such a way that it could generate a subgroup in which every event log had to meet the criteria of feature 1.3.2. The wording excluded the presence of event logs not fulfilling these additional requirements. In a configuration, in which the generated subgroups contained all types of event logs, feature 1.3.2 was not met.

22. Thus, the attacked embodiments were not configured to operate in accordance with the technical teaching of said feature when shipped to customers. Cortex XSIAM would not be configured to form a subgroup of event logs all fulfilling feature 1.3.2. It would ingest all types of data, and it would stitch and correlate different data together. A specific configuration to exclusively have Cortex XSIAM ingest Threat Logs would be nonsensical from a technical perspective, would need specific modifications and be a rather odd modification.

23. Furthermore, the SmartScore was calculated not for each event log (**feature group 1.4**). Instead, Threat events (and potentially other data points) were aggregated to Alerts and those Alerts were (again) aggregated to so-called Incidents as illustrated in the following depiction presented in the Statement of defence in the infringement proceedings (cf. para 109):



It was this aggregation and correlation logic that helped users of Cortex XSIAM to manage and remediate huge amounts of data, but not the approach claimed by the patent-in-suit. Under the correct claim construction an event log was the output of a monitoring device, for example a firewall. In addition, an event log was also the input for the Cyberanalysis Applications System. Incidents lacked both of these requirements. Incidents were the output of Cortex XSIAM, i.e. the output of the Cyberanalysis Applications System.

24. Regarding **features 1.5 and 1.6**, Defendant stresses that the "Incident Response Interface", of the attacked embodiment, which Claimant maps as the analysis system pursuant to feature 2, would not receive Incidents sorted in accordance with their respective SmartScore. Instead, they were simply stored by their so-called *Incident-ID*. The subsequent possibility to sort Incidents according to their SmartScore would not change this, as this would be done after the completed transmission to the analysis system.

COUNTERCLAIM FOR REVOCATION

25. Defendant bases its Counterclaim for revocation on the following grounds of Art. 138 EPC in conjunction with Art. 65 (2) UPCA:

    - insufficient disclosure (Art. 138 (1) b) EPC in conjunction with Art. 83 EPC),

    - lack of novelty (Art. 138 (1) a) EPC in conjunction with Art. 54(1) and (2) EPC) over US’358 (HLCC10/HAWK) and

    - lack of inventive step (Art. 138 (1) a) in conjunction with Art. 56 EPC) in light of US’358 and common general knowledge, US’733 (HLCC11) and common general knowledge, or FELLIN (HLCC12) or *ArcSight Console User’s Guide* (HLCC13) and common general or knowledge or FELLIN (HLCC12).

26. With its reply to the Statement of defence to the Counterclaim for revocation and its Defence to the Application to amend the patent Defendant bases its request for revocation on the following additional grounds of Art. 138 EPC in conjunction with Art. 65 (2) UPCA:

    - lack of novelty (Art. 138 (1) a) in conjunction with Art. 54(1) and (2) EPC) over US’053 (HLCC20, SPLUNK) and

    - lack of inventive step (Art. 138 (1) a) in conjunction with Art. 56 EPC) in the light of US’358 (HAWK/HLCC10) and EP’879 (HLCC17), with additional reference to the article presented as HLCC18, or FELLIN (HLCC12).

27. For further details on the points at issue, reference is made to the briefs and the

accompanying exhibits.

GROUNDS FOR THE DECISION

28. The Infringement action is admissible but unfounded as the respective admissible Counterclaim for revocation, as originally filed, is founded and the Application to amend the patent-in-suit is unfounded.

A. ADMISSIBILITY

29. Both the Infringement action and the Counterclaim for revocation, as originally filed, are admissible. In particular, neither Claimant nor Defendant did raise any objection against the jurisdiction and local competence of the Local Division Mannheim.

30. It can remain open, whether Defendant´s assessment that the main request of the patent-in-suit lacks novelty over HLCC20 and an inventive step in the light of HAWK with HLCC17 and HLCC18 or HLCC12, which was first put forward in the Reply to the Statement of defence to the Counterclaim for revocation, is admissible pursuant to R. 263 RoP (cf. CoA, order of 02 September 2025, UPC_CoA_807/2025, CeraCon vs. Sunstar; LD Düsseldorf, decision of 08 May 2025, UPC_CFI_11/2024, GRUR-RS 2025, 9427), because the main request of the patent-in-suit does not withstand the novelty attack already asserted in the Counterclaim for revocation solely on the basis of HAWK.

31. Introducing HLCC17, HLCC18 and HLCC20 for the first time in the Defence to the Application to amend the patent is at least admissible insofar as Defendant relies on it in the context of the defence against the claim amendments.

B. SCOPE OF THE PATENT IN SUIT

32. The patent-in-suit relates to network security and in particular to accelerating a cyberanalysis workflow.

33. According to the description of the patent-in-suit, network security is becoming increasingly important as the information age continues to unfold. Network threats may take a variety of forms (e.g., unauthorised requests or data transfers, viruses, malware, large volumes of network traffic to overwhelm network resources, and the

like). Many organisations subscribe to network threat services that periodically provide information associated with network threats, for example, reports that include listings of network-threat indicators (e.g. network addresses, uniform resource identifiers (URIs), and the like), or threat signatures (e.g. malware file identifiers), or threat behaviours (e.g. characteristic patterns of advanced persistent threats). The information provided by such services may be utilised by organisations to identify threats against their networks and associated assets. For example, network devices may monitor network communications and identify any communications between endpoints with network addresses that correspond to threat indicators (cf. para. [0001]).

34. Once identified, these communications events may be logged, and the events logs may be provided to a cyberanalysis system or human cyberanalysts for further investigation into the nature and severity of the threat events and for deciding about potential remedial actions. Typically, the cyberanalysis system or cyberanalysts will determine that only a small portion of these logged threat events will be reportable, in the sense that the events should be reported to the proper authorities who may be responsible for executing the associated remedial actions and for ensuring the security of the network, and who may be responsible for enforcing regulatory compliances or reporting compliance violations (cf. para. [0002]).

35. In many modern enterprise networks, however, the volume and creation rate of network threat event logs often overwhelms the human cyberanalysts' capacities for investigating all of the events. Thus, it is imperative that cyberanalysts' work be assigned efficiently. To that end, the cyberanalysis system or cyberanalysts should investigate only those events that have a high probability of being reportable events and not waste time and effort investigating threat events that are unlikely to be reportable (cf. para. [0002]). If the cyberanalysis workflow cycle proceeds too slowly, the backlog, or queue, of events to be serviced may grow until the queue's maximum size is exceeded, at which time potentially reportable events may be dropped from the queue and never be investigated. This could compromise cybersecurity (cf. [0013]).

36. Various methods and devices for detecting malicious network content are therefore

disclosed in the state of the art, especially in several US patent applications and specifications (cf. paras. [0003] to [0007]).

37. Against this background, the patent-in-suit is based on the objective technical problem to provide an accelerated cyberanalysis workflow to detect reportable threat events in a computer system in a more reliable and efficient manner (cf. paras. [0002], [0015] and [0027]).

38. As a solution, the patent in-suit-provides with the combined claims 1, 2, 10 and 14 a system, the features of which can be structured as follows in accordance with the parties' feature breakdown (cf. Exhibit C9):

Feature		**Origin**
0	A system comprising a computing device and an analysis system,	Claim 14
1	the computing device being configured to perform the steps of:	Claim 14
1.1	receiving a plurality of packets;	Claim 10
1.2	determining, based on threat intelligence data, a plurality of potential threat communications events;	Claim 10
1.3	generating, based on the plurality of potential threat communications events, a plurality of event logs;	Claim 10
1.3.1	wherein each event log of the plurality of event logs is associated with an event that corresponds to one or more threat detection rules,	Claim 1
1.3.2	wherein each event log corresponds to one or more actions that were applied, based on the one or more threat detection rules, to one or more packets;	Claim 1
1.4	determining, based on applying at least one algorithm to each event log, a reportability likelihood for each of the plurality of event logs,	Claim 1

1.4.1	wherein the reportability likelihood indicates a likelihood that a given event log is associated with an event that is reportable to an authority;	Claim 1
1.4.2	and wherein the reportability likelihood is a combined reportability likelihood, and wherein determining, by the computing device, the reportability likelihood for each event log comprises:	Claim 2
1.4.2.1	determining, by the computing device, a first reportability likelihood for each event log based on a static algorithm;	Claim 2
1.4.2.2	determining, by the computing device, a second reportability likelihood for each event log based on a machine-learned algorithm provided by a machine learning system; and	Claim 2
1.4.2.3	determining, by the computing device, the combined reportability likelihood for each event log based on the first reportability likelihood and the second reportability likelihood;	Claim 2
1.5	sorting an event queue of the plurality of event logs based on the reportability likelihood of each of the plurality of event logs; and	Claim 1
1.6	transmitting, by the computing device and to the analysis system, the plurality of event logs sorted in the event queue;	Claim 1
2	the analysis system being configured to receive the plurality of event logs.	Claim 14

39. For the decision at hand some features need further explanation:

Feature 0 (System and Components)

40. The asserted claim combination is a system claim, including two necessary components: a computing device and an analysis system.

41. The computing device does not need to comprise specific components or to have any specific structural properties beyond those which are technically necessary for

implementing the functions according to feature group 1. Its functionalities can be distributed among multiple components in various networks (cf. para. [0071]).

42. The analysis system may or may not be separate from the computing device (cf. para. [0043], l. 4 et seq.; FIG. 2). Its functionality is defined in feature 2 (see below).

Features 1, 1.1 and 1.2 (Potential Threat Communications Events)

43. **Feature 1** is a means-plus-function feature. The computer device must *be configured* in such a way that is able to fulfil the requirements of feature group 1. Whether it is actually used for this purpose or can be used for other purposes that do not fall within the scope of the patent-in-suit is irrelevant (cf. CoA, order dated 14 February 2025, UPC_CoA_382/2024, GRUR 2025, 820 Rn. 47). The same applies for the analysis system according to feature 2 (*'being configured'*).

44. According to **features 1.1 and 1.2** the computing device receives a plurality of (data) packets and determines a plurality of potential threat communication events. Determining the threat communications events is based on *threat intelligence data.*

45. **Feature 1.1** covers any type of (data) packet, including raw data and a collation of raw data in a suitable format, as the teaching of the patent-in-suit is not limited to certain types or protocols of data packets. Any summary of data in a suitable format for further processing according to the features/feature groups 1.3 to 2 falls within the scope of the asserted combination of claims. Its scope is in particular not limited to specific (standardised) data protocols in the field of network technology. Conversely, from a functional point of view, it makes no difference how the (data) packets are assembled and whether they use a certain (standardised) protocol, as long as they can be fed into the subsequent processing steps in a technically meaningful manner which allows to analyse, whether or not they may be a potential threat. Accordingly, the patent description does not set out any functional reason, which would mandate that the incoming data is to be organized in a specific packeting format.

46. As an exemplary embodiment, the patent specification describes a TCP/IP communications monitoring device (cf. para. [0009]), called a Cyber Threat Intelligence (CTI) gateway, configured to detect and log communications that match threat

indicators, signatures, behavioural patterns, and the like (cf. para: [0025])*,* i.e. *threat intelligence data* (**feature 1.2**).

47. Such packets, matching threat intelligence data, are considered as potential threat communication events in the meaning of **feature 1.2**. The source and type of suitable threat intelligence data, i.e. data that indicates a potential threat event, and its implementation in a way that the computing device has at least access to said data is left to the discretion of a person skilled in the art.

Feature Group 1.3 (Event Logs)

48. **Feature group 1.3** relates to the creation of event logs associated with potential threat communication events.

49. The term ‘event log’ is not defined in the patent-in-suit. In accordance with the context of the asserted claim combination, an event log, which is a log that is based on potential threat communications events (**feature 1.3**), must at least meet two additional criteria:

50. First, it must be associated with an event that corresponds to one or more threat detection rules (**feature 1.3.1**). Second, it must correspond to one or more actions that were applied, based on the one or more threat detection rules, to one or more packets (**feature 1.3.2**). Examples for such actions applied to a data packet based on threat detection rules, to which the asserted claim combination is not limited to, are mentioned in the description of the patent-in-suit (cf. para. [0009], col. 3, l. 10 et. seqq.).

51. The further technical implementation of an event log is left to the discretion of a person skilled in the art. Contrary to Defendant’s assertion, an event log is not necessarily the output of a monitoring device, for example a firewall, and the input of a cyber analysis system. The subject matter of the patent-in-suit is neither limited to the exemplary embodiment shown in FIG. 1 nor to a Security Information and Event Management (SIEM) system known in the state of the art. Rather, every data packet log which meets the requirements of feature group 1.3 is an event log within the meaning of the patent-in-suit, regardless of which functional unit of the computing device it originates from or is aggregated by.

52. Furthermore, contrary to Claimant's assessment, feature group 1.3 does not restrict the content of the event log to a certain type of data, particularly it does not exclude collecting *'raw data'* information or communication of a potential threat event. It may relate to any information and kind of data in a format suitable for logging.

53. Moreover, unlike Claimant's assertion, feature 1.3.2 neither requires an (explicit) entry of the applied action based on a threat detection rule in the event log nor the possibility to *'somehow'* derive the applied action from the content of the event log.

54. Feature 1.3.2 only specifies that the event log as such *corresponds* to an action applied based on a threat detection rule. Accordingly, it is sufficient, if the event log is actually related to the application of a threat detection rule, without the application of the rule, i.e. the applied action, has to be explicitly apparent or *"somehow"* derivable from the event log itself. This requirement is, for example, met if the respective event log is created only because of the applied action.

55. This construction of the wording of feature 1.3.2 is in line with the description of the patent-in-suit: According to para. [0010] the applied action is one of various different aspects/information that could optionally be included in an event log *(The log may also include context information, such as […], any actions applied to the packet by the monitoring device)*. The asserted claim combination is not restricted to this exemplary embodiment. As a result, any information that originates from a potential threat communication event and is recorded (aggregated) into a log, which is subject to further assessment, i.e. a task for cyberanalysis (cf. para. [0044], l. 28 *new task (event log)"*), fulfils the requirements of an event log as claimed.

56. Further, the sole *log* of a (data) packet based on a respective is a suitable action based on a respective rule. **Feature 1.3.1** does not restrict the potential actions to a specific type. This is consistent with the technical teaching of the patent-in-suit. Functionally, it does not matter on which type of actions the potentially overwhelming amount of threat event data, which has to be reviewed during cyberanalysis, is based on.

57. In accordance with the wording of feature 1.3.1 and the functional aspects mentioned above, the description of the patent-in-suit explicitly states exemplary actions which

can be applied to a packet including – *inter alia* – logging *(log the packet*, cf. para. [0009]). This *'packet log'* is subsequently *'aggregated into'* an *'event log'* for cyberanalysis purposes (cf. [0010]). Accordingly, the event log is in general a format in which useful data relating to a potential threat event is gathered for further processing.

58. This construction of feature 1.3.1 is also in line with the more detailed description of an exemplary embodiment relating to FIG. 1 (cf. para. [0032]). In this embodiment, a collection component or Cyber Threat Intelligence (CTI)-Gateway 120 *"will detect the HTTP request and may log the resultant HTTP session as a threat event. Collection Component or CTI-Gateway 120 send threat event logs to Cyberanalysis Application System 130"*. Also in this context, no other applied action to the detected potential threat event, i.e. the HTTP request, is described except its log. The logged event is afterwards transferred to a cyberanalysis application system as an event log for further assessment (cf. para. [0035], l.36 et. seq.).

59. Thus, in summary, any data packet that is logged (i.e. an action is performed on it) based on a respective threat detection rule and further processed as an 'event log', in the sense that the logged packet is subject to further proceedings in accordance with feature groups 1.4 to 2, falls within the scope of the asserted claim combination.

60. The Panel does not need to decide on the other questions of claim construction discussed between the parties in relation to feature group 1.3, as they are not decisive for the present decision. This applies in particular to the question of whether the term *'each'* precludes the computing device from also creating logs that do not meet the requirements of feature 1.3.2 and whether different threat events may be bundled in another (higher level) data object that is the sole subject of further processing, in particular according to feature group 1.4.

Features 1.4 and 1.4.1 (Reportability Likelihood)

61. **Features 1.4 and 1.4.1** relate to determining a reportability likelihood for each event log generated according to feature group 1.3.

62. The term *'reportability likelihood'* refers to the event logs generated in accordance with feature group 1.3 and is an indicator of the likelihood that the associated potential

threat event is reportable to an authority, i.e. ‘a real’ threat event. Provided that the further requirements of feature group 1.4. are met, the technical implementation of said indicator is left to the discretion of a person skilled in the art. This applies in particular for the determination of a specific score or value in relation to a reference score or value in order to assess the reportability likelihood. Furthermore, feature group 1.4 does not preclude taking other aspects into account alongside the values or scores determined in accordance with its requirements when establishing the final reportability likelihood.

63. Rather, the reportability likelihood must at least be *based on* applying at least – against the, in regard of the asserted claim combination, obviously incorrect wording (one) – *two* algorithms. The algorithms for determining the reportability likelihood are the subject matter of feature group 1.4.2

Feature Group 1.4.2 (Algorithms and Combined Likelihood)

64. According to **feature group 1.4.2**, the reportability likelihood is a combined likelihood. The determination of said combined likelihood is claimed by features 1.4.2.1 to 1.4.2.3.

Static Algorithm

65. The computing device determines a first reportability likelihood for each event log based on a *static* algorithm **(feature 1.4.2.1)**. A *static* algorithm is any algorithm that does not change its assumptions or classification automatically, for example via training. This claim construction corresponds to the description of the patent-in-suit.

66. In an exemplary embodiment (cf. dependent claim 8 of the patent-in-suit), a static algorithm *may* be a human-designed (H/D) algorithm, which *may* be explicitly programmed. It *may* use conditional logic and/or Boolean logic applied to the characteristics of event logs (cf. para. [0045], col. 16, l. 56. et. seq.) Such an algorithm does not automatically learn (cf. para. [0040], col. 14, l. 38 et. seq.).

67. As can be seen from the exemplary embodiment according to FIG. 4, the decision table of a static algorithm may represent a human designed fix computer program statement, for example, *"IF ((Indicator_Type == FQDN) AND ((0 <= Indicator_Age < 30) OR*

(CTI_Score = High))) THEN Reportability Likelihood := 0.7" (cf. para. [0045], col. 17, l. 1 et. seq.). In other words, if the conditions of the exemplary computer program statement are met, the static algorithm will always determine a reportability likelihood of 0.7 and it will not change this assumption over time.

68. Contrary to Claimant's arguments, the asserted claim combination is not limited to this embodiment. A static algorithm in the general sense, that it does not change its assumptions automatically via training, does not necessarily have to be consistently human-understandable and explicitly programmed nor to be heuristic. These additional requirements are only part of the description (cf. paras. [0017] and [0040]) and (partially) claimed with the dependent claim 8 of the patent-in-suit but no reflected in feature 1.4.2.1.

69. In summary, the technical difference between a static algorithm according to **feature 1.4.2.1** and a machine-learned algorithm (M/L algorithm) pursuant to **feature 1.4.2.2** is that the latter is capable of changing its assumptions automatically, e.g. if a certain characteristic of an event log pertains to a potential threat or not, through training.

Machine-learned algorithm

70. Apart from that, **feature 1.4.2.2** does not contain any further mandatory requirements regarding the design of the machine-learned algorithm to determine a second reportability likelihood. Its technical implementation is left to the discretion of a person skilled in the art. An M/L algorithm *may* combine event characteristic by correlating event characteristics with previously identified threats (cf. para. [0017], col. 6, l. 17 et. seq.). It *may* be *any* supervised learning algorithm, such as – but not limited to – artificial neural networks, genetic programming, and the like (cf. para [0049]).

71. In accordance with common general knowledge the description of the patent-in-suit states that training data for supervised learning algorithms is composed of labelled training examples (cf. para. [0049], col. 18, l. 13 et. seqq.), which means that the training data is already categorized correctly (e.g. 'threat' or 'no threat'). Thus, embodiments that only include human selected labelled training data and may be heuristic in the sense, that the machine-learned algorithm only refers to its training examples, i.e. its

experience, to determine whether an (unknown) event is a threat event, are not excluded from the scope of feature 1.4.2.2.

72. Additionally, the M/L algorithm *may* use many more characteristics and *may* combine those characteristics in more complex ways, e.g., by computing nonlinear multi-variable functions, that *may* not be readily designed or explicitly programmed or well-understood by human (cf. para. [0017], col. 6 l. 20 et. seq.). Such well-trained M/L algorithms will be able to recognize complex patterns in event characteristics. In effect, an M/L algorithm – according to an exemplary embodiment – has learned how to emulate the human reasoning, much of which is implicit knowledge that is difficult for humans to explicitly program as a computer logic (cf. para. [0040], col. 14, l. 41 et. seq.).

73. Against Claimant's assessment, the subject matter of feature 1.4.2.2 is not limited to these exemplary embodiments, as they are not reflected in the asserted claim combination. As Defendant correctly assumes, **feature 1.4.2.2** does not specify the type of the machine-learned algorithm and does contain any specifications regarding training data or specific problems it has to solve in comparison to a static algorithm. Therefore, the use of any machine-learned algorithm capable of changing its assumptions, i.e. learning automatically, when trained and of providing a score or value that indicates the reportability likelihood of the event log in question, falls within the scope of the patent-in-suit.

74. Contrary to Claimant's opinion, feature 1.4.2.2 in particular does not require that the machine learning system learns to take into account the '*human bias*' or to develop – as Claimant suggested during the oral hearing – a cyberanalyst's *'gut feeling'* when assessing an event in a computer or network system.

75. Nothing to the contrary can be inferred from the terms '*provided by a machine-learning system'* in feature 1.4.2.2 on which Claimant has most recently focused.

76. As mentioned above, the machine-learning generation engine may be *any* supervised learning algorithm (cf. para. [0049]). There are no further indications that the machine learning system as such is, in the words of Claimant, *'a guarantee for achieving benefits'* like the reflecting of the *'human bias'*. In this context as well, feature 1.4.2.2 cannot be

constricted to the exemplary embodiment presented in para. [0040], line 42 et. seqq., which – if at all – could indicate such advantages of a M/L algorithm.

77. Rather, a *machine learning system* is any computer system that, when trained with (labelled) training data, is capable of providing a specific algorithm which can be applied on incoming event logs in a computer or network system. It does not have to be a functional unit that can be strictly distinguished from a basic algorithm (the algorithm to be trained), which may be programmed by a human. Rather, the specific algorithm applied after training on (new) incoming event logs is provided by a machine learning system, even if it is based on a human programmed algorithm, because it has to be fed with training data to come to technical useful assumptions.

Combined Likelihood

78. Finally, according to **feature. 1.4.2.3**, the computing device determines a combined reportability likelihood for each event log based on the first reportability likelihood and the second reportability likelihood.

79. The method used to combine the likelihoods is left to the discretion of a person skilled in the art. No restrictions are established in feature 1.4.2.3 as long as the combined likelihood reflects a likelihood that the respective event log is associated to a reportable event or not (feature 1.4.2.1). Contrary to Claimant, operations like summing or multiplying are possible, as well as any other combination that takes both likelihoods into account. This also includes selecting only the higher value after assessing both (cf. para. [0041] l. 23 et. seqq.).

80. In exemplary embodiments, to which the asserted claim combination is not limited to, the combination of a static and a machine-learned algorithm *may* make the overall system more robust to irregular, idiosyncratic, or anomalous threats, which *may* not correlate well with patterns of previous reportable events used as training data for a machine-learned algorithm (cf. para. [0021]).

Features 1.5 to 2 (Event Queue and Analysis System)

81. An event queue in the meaning of **feature 1.5** is a set of event logs ordered based on

the reportability likelihood. Events with a low reportability likelihood may never be investigated by cyberanalysts. The service queue gives a cyberanalyst information on the reportability likelihood and can reduce the average workflow cycle time for queued events, resulting in workflow acceleration (cf. para. [0015], l. 21 et seqq.).

82. According to its wording (‘based on’), feature 1.5 does not require mono-causality, i.e. that the event queue is sorted exclusively in the order of the event logs’ reportability likelihoods as determined according to feature group 1.4. Rather, it is sufficient that the reportability likelihood is a factor that plays into the concrete ranking within the queue. In this regard, it is also noted that claim 12 of the patent indicates that the sorting may be performed based on other characteristics.

83. Pursuant to **feature 1.6**, the computing device transmits the event logs sorted in the event queue to an analysis system. The question whether the analysis system must receive the event logs in sorted order is irrelevant to the present decision and can, accordingly, remain open.

84. The purpose of an analysis system according to **feature 2** is to review information on potential threat events, conduct investigation into such events, determine an event’s type and severity, determine mitigating or remedial actions, or report events to management and/or network security operations (cf. para. [0011], l. 50 et seqq.). The further technical implementation of an analysis system is left to the discretion of the person skilled in art.

85. It may be external to the computing device but does not need to be. As an exemplary embodiment, the patent-in-suit describes the analysis system being configured to access a work queue of logs via a user interface designed to assist event analysis (cf. para. [0011], l. 43 et seqq.). The analysis system may include functions executed by cyberanalysts using threat event analysis applications such as SIEM to investigate the events and determine if the events should be reported to authorities (cf. para. [0032]).

C. COUNTERCLAIM FOR REVOCATION

86. The admissible (see supra) Counterclaim for revocation, being directed against the German and French part of the patent-in-suit, is founded. Claimants’ main request lacks

novelty over US’358.

Lack of novelty over US 2015/0213358 A1 (US’358, HLCC10/HAWK)

87. The subject-matter of the asserted claim combination of the main request is not new with respect to US’358.

Subject of ’358

88. US’358 relates to apparatus and methods facilitate analysis of events associated with network and computer systems. In particular, it relates to apparatus and methods by which to identify event occurrences in networks or computer systems, such as intrusion attempts, that are significant, and score the identified event occurrences with quantitative scores (cf. abstract and para. [0003]).

89. According to the description of US’358, as known in the state of art, during system operation, many varied system events occur, both events internal to the system as well as external events that potentially affect and threaten operation of the system. A manager or operator of the system, in order fully to be aware of the system operation, should be aware of system-related events, particularly events that might deleteriously affect operation of the system (cf. para. [0005]).

90. Logging of the occurrences of such events, when detected, permits subsequent review of the events by the system manager or operator enabling them to take responsive action. However, because of the potentially large number of event occurrences, the log of the event occurrences is potentially very lengthy, and review of the logged event occurrences might well be time-consuming. Furthermore, if a small number of significant event occurrences are interspersed amid a large number of insignificant event occurrences, a reviewer might not properly notice significant event-occurrence entries in the log (cf. para. [0006]). Accordingly, an improved manner by which to provide for review and analysis of system-event occurrences would be beneficial (cf. para. [0007]).

91. Against this background, the technical problem according to US’358 is to provide improved system operation and management, particularly for review and analysis of

system events across multiple monitored systems or networks (cf. para. [0008]).

92. As a solution, US’358 suggests an exemplary apparatus for collecting data across multiple networks according to FIG. 1 below:

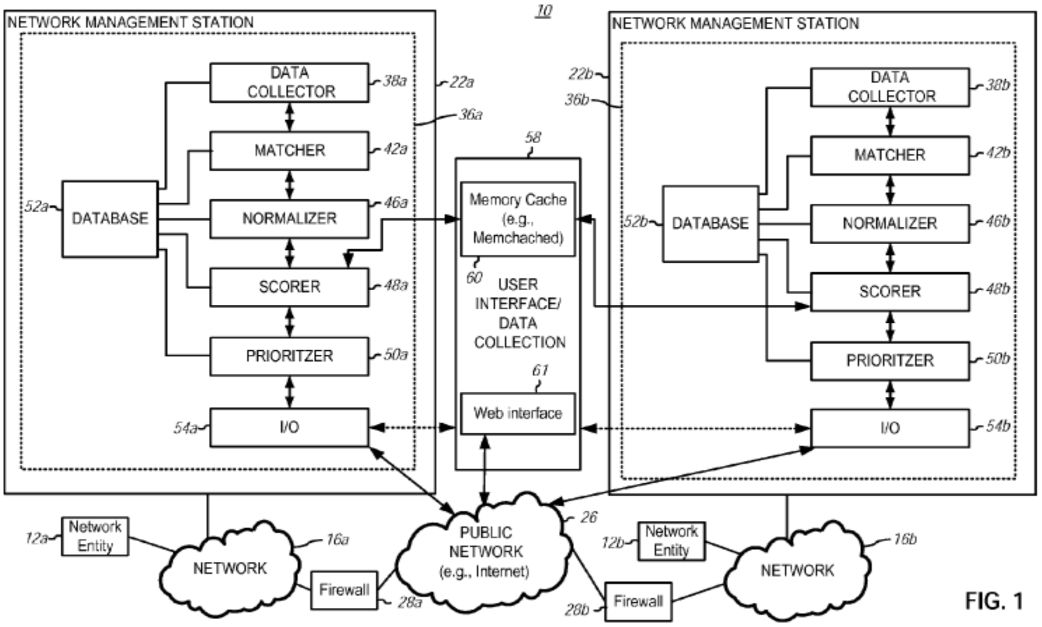


FIG. 1

93. The exemplary system or systems, shown generally in FIG. 1 at 10, include(s) one or more entities 12a, 12b, such as work stations or servers, by the way of example, that are communicatively coupled to respective networks 16a, 16b. In the example of FIG. 1, each network 16 may also be communicatively coupled to a management station 22a or 22b configured to monitor and store event-occurrences that may occur in the networks 16a, 16b (cf. para. [0033]).

94. The network management stations 22 each include an exemplary apparatus 36a, 36b. The apparatuses 36a, 36b facilitate analysis of operation of the network systems by collecting information related to the occurrence of events at the network system in a manner that provides a manager or operator of the network system with indications of events that are considered to be significant, thereby permitting the manager or operator to more quickly to take responsive action (cf. para. [0036]). Each apparatus 36 includes a data collector 38, a matcher 42, a normalizer 46, a scorer 48, a prioritizer 50, a database 52 and an input/output (I/O) 54. The I/O may be configured to generate and transmit information to be used and displayed at a common user interface 58 (cf. para.

[0037]).

95. The data collectors 38 operate as event collection engines utilizing, for example, a SYSLOG or SNMP, or other analogous collection algorithm. Data collected by data collectors 38 pertain to events occurring within, or related to, the network system (e.g., the associated network 16 and various network entities 12). The collected event occurrence information, comprising raw data, is stored in the databases 52a or 52b, and thus available for subsequent retrieval (cf. para. [0038]).

96. The collected data stored in databases 52, or immediately collected by collector 38, are accessible by matchers 42a, 42b. Matchers 42 operate to match the collected data with predetermined conditions or 'event rules', thereby forming event occurrence items. The event rules are user defined rules or are otherwise defined. Matchers 42 match selected ones of the event rules with the event occurrence items that have been collected by the data collector. If the event occurrence item corresponds to an event rule, then the event occurrence item is considered to be a potentially significant event occurrence (cf. para. 0039]).

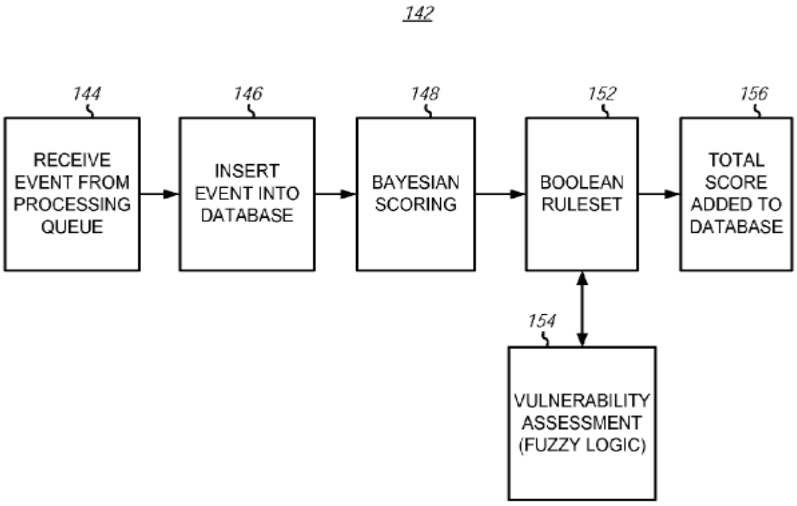
97. Event occurrence items that match the event rules may then be normalized by normalizers 46. Normalizers 46 operate to extract, or otherwise identify the significant portions of the matched, event item occurrences. The normalizers 46 are also capable of accessing the databases 52 to be provided with the event occurrence items matched by the matcher. In addition, normalized event occurrence items are also cached, or stored, at the database 52 (cf. para. [0039]).

98. The scorers 48, capable of accessing the database 52, operate to score (i.e. provide a quantitative value) the normalized, matched event occurrence items. Scoring is performed by comparing the normalized, event occurrence item with score event rules. If the event occurrence item corresponds to the score event rule, then a match is made, and a score associated event occurrence item is incremented. The score associates a score with the event occurrence item, and the score associated with the event occurrence item is stored at the database, indexed together with the associated event occurrence item (cf. para. [0040]).

99. Prioritizers 50 access the databases 52 to obtain, or to be directly provided with, the scores associated with the event occurrence items. Prioritizers 50 prioritize the event item occurrences based upon the score assigned to the individual ones of the event occurrence items by scorers 48. The prioritizers 50 prioritize the event occurrence items, e.g. in numerical order, from highest score to lowest score, to prioritize the event occurrence items from potentially most significant, based upon the score, to least significant, based upon the score. The prioritized item entries are used, for example, to form a list of event occurrence items. Priority indications are also associated with the event occurrence items, stored at the databases 52 or immediately provided to the I/O 54 (cf. para. [0041]).

100. The I/O 54 (in the description of US'358 under para. [0042] obviously wrong referred to as "56", which does not exist in FIG. 1) may be configured to generate information pertaining to a prioritized listing, as well as information relating to the prioritized event occurrence items. This information is provided to or queried by the user interface 58 to permit a network system manager or operator to view the event occurrence items considered to be of greatest significance prior to lesser-prioritized event occurrence items (cf. para. [0042]).

101. Regarding the process of scoring of event occurrence items by scores 48 that have been matched with an event rule, the following FIG. 5 of US'358 illustrates an exemplary operation 142 in a more detailed manner (cf. para. [0067]):


142
144
RECEIVE EVENT FROM PROCESSING QUEUE
146
INSERT EVENT INTO DATABASE
148
BAYESIAN SCORING
152
BOOLEAN RULESET
156
TOTAL SCORE ADDED TO DATABASE
154
VULNERABILITY ASSESSMENT (FUZZY LOGIC)

102. In general, first, and as indicated by the block 144, the event occurrence item is received from a processing queue. Then, and as indicated by the block 146, the event occurrence item is inserted into a database. In addition, as indicated by the block 148, Bayesian scoring is performed. Then, and as indicated by the block 152, Boolean rules of a score rule set are compared with the event occurrence item (cf. para. [0068]. Finally, as indicated by the block 156, a summed score, i.e. a cumulated or cumulative score, of matches of score rules of a score rule set and the event occurrence item is obtained (cf. para. [0070]).

103. The score value is determined from a variable length rule set that determines a successful or unsuccessful match against the unique rules of the rule set. Each rule of the score rule set can have multiple arguments and, in the exemplary implementation, implemented as a Boolean rule, as a positive or negative value. The score, responsive to a successful match of a specific score rule, is summed together with other score-rule results in order to arrive at a final, overall score (cf. para. [0071]). A rule set is any list of rules that is associated with a positive, or negative, score. When a rule set matches against a provided event, the associated score is added to the existing score. The initial score is zero (cf. para. [0074]).

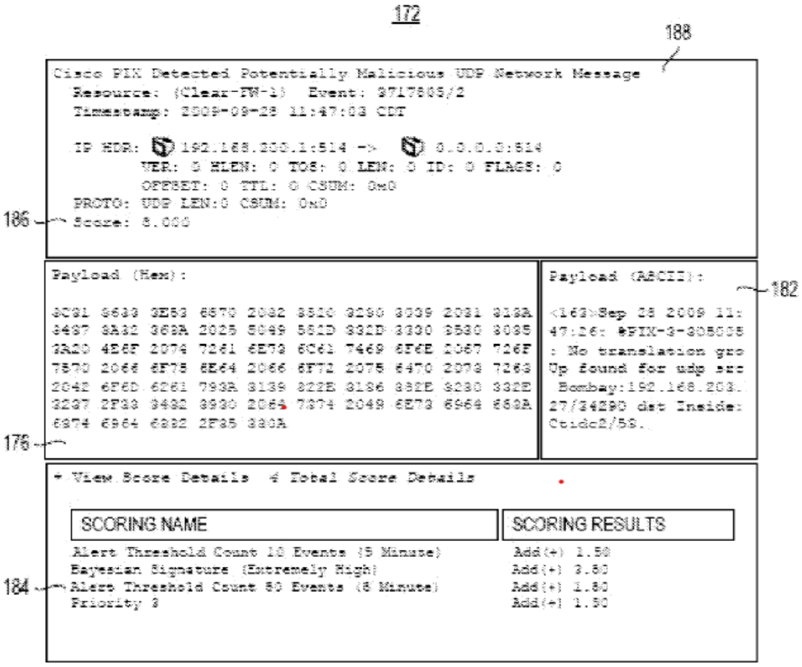
104. Exemplary actions that define score rules that are matched against an event occurrence item include an alert name (regular expressions), an alert category (regular expressions), audit actions, *Bayesian score*, count (by host/alert), a count (by host/category), a host, either destination or source, an event payload, a port, either destination or source, an alert priority, a resource, and a timestamp. These are exemplary rules of a score rule set that are utilized in various implementations to define matches that are scored (cf. para. [0075]).

105. In an exemplary implementation according to FIG. 5, before an event is compared against rules of a score rule set, a Naïve Bayesian score is determined. The Bayesian score is included with existing event properties that are processed by the score rule sets (cf. paras. [0073] and [0074].

106. When a Bayesian score is utilized, a Bayesian histogram analysis algorithm is utilized to uniquely fingerprint known security and performance issues, while establishing a

base line for positive or neutrally-acceptable network traffic, utilizing standard deviation. The algorithm identifies unique attributes within a specified target event. The activity in conjunction with standard deviation facilitates pattern matching. Thereby, both known or trained information is matched and, through use of standard deviation, target events that have not been trained or identified are also matched. An operating baseline is thereby established (cf. para. [0076]).

107. The following FIG. 6 illustrates an exemplary screen display 172 generated during operation of an embodiment of the process of US'358 and provided to a user for further assessment.



```
Cisco PIX Detected Potentially Malicious UDP Network Message
  Resource: (Clear-FW-1)  Event: 3717805/2
  Timestamp: 2009-09-28 11:47:03 CDT

  IP HDR: 192.168.203.1:514 ->  0.0.0.0:514
         VER: 0 HLEN: 0 TOS: 0 LEN: 0 ID: 0 FLAGS: 0
         OFFSET: 0 TTL: 0 CSUM: 0x0
  PROTO: UDP LEN:0 CSUM: 0x0
  Score: 8.000
```

```
Payload (Hex):

3C31 3633 3E53 6570 2032 3820 3230 3039 2031 313A
3437 3A32 363A 2025 5049 582D 332D 3330 3530 3035
3A20 4E6F 2074 7261 6E73 6C61 7469 6F6E 2067 726F
7570 2066 6F75 6E64 2066 6F72 2075 6470 2073 7263
2042 6F6D 6261 793A 3139 322E 3136 382E 3230 332E
3237 2F33 3432 3930 2064 7374 2049 6E73 6964 653A
4374 6964 6332 2F35 330A
```

```
Payload (ASCII):

<163>Sep 28 2009 11:
47:26: %PIX-3-305005
: No translation gro
Up found for udp src
 Bombay:192.168.203.
27/34290 dst Inside:
Ctidc2/53.
```

```
+ View Score Details  4 Total Score Details

 SCORING NAME                                  SCORING RESULTS
 Alert Threshold Count 10 Events (5 Minute)     Add(+) 1.50
 Bayesian Signature (Extremely High)            Add(+) 3.60
 Alert Threshold Count 80 Events (8 Minute)     Add(+) 1.80
 Priority 3                                     Add(+) 1.50
```

108. The exemplary screen display is related to a single event occurrence item (herein: *Detected Potentially Malicious UDP Network Message*). The raw data comprising the message forming the event occurrence item is displayed in hexadecimal form at the portion 176 and in ASCII (American Standard Code for Information Interchange) form at the portion 182. Scoring results are indicated at the portion 184, a total score is indicated at 186, and other descriptive information is displayed in the portion 188. The display 172 is displayed at a display screen of the user interface, which provides a role-based access control for administration over secure encrypted session (cf. para. [0077]).

Lack of Novelty

109. US’358 discloses all features of the main request directly and unambiguously.

110. It describes a system and a method comprising **features 0, 1.5, 1.6 and 2** which is not disputed by Claimant and also not the result of an erroneous legal assessment.

Features 1.1 (Packets)

111. US’358 discloses **feature 1.1**. directly and unambiguously.

112. As mentioned in the context of claim construction, **feature 1.1** does not restrict (data) packets to a certain type or protocol. Any collection of data in a suitable format for further processing received by the computing device is a (data) packet in the meaning of feature 1.1. Therefore, collected ‘raw data’, which may be the subject of further processing according to US’358, is not excluded from the scope of the patent-in-suit. Regardless of this, the disclosure of the general method of US’358 is not restricted to ‘raw data’, but further mentions ‘system data’ in general and data ‘packets’ in a network (cf. paras. [0010], [0034] and [0064] ‘*network package information’*).

Features 1.2 (Threat Intelligence Data)

113. Further, US’358 discloses **feature 1.2** directly and unambiguously. The matchers 42 determine, based on threat intelligence data, a plurality of potential threat communications events relating to the collected data packets.

114. Contrary to Claimant’s assessment, US’358 refers – *inter alia* already in its introduction – explicitly and primarily to threat event occurrences in a network or computer system such as *security threats* and *intrusion attempts*, which might be *deleterious* for the system. Consequently, a person skilled in the art readily understands that the ‘event rules’ on the basis of which the matchers 48 identify a potentially significant events primarily refer to a data record, defined by a user or otherwise (cf. para. [0039]), suggesting such a threat event, i.e. *threat intelligence data* in the meaning of the patent-in-suit.

Feature Group 1.3 (Event Logs)

115. Moreover, US’358 discloses **feature group 1.3**.

116. The normalized event occurrence items, which are subject of the following steps of scoring and prioritizing, are *event logs* in the meaning of the patent-in-suit (**feature 1.3**).

117. After matching, i.e. identifying potentially threatening events, the matched data can be stored in the database 52. The normalizers 46 are capable of accessing the databases 52 to be provided with the matched event occurrence items. Therefore, normalized event occurrence items are generated because of the detected potential threat events (**feature 1.3**) and are generated (only) because of the application of the event rules by the matchers 42, i.e. correspond to the application of said rules and, thus, are associated with the potential threat events (**feature 1.3.1**). According to para. [0064] of US’358, the matching rules can contain an alert name, a category, a knowledge base identification, host and network packet information. This information is associated with event occurrence items and provides additional information to assess their threat level.

118. The matched and normalized event occurrence items further *correspond to an action* that has been applied *based on* said application of event rules to the collected (data) packets (**feature 1.3.2**).

119. As already explained, the applied action neither has not to be written explicitly into the event log nor has ‘somehow’ to be derivable. Rather, the wording of the claim (‘corresponds’) is significantly broader. Accordingly, it is sufficient, if the event log is actually related to the application of such a threat detection rule. This applies to *normalized event occurrence items* as disclosed in US’358 that are created (only) because of the previous (positive) comparison of collected data with the event rules by the matchers 42.

120. Furthermore, it is sufficient that the *applied action* on the data packet is the *log* itself because of a match with (at least) one threat detection rule. Logging according to the patent-in-suit means, see supra, to create a record of the piece of data and to

put this information into a data format suitable for further processing, especially scoring in accordance with feature group 1.4.

121. This requirement is met by normalized event occurrence items, as the normalizers 46 access the stored raw data packets originating from the matchers 42 (cf. para [0039) and corresponding FIG 1, 42a and 42b), which logged them respectively, i.e. performed an action on them, and generate a data format containing information about the potential threat event, which is the subject matter of further steps of scoring and prioritizing.

122. Contrary to claimant's assertion, the log of matched occurrence items is obviously based on a corresponding rule as it occurs when the matchers arrive at a ‘positive’ result because it matches the ‘event rules’.

Feature Group 1.4 (Machine Learned Algorithm and Combined Likelihood)

123. Finally, **feature group 1.4** is also disclosed in US’358.

Static Algorithm

124. As Claimant does not dispute, the disclosed use of a pre-defined Boolean rule set (cf. para. [0071]), which determines a score by matching the (matched and normalized) event occurrence item log against the rules of a score rule set, providing a positive or a negative value for each match, meets the required application of a static algorithm for determining a first reportability likelihood (**feature 1.4.2.1**).

Machine Learned Algorithm

125. Contrary to Claimant’s assessment, US’358 also discloses directly and unambiguously to determine a second reportability likelihood for said normalized, matched occurrence items based on a machine-learned algorithm provided by a machine learning system **(feature 1.4.2.2)**.

126. As Defendant correctly assumes, **feature 1.4.2.2** does not specify the type of machine-learned algorithm. Therefore, as explained in the context of claim construction, the use of any machine-learned algorithm capable of providing a value

that indicates the reportability likelihood of the event log in question falls within the scope of the patent-in-suit. The machine-learned algorithm according to feature 1.4.2.2 is especially not limited to embodiments which are – citing Claimant – *‘trained to determine such an unknown entity as the experience (expertise) of the cyberanalyst’* or *‘which includes the human bias of detecting dynamic relationship between features and can therefore anticipate new combinations of features*’.

127. Against this background, a Naïve Bayes algorithm as disclosed in US’358 is a (well-known) supervised machine-learning algorithm, which Claimant no longer disputes as such. It just stated in its written submissions, not any *‘arbitrary’* machine-learning algorithm fall within the scope of the patent-in-suit. This assumption is correct insofar as the machine learning algorithm has to be capable of providing a value that indicates the reportability likelihood of an event log. This is the case, however, according to Claimant’s own description of how a Naïve Bayes algorithm works. Accordingly, US’358 discloses a method for determining a *Bayesian score* using a trained probability prediction based on the training data.

128. The circumstance that a Naïve Bayes algorithm uses pre-defined parameters, statistics and training data provided by human and considers each feature of an event independently does not refute this assumption. As directly disclosed in US’358 (cf. para. [0076]), the Naïve Bayes algorithm is learning characteristics or attributes of information fed to the system through training as ‘good’ or ‘bad’ (cf. paras. [0076] and [0081]). Accordingly, even Claimant admits, a pre-defined feature might shift *via learning* from one category to another (e.g. from ‘associated with a potential threat event’ [Bad] to ‘not associated with a potential threat event’ [Good]). The latter would not be the case for a static algorithm comprising a strictly predefined conditional logic related to the threat-level of an event.

129. As the Naïve Bayes algorithm uses its experience derived from labelled training data to determine a probability-based value, i.e. *Bayesian score*, to assess whether an event occurrence item in a network is dangerous or not, it is also a machine-learned algorithm provided *by a machine learning system*. This is because, depending on the training data, the algorithm is able to automatically change its assessments –

contrary to a static algorithm – regarding the same event. This capability, i.e. the specific probability distribution, is implemented via training the computer system and is, accordingly, provided by a machine learning system.

130. No further requirements can be derived from feature 1.4.2.2.

131. That any expansion or modification of pre-defined features itself would – according to Claimant – require a human to define it does not change the fact, that a Naïve Bayes algorithm learns, when based on training data, to weigh certain features, which could indicate a potential threat event in a different way, and is therefore a machine learned algorithm. Moreover, when a pre-defined feature shifts via learning from one category to another, it is a *new* feature for the latter category.

Combined Reportability Likelihood

132. Finally, US'358 also discloses feature **1.4.2.3**. Against Claimant's assertion, the combination of the two gained reportability likelihoods does not need to go beyond a mere summation Such a restriction has no basis in the patent-in-suit.

133. Rather, according to the patent-in-suit, the combination could be greater than or equal to both reportability likelihoods (para. [0041]). A summed score, as disclosed in para. [0070] and FIG. 5 and FIG. 6 of US'358, meets this requirement, as it is obvious to a skilled person that such summed score can be put into the context of a maximum score derivable from the applied algorithm and optionally use other factors. Accordingly, US'358 relates to prioritizing the matched event occurrence items.

134. Moreover, US'358 does not just disclose the mere summation of both indicators, but also the evaluation of the score determined by the Naïve Bayes algorithm *(Bayesian score*, cf. para. [0075]) based on a further rule according to the Boolean algorithm (cf. para. [0080] et. seq., correlation Key: *bayes_weight* according to TABLE 1). In other words, the determined *Bayesian score* is considered as a factor when event occurrence items are scored according to the Boolean rule set. Therefore, both indicators are combined to provide an overall score that relates to the likelihood of the logged event being a real threat.

D. APPLICATION TO AMEND THE PATENT

135. The application to amend the patent is unfounded. Accordingly, it can remain open, whether the number of in total 32 auxiliary requests is reasonable and admissible pursuant to R. 30.1 (c) RoP (cf. LD Mannheim, decision dated 6 June 2025, UPC_CFI_471/2023 para. 140).

Claim amendments 1 to 3 lack novelty over US'358 (HAWK).

136. Claim amendment 1 corresponds to the main request wherein feature 1.4.2.1 reads as follows:

determining, by the computing device, a first reportability likelihood for each event log based on a static algorithm, wherein the static algorithm is a human-designed algorithm" **[feature CA1-1.4.2.1]**.

137. Claim amendment 1 defines the static algorithm as human designed. Accordingly, any algorithm that has not been entirely, if technically possible, created by artificial intelligence meets this requirement. HAWK discloses feature CA1-1.4.2.1 directly and unambiguously as the static Boolean rule is based on user defined detection rules (cf. paras. [0054], [0058]) and a pre-defined score rule set (cf. paras. [0071], [0075]), which can obviously be human-designed/programmed.

138. The same applies for claim amendment 2.

139. In claim amendment 2 feature 1.4.2.1 reads as follows:

determining, by the computing device, a first reportability likelihood for each event log based on a static algorithm, wherein the static algorithm is a heuristic algorithm **[feature CA2-1.4.2.1]**.

140. In the absence of any indications to the contrary, a *heuristic* algorithm is – in accordance with common general knowledge – one that is based on prior knowledge and experience which is used to determine and assess (new) potential threat events (cf. para. [0046], l. 8 et seqq., *'heuristic knowledge provided by (human) cyberanalysts'*, and Figure 4 of the patent-in-suit).

141. Contrary to Claimant's assessment, feature CA2-1.4.2.1 does not establish any restrictions concerning the machine-learned algorithm pursuant to feature 1.4.2.2. In particular, to use a *heuristic* machine-learned algorithm as well, in the sense that it is (solely) trained with previously identified threat and benign events and thus based on this experience, is not excluded from the scope of claim amendment 2.

142. Feature CA2-1.4.2.1 is unambiguously and directly disclosed by HAWK. The use of a pre-defined Boolean ruleset is an algorithm obviously based on prior experience to determine whether an event occurrence item is a potential threat or not. As HAWK states, the ruleset is any list of rules that is associated with a positive or negative score, like destination (URL) or a username (cf. para. [0075] and Table 1 of HAWK).

143. Claim amendment 3 combines the additional features of claim amendments 1 and 2 defining the static algorithm in feature 1.4.2.1 as *a human-designed heuristic algorithm* (**feature CA3-1.4.2.1**).

144. Thus, it lacks novelty over HAWK for the same reasons as claim amendments 1 and 2.

Claim Amendment Group 4 lacks an inventive step

145. In Claim Amendment 4, feature 1.4.2.2 of the main request reads as follows:

determining, by the computing device, a second reportability likelihood for each event log based on a machine-learned algorithm provided by a machine learning system, wherein the machine-learned algorithm is based on an artificial neural network; and **[feature CA4-1.4.2.2].**

146. Feature CA4-1.4.2.2 specifies the machine-learned algorithm as *"based on an artificial neural network"*. In absence of any indications to the contrary, the term artificial neural network (ANN) in the description of the patent-in-suit (cf. para. [0017] l. 19; para. [0049] l. 15) refers to common general knowledge and therefore to computer systems designed to mimic how the human brain processes information and to use artificial neurons to analyse data, identify patterns and make predictions. A supervised ANN may be fed with labelled training data, e.g. previously identified threat events, to learn features that classify an event in a network as potentially

threatening.

147. Apart from that, feature CA4-1.4.2.2 establishes no further mandatory characteristics of the ANN. Thus, also in the context of claim amendment 4, the machine-learned algorithm does not have to be capable ‘*to reflect the unknown pattern of the cyberanalysts’s experience’*.

Novelty

148. Claim amendment 4 is novel over HAWK. To use an ANN to determine a second reportability likelihood is not directly and unambiguously disclosed in HAWK, which is undisputed by Defendant.

Lack of inventive step

149. However, claim amendment 4 lacks an inventive step starting with HAWK in conjunction with common general knowledge or, alternatively, in combination with EP’879 (HLCC17).

Legal Framework

150. According to Art. 56 EPC, an invention shall be considered as involving an inventive step if, having regard to the state of art, it is not obvious to a person skilled in the art.

151. The suitable starting point for the assessment of inventive step is not limited to the closest prior art. Since there may be several ways to arrive at a conclusion, several suitable starting points may exist. The decisive point is rather, whether such starting point constitutes a suitable starting point, which the relevant person skilled in the art would take into account, if confronted with the problem to be solved (cf. Central Division Munich Section, decision of 16 July 2024, UPC_CFI_14/2023 mn. 8.6; Central Division Paris Seat, decision of 21 January 2025, UPC_CFI_311/2023 mn. 57). In this regard, on a regular basis, a solution as claimed is obvious, if, starting from a suitable starting point in the prior art, the skilled person would be motivated (i.e., have an incentive) to consider the solution of the invention and implement it as a next step (cf. Central Division Munich Section, decision of 16 July 2024, UPC_CFI_14/2023 mn.

8.6; Court of Appeal, decision of 25 November 2025, UPC_CoA_528/2024, Amgen v Sanofi-Aventis; decision of 25 November 2025, UPC_CoA 464/2024, Meril v Edwards)

152. Applying these principles, the subject-matter of claim amendment 4 is not based on an inventive step, starting from HAWK in conjunction with common general knowledge or, alternatively, in combination with HLCC17.

Lack of inventive step starting from HAWK in combination with common knowledge

153. The use of a particular means may be obvious even without a corresponding specific motivation if, by its nature, said means, as general means to be considered for a plurality of applications, belong to the general knowledge of the relevant skilled person, the use of the functionality in question is objectively appropriate in the context to be assessed, and no special circumstances can be identified that would render its application appear impossible, difficult or otherwise impractical from a technical point of view (cf. Local Mannheim, decision of 2 April 2025, UPC_CFI_359/2023 mn. 121, following BGH, decision of 15 June 2021 – X ZR 58/19, GRUR 2021, 1277 mn. 47 – Führungsschienenanordnung).

154. These requirements are met in the case at hand.

Suitable starting point

155. HAWK is a suitable starting point with respect to feature CA4-1.4.2.2. As mentioned above, HAWK addresses identical technical problems as the patent-in-suit and seeks to provide improved systems and methods for review and analysis of potential threat events across networks to efficiently manage large amount of data. To achieve this target, HAWK is not explicitly restricted to the exemplary embodiment suggesting the use of a Naïve Bayes classifier to assess potential threat events in addition to a static Boolean rule.

156. Rather, HAWK also discloses the use a method for Support Vector Machine Learning (SVM) of event/log collections for comparison of future similar activity. In particular, the SVM learning is being applied for log analysis as it relates to pattern and state

detection (cf. para. [0136]). In general, HAWK suggests using (any) commonly known automated unsupervised learning rule for the purpose of log analysis. Especially, claim 12 as well as paras. [0142] et seqq. generally refer to the use of automated unsupervised learning rules.

Common general knowledge

157. Claimant did not dispute Defendant's allegation that an ANN – amongst other machine-learned algorithms – had been commonly known in the context of assessing (potential) threat events in a computer network before the earliest priory date of the patent-in-suit. As Claimant again asserts in this context that the previously known ANNs are (allegedly) not designed for *'detecting patterns with regard to the cyberanalysts' experience'*, this is, as already explained above, irrelevant. Neither feature 1.4.2.2 of the main request nor feature CA4-1.4.2.2 of claim amendment 4 establish such a restriction. Therefore, the use of any commonly known ANN that is capable of determining a value assessing the threat level of an (potential) threat event in a computer network is sufficient.

Lack of inventive step

158. HAWK, *see supra*, indicates to use various machine learned methods in the context of threat event analysis. Thus, a person skilled in the art would resort to other M/L algorithms available in common general knowledge to assert the threat level of an event in a computer network, including ANN, as disclosed in EP'879 (HLCC17, cf. para [0096]) and an article by Jake Ryan et. al., *'Intrusion Detection with Neural Networks'*, published on 1 December 1999 (HLCC19), for example.

159. Using a commonly known ANN instead of, especially, a Naïve Bayes algorithm to determine a value relating to potential threat level of an event in a computer network is objectively appropriate and fits within the HAWK teaching, as there are, contrary to Claimant's arguments, no technical difficulties or impracticalities that would make it imperative for the machine-learned algorithm to be a Naïve Bayes classifier. Rather, HAWK relies on determining an indicator relating to the threat level of an event in a computer system by using, in the exemplary embodiment, a

Naïve Bayes classifier to provide a Bayesian score which can, additionally, be assessed and scored by a rule of a static algorithm. The exact same functionality can be established by another machine-learned algorithm, e.g. an ANN, providing – for example – a probability value.

Lack of inventive step starting from HAWK in combination with HLCC17

160. Even if the fact that the use of (other) machine-learned algorithms, such as an ANN, to assess the threat level of a potential threat event in a computer network would not have been part of common general knowledge of the skilled person at the (earliest) priority date of the patent-in-suit, the subject-matter of claim amendment 4 would be anticipated starting from HAWK in combination with HLCC17.

Motivation starting with HAWK

161. The skilled person tasked with improving assessment and processing of potential threat events in a computer network has motivation to start from HAWK, because it provides superior properties due to the advantageous design of involving different algorithms for determining an overall reportability likelihood of a potential threat event. As HAWK itself is not, *see supra*, limited to the use of a Naïve Bayes algorithm but also relates to SVM and unsupervised machine-learning algorithms in general in the context of assessing and processing potential threat events in a computer network, the skilled person consulting HAWK would be motivated to find an alternative or improved machine-learned algorithm to provide a value that reflects a reportability likelihood of a certain event to further improve the analysis of potential threat events.

162. Consulting HLCC17 would lead the person skilled in art to the scope of feature CA4-1.4.2.2 without an inventive step.

Subject of EP'879/HLCC17

163. HLCC17 relates to malicious software detection in a computing system.

164. According to the description of HLCC17, with millions of online resources that are available via millions of corresponding Uniform Resource Locators (URLs),

organizations have difficulty monitoring and identifying those information access requests that are associated with malicious content, such as malware or other malicious code (cf. para. [0003]).

165. Against this background, HLCC17 wants to provide systems and methods for detecting malicious software and/or otherwise undesirable access of online resources in a computing system, such as among a network of computers of an organization. Some of the suggested systems or methods can analyze data, such as URL data items, in order to identify the infected systems. The disclosed invention also tries to improve functioning of a computing system by reducing data to be analyzed to those data items most likely associated with malicious software, improving processing speed when determining potentially malicious addresses (cf. para [0004]).

166. In one exemplary embodiment, the system uses machine learning techniques to identify a URL as malicious. Relating to FIG. 10A and FIG 10B, as shown below, the description of HLCC17 illustrates the steps of training and evaluation of machine learning (cf. para. [0092]):

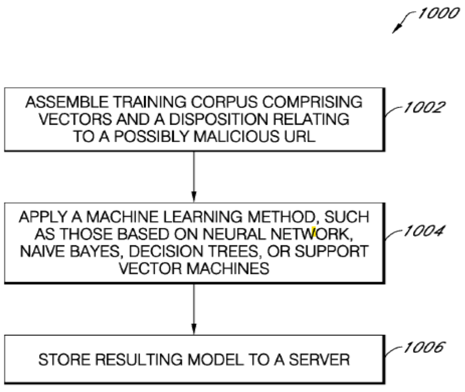

1000
ASSEMBLE TRAINING CORPUS COMPRISING VECTORS AND A DISPOSITION RELATING TO A POSSIBLY MALICIOUS URL
1002
APPLY A MACHINE LEARNING METHOD, SUCH AS THOSE BASED ON NEURAL NETWORK, NAIVE BAYES, DECISION TREES, OR SUPPORT VECTOR MACHINES
1004
STORE RESULTING MODEL TO A SERVER
1006

FIG. 10A

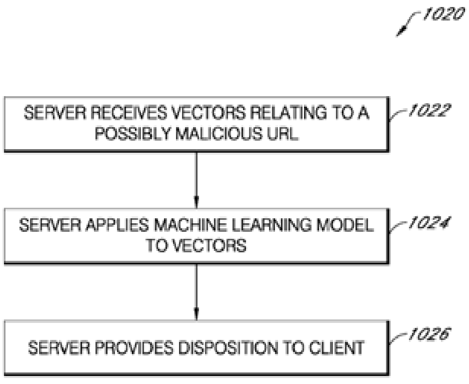

1020
SERVER RECEIVES VECTORS RELATING TO A POSSIBLY MALICIOUS URL
1022
SERVER APPLIES MACHINE LEARNING MODEL TO VECTORS
1024
SERVER PROVIDES DISPOSITION TO CLIENT
1026

FIG. 10B

167. FIG. 10A is a flowchart of an example of a machine learning method 1000 as applied to malware detection. During training phase, a corpus of training data 1002 is used to derive a model. It is desirable for the data inputted to the machine learning to be representative of real-world scenarios. The model also takes as an input a disposition determined by a human analyst with expertise in diagnosing a URL as benign or

malicious (cf. paras. [0093] to [0095).

168. Next, as shown in block 1004, a machine learning method is applied to the training corpus 1002.The methods by which training can be done include, but are not limited to Support Vector Machines, Neural Networks, Decision Trees, Naïve Bayes, Logistic Regression, and other techniques from supervised, semi-supervised and unsupervised training. The training or model-derivation may be practiced with any of the above techniques so long as they can yield a method for classifying URLs as benign or malicious (cf. para. [0096]).

169. Once the training is sufficient and the model is derived, it can be used to automatically evaluate new instances of URLs that are presented to the computer network in practice. In this regard, there is a second evaluation phase, which is shown in the flowchart of FIG. 10B, wherein the model is applied in block 1024 to determine whether a URL is likely malicious or benign. In block 1026 the server outputs a decision based on the model. The output can be a binary classification (malicious or not malicious). Advantageously, the output is a score that represents a likelihood of this distinction, such as a score from 0 to 100 where 0 represents an overwhelming likelihood that the URL is benign and 100 represents an overwhelming likelihood that the URL is malicious (cf. para. [0097]).

<u>Lack of an Inventive Step</u>

170. Against this background, a person skilled in the art starting with HAWK and motivated to search for an alternative or better machine-learned algorithm to determine a likelihood of a threat of an event in a computer network, would easily understand that Naïve Bayes algorithm directly disclosed in HAWK in the context of evaluating and scoring the threat level of an event occurrence item can be substituted with any other known machine-learned algorithm capable of determining values indicating the likelihood of a malicious event in a computer network, such as a Neural Network, as they are disclosed in HLCC17 as viable options in the context of intrusion detection and event scoring. This is especially true because HAWK itself does not only disclose Naïve Bayes, but also SVM and in general unsupervised algorithms in context of threat event analysis, the latter of which are

also explicitly mentioned in HLCC17 as further viable alternatives.

171. Thus, a skilled person motivated to search for an alternative machine learned algorithm would and could implement the machine learned algorithms disclosed in HLCC17, including ANN, as they are advantageously capable of determining a value indicating the likelihood of a threat event (cf. para. [0097]). This is consistent with the technical teaching of HAWK according to which such a value might additionally be evaluated by a static algorithm.

172. <u>Claim amendments 4A to 4C</u> which add the features of claim amendments 1 to 3 to claim amendment 4 respectively are not patentable for the same reason. Insofar, additional reference is made to the arguments in relation to claim amendments 1 to 3.

<u>*Claim Amendment Group 5 is not patentable*</u>

173. In <u>Claim Amendment 5</u> feature 1.4.2.2 of the main request reads as follows:

determining, by the computing device, a second reportability likelihood for each event log based on a machine-learned algorithm provided by a machine learning system, <u>*wherein the machine-learned algorithm is trained based on previously analyzed event logs*</u>*; and* **[feature C5-1.4.2.2]**

174. Feature C5-1.4.2.2 requires using previously analyzed event logs to train the machine-learned algorithm. It does not exclude other training data or impose further restrictions regarding the applied training method. Any suitable method, be it supervised, semi-supervised or unsupervised, falls within the scope of claim amendment 5.

175. Feature C5-1.4.2.2 is directly and unambiguously disclosed by HAWK.

176. Relating to FIG. 20, as shown below, HAWK describes an exemplary process to update the training database of the Bayesian algorithm:

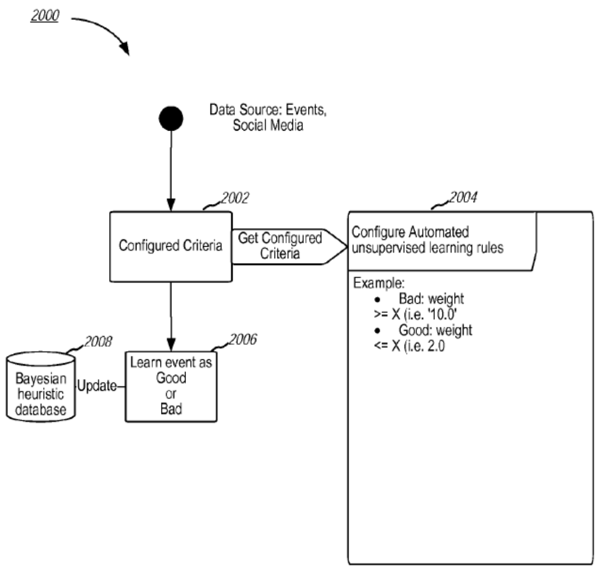

2000
Data Source: Events, Social Media
2002
2004
Configured Criteria
Get Configured Criteria
Configure Automated unsupervised learning rules
Example:
• Bad: weight >= X (i.e. '10.0'
• Good: weight <= X (i.e. 2.0
2008
2006
Bayesian heuristic database
Update
Learn event as Good or Bad

FIG. 20

177. The method illustrated in FIG. 20 comprises a method for unsupervised learning support of the training of analytics for the purpose of log analysis (cf. para. [0142]), like events and social media.

178. The data source input includes events and social media data. At block 2002 a user can configure criteria, i.e. automated unsupervised learning rules, by setting specified search parameters, illustrated in table 2004. As an example of such rules, an event is designated as a "bad" event if a weight is set greater than or equal to a value X. Conversely, an event is designated a "good' event if a weight less than or equal to a value X (cf. para. [0143]). Accordingly, at block 2006, for each event that matches the given criteria, the event can be learned as good or bad, which of course is depending upon the matching criteria. Additionally, the Bayesian heuristic database (2008) may be updated with the learned event information. The same is disclosed in claim 12 of HAWK.

179. Accordingly, HAWK discloses the use of previously analyzed events to update the database of the machine-learned Naïve Bayes algorithm to train it with said information.

180. Claim amendments 5A to 5G, which add features of previous claim amendment (groups) 1 to 4 to claim amendment 5 respectively, are not patentable for the same

reasons mentioned above in the context of said claim amendments.

Claim Amendment Group 6 is not patentable.

181. In comparison to claim amendment 5 feature 1.4.2.2 reads as follows:

determining, by the computing device, a second reportability likelihood for each event log based on a machine-learned algorithm provided by a machine learning system, wherein the machine-learned algorithm is trained based on event logs previously analyzed by a cyberanalyst with regard to reportability; and [**feature CA6 -1.4.2.2**]

182. Feature CA6-1.4.2.2 lacks at least an inventive step over HAWK. As explained above in the context of claim amendment 5, HAWK discloses to base the training of the machine-learned Naïve Bayes algorithm on previously analyzed event logs. Based on this disclosure, it does at least not require an inventive step for a person skilled in the art to perform said previous analysis by a cyberanalyst.

183. Rather, this is a typical implementation of a supervised learning method that uses labelled training data. This is also suggested in HAWK by the fact that a user, and thus a user of the analysis system, can set the corresponding analysis criteria (cf. para. [00143]). This will regularly be a cyberanalyst, which is also indicated by para. [0005], *a manager or operator of the system*, and para. [0042], *network system manager or operator*.

184. Claim Amendments 6A to 6G implement feature CA6-1.4.2.2 and additionally add features of claim amendments (groups) 1 to 4 to 5G respectively and are therefore not patentable for the same reasons mentioned above.

Claim amendment group 7 lacks an inventive step

185. In claim amendment 7 the following additional features CA7-1.7 and CA7-1.8 are added to the independent claim 1 of the main request:

receiving, from the analysis system, report data generated based on analyzed event logs; and **[feature CA7-1.7]** *updating training data for the machine-learned algorithm based on the report data generated based on the analyzed event logs* **[feature CA7-1.8]**.

186. These additional features of claim amendment 7 lack at least an inventive step over HAWK. As mentioned above, HAWK discloses the general idea of claim amendment 5, i.e. to train and update the machine-learned algorithm based on previous analyzed events. Against this background, the more detailed implementation of this basic idea in claim amendment 7 cannot constitute an inventive step over HAWK. Rather, it is obvious to a person skilled in the art, that previous analyzed events and corresponding date can be derived from a (cyber)analysis system wherein the final assessment of an event in a computer network takes place.

187. Claim amendment 7A adds feature CA4-1.4.2.2 of claim amendment 4 to claim amendment 7 and lacks, accordingly, an inventive step starting with HAWK for the same reasons mentioned above in the context of claim amendment 4.

Claim Amendments 8 and 8A lack an inventive step

188. In Claim amendment 8 feature 1.4.2.2 of the main request is amended as follows:

determining, by the computing device, a second reportability likelihood for each event log based on a machine-learned algorithm provided by a machine learning system, wherein the machine-learned algorithm determines, for each event log, the second reportability likelihood for the event log based on at least one of: **[feature CA8- 1.4.2.2]**

a domain name associated with the event log, ***[CA8-1.4.2.2.1]***

an entropy value of the domain name associated with the event log, [***CA8-1.4.2.2.2]***

a number of labels of the domain name associated with the event log, ***[CA8-1.4.2.2.3]***

a string length of the domain name associated with the event log, ***[CA8-1.4.2.2.4]***

a size of data associated with the event log, ***[CA8-1.4.2.2.5]***

threat intelligence provider data associated with the event log, and ***[CA8-1.4.2.2.6]*** *a number of threat intelligence providers associated with the event log;* and ***[CA8-1.4.2.2.7]***

Construction of Claim amendment 8

189. According to feature CA8-1.4.2.2 it is sufficient if the determination of a second reportability likelihood is based on *at least one* of the aspects of features CA8-1.4.2.2.1 to 1.4.2.2.7. Accordingly, the use of just one of said aspects, potentially in combination with other aspects not mentioned in claim 1 pursuant to claim amendment 8, is sufficient.

190. Conversely, claim amendment 8 does not impose the existence of a group including all aspects according to features C8-1.4.2.2.1 to C8-1.42.2.7 and the selection of at least one of these aspects from the existing group, but rather the determination of the second reportability likelihood based on at least one of the aspects mentioned in the following features (feature C8-1.4.2.2). If at least one of these aspects is relevant for determining the second reportability likelihood, it is irrelevant whether the other aspects were available in abstract terms for said determination.

191. This is in line with Claimant's own understanding of claim amendment 8 as presented in the Reply to the Statement of defence regarding the Infringement action (cf. para. 110 et seqq.) and confirmed during the oral hearing.

192. Furthermore, as Defendant correctly assumes, feature C8-1.4.2.2 does not specify that the machine-learned algorithm itself has to determine the aspects of features CA8-1.4.2.2.1 to 1.4.2.2.7. Rather, the determination of the reportability likelihood is also *based on*, for example, an entropy value of the domain name associated with the event log (feature C8-1.4.2.2.2), if the machine learned algorithm takes such an entropy value into account to determine the second reportability likelihood, even if the entropy value was determined by another functional unit or implemented in the machine-learned algorithm from an outside source.

193. An *entropy value* of the domain name in the meaning of feature C8-1.4.2.2.2 is any value that indicates whether a domain name consists of expectable words or strings of letters and/or numbers, which could indicate a benign event, or of random

combination of strings, which could indicate a malicious event. This common general understanding is in line with the description of the patent-in-suit (cf. para. [0054]). As an exemplary embodiment, the patent-in-suit suggests using probability distribution for bigrams (consecutive pairs of characters) or trigrams (consecutive triples of characters), or, in general, polygrams, in English text to gain an entropy value *Entropy()* (cf. para. [0057]). Contrary to Claimant, claim amendment 8 is not limited to the exemplary embodiment in para. [0056] that describes how such an entropy value could be calculated.

194. Based on this claim construction, claim amendment 8 lacks at least an inventive step starting from HAWK in combination with HLCC17.

<u>Motivation starting with HAWK</u>

195. As HAWK does not specify the aspects the machine-learned Naïve Bayes algorithm could take into account to determine a Bayesian score, a person skilled in the art, faced with the technical problem to find such relevant attributes, would be motivated to search for suitable characteristics that could indicate a potential threat within network communication.

196. As HLCC17 relates to the identical technical field as HAWK, the person skilled in art would be motivated to consult HLCC17.

<u>Further scope of HLCC17</u>

197. As mentioned above in context of claim amendment 4, HLCC17 in general relates to systems and methods for detecting malicious software in a computing system and computer-related media (cf. para. [0002]). More specifically, HLCC17 relates to a system and method for identifying malicious Uniform Resource Locator (URL) data items from a plurality of unscreened data items that have not been previously identified as associated with malicious URLs (cf. claims 1 and 10).

198. The claimed solution of HLCC17 involves assigning a score based on a plurality of factors relating to the possible malicious URL data items. As described in para. [0074], potentially malicious URL identified by one or more pre-filter systems can be

passed to a scoring processor. The scoring processor assesses a plurality of factors or ‘vectors’ relating the URL and can assign a score to the URL based on a machine learning algorithm. Any of such pre-filters can also be incorporated as vectors for the machine learning algorithm.

199. One possible factor or ‘vector’ is a so called ‘n-gram vector’ (cf. para. [0075]), which is described in HLCC17 with reference to the following FIG.7A:

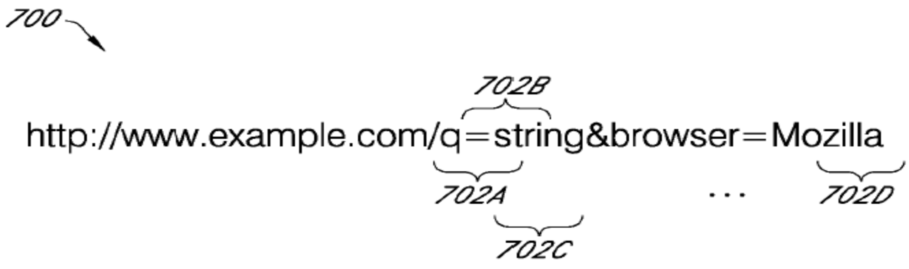


FIG. 7A

200. An n-gram is a unique sequence of N consecutive characters. URL 700 of FIG. 7A shows the “filepath” divided into series of n-grams. In this example, each n-gram represents three consecutive characters. Other numbers of characters (such as four, five or six) are also contemplated. N-gram 702A comprises the first three-character block of the filepath (namely, q=s). N-gram 702B comprises the second three-character block of the filepath (namely, =st). N-gram 702C comprises the third three-character block of the filepath (namely, str). The (whole) filepath is divided into a series of such three-character bocks, concluding with N-gram 702D, representing the last three-character block of the filepath (namely, lla) (cf. para. [0076]).

201. Suitable program instructions are executed by a computer processor in order to cause the computing system to parse a potentially malicious URL to identify the domain name and filepath and detect occurrences of n-grams in the filepath by sequentially moving fixed-length window (e.g. three characters) over the filepath and identifying the string values at each window position. For example, the data storage associated with *example.com* can be incremented 1 count for the n-gram ‘g=s’, 1 count for the n-gram ‘=st’, 1 count for the n-gram ‘str’, and one count for the

n-gram “lla” (cf. para. [0077]).

202. Suitable program instructions are further executed by a computer processor in order to cause the computing system to calculate a distribution of the n-grams for the filepaths of a domain name. Advantageously, the domain name is associated with a very large amount of Internet traffic. The following FIG. 7B shows an example distribution for a benign domain name and illustrates a smooth distribution between n-grams with a large number of occurrences and n-grams with a small number of occurrences (cf. para. [0078]):

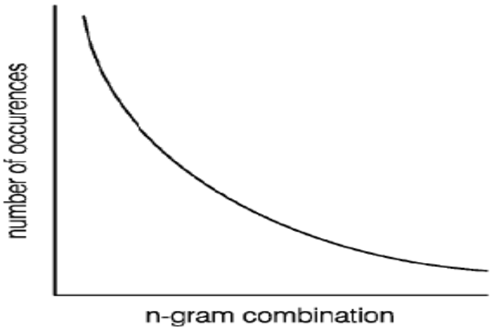


FIG. 7B

203. The distribution of FIG. 7B is the expected distribution. FIG 7C and 7D are example distributions for malicious domain names. Figure 7C represents a domain name where each n-grams has a small number of occurrences. One can expect this kind of distribution where each filepath represents strings of random characters. FIG. 7D represents a domain name where a small number of n-grams each have a large number of occurrences. One can expect this kind of distribution where the same filepath is used repeatedly (cf. para. [0078]).

204. FIG. 7E is flowchart of an example of a n-gram distribution comparison method 720 of a vectoring system as applied to malware detection (cf. para. [0079]):

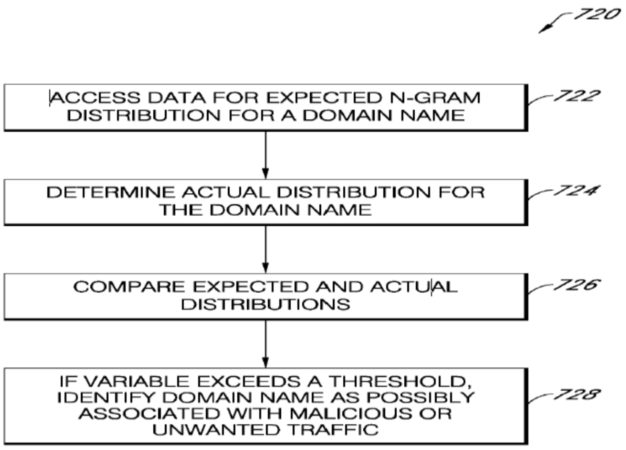

720
ACCESS DATA FOR EXPECTED N-GRAM DISTRIBUTION FOR A DOMAIN NAME
722
DETERMINE ACTUAL DISTRIBUTION FOR THE DOMAIN NAME
724
COMPARE EXPECTED AND ACTUAL DISTRIBUTIONS
726
IF VARIABLE EXCEEDS A THRESHOLD, IDENTIFY DOMAIN NAME AS POSSIBLY ASSOCIATED WITH MALICIOUS OR UNWANTED TRAFFIC
728

FIG. 7E

205. As shown in block 722, the system can access data for an expected n-gram distribution for a benign domain name. In block 724, the system determines the actual distribution to a particular domain name. In block 726, the expected n-gram distribution and actual distribution are compared. As shown on block 728, if variance between the distributions exceeds a threshold, the URL associated with the domain name can be identified as possibly malicious. The variance and/or other suitable parameters relating to n-grams can be output to the scoring processor (cf. para. [0080], wherein scoring can be applied by a machine-learning technique using any vector, including n-gram vectors or relating parameters, as described in HLCC17 (cf. paras [0092] et. seqq., para. [0074]).

Lack of an inventive step

206. Taking said disclosure of HLCC17 into account, a person skilled in art searching for suitable attributes to apply the machine-learned algorithm on, would be motivated to implement the idea to use an entropy value, i.e. the variance of the distribution of n-grams in relation to a threshold, relating to a domain name derived from using n-gram vectors for identifying potentially malicious domain names. That is in accordance with one possible implementation suggested by the patent-in-suit.

207. Further, the technical teaching of HLCC17 is especially not limited to evaluate the *'filepath'* of an URL, i.e. the part of an URL after the top-level domain (like: *.com*), as can be directly and unambiguously taken from FIG.7E and the respective description

in para. [0080], which relate to applying n-gram vectors on the domain name in general.

208. Nevertheless, even if HLCC17 would not explicitly disclose the use of the second-level domain name to derive an entropy value, but only the 'filepath' of the URL, it would not require an inventive step to apply the general concept of using n-gram vectors to identify malicious content to the entire URL, including the second-level domain name. This is particularly the case, since HLCC17 itself highlight that it is the domain name of the filepath, which is considered benign or malicious.

209. Further, HLCC17 discloses to use the n-gram vectors – *inter alia* – when applying the machine learned model to determine a score that represents a likelihood of the URL being benign or malicious (cf. para. [0097]; [0074], [0080]). Therefore, the person skilled in art consulting HLCC17 readily understands, that the entropy value gained through applying n-gram vectors on the domain name is useful for determining a likelihood of a potential threat.

210. As stated above, according to claim amendment 8 the machine-learned algorithm does not have to determine the entropy value itself. Thus, it is meaningless, if the Naïve Bayes algorithm disclosed in HAWK would be capable of determining such a value. Rather, it is sufficient if it is capable of taking such a value into consideration when determining the (second) reportability likelihood, which is undoubtedly the case.

211. Furthermore, as it is sufficient that the determination of the second reportability likelihood is based on one of the aspects of features C8-1.4.2.2.1 to 1.4.2.2.7, it also suffices that the combination of HAWK and HLCC17 leads to an embodiment that uses an entropy value according to feature C8-1.4.2.2.2 for determining the second reportability likelihood to assess a lack of an inventive step of claim amendment 8 over the aforementioned documents.

212. <u>Claim amendment 8A</u> adds feature CA4-1.4.2.2 of claim amendment 4 to claim Amendment 8. Thus, the machine learned algorithm <u>*is based on an artificial neural network,*</u> which cannot contribute an inventive step over HAWK in combination with

HLCC17 for the same reasons mentioned in context of claim amendment 4.

Claim amendment 9 lacks an inventive step

213. In claim amendment 9, the following amendments in comparison to claim 1 of the main request are made (wherein underlining indicates an addition and strike-through a deletion compared to claim amendment 8):

wherein the machine-learned algorithm determines, for each event log, the second reportability likelihood for the event log based on ~~at least one of~~: **[CA9-1.4.2.2];**

~~a domain name associated with the event log,~~

an entropy value of the domain name associated with the event log, **[CA9-1.4.2.2.1]**

wherein the entropy value is computed based on a polygram probability distribution based on the domain name **[CA9-1.4.2.2.1.1]** ~~*a number of labels of the domain name associated with the event log*~~*,*

~~*a string length of the domain name associated with the event log,*~~

~~*a size of data associated with the event log,*~~

~~*threat intelligence provider data associated with the event log, and*~~

~~*a number of threat intelligence providers associated with the event log*~~*;*

214. Claim amendment 9 lacks an inventive step over HAWK in combination with HLCC17. According to feature CA9-1.4.2.2.1 the use of an entropy value of the domain name associated with the event log to determine the second reportability likelihood is now mandatory. However, to use such a value is disclosed in HLCC17, *see supra*.

215. A polygram in the meaning of feature CA9-1.4.2.2.1.1 per definition includes two or more consecutive letters and hence encompasses – *inter alia* – bigrams and trigrams. Since HLCC17 discloses – *inter alia* – trigrams, the added feature does not provide a further distinction.

Claim amendment 10 lacks an inventive step

216. Claim Amendment 10 is based on claim Amendment 9. Additionally, the following feature is added after CA9-1.4.2.2.1.1.

wherein the polygram probability distribution is computed for the leading label of the effective ^{2nd}-level domain of the fully qualified domain name **[feature CA10-1.4.2.2.1.2]**

217. Feature CA10-1.4.2.2.1.2 does not preclude the possibility that the second reportability likelihood is based on further (entropy) values or characteristics as long as an entropy value of the second level domain name is at least part of this determination.

218. As HLCC17 does not limit the application of n-gram vectors to determining an entropy value to the 'filepath' of an URL but rather discloses this concept for domain names in general, it renders feature CA10-1.4.2.2.1.2 obvious. Since the second-level domain is part of the domain name, it can be incorporated into the n-gram probability distribution alongside its 'leading label' and 'effective' components.

219. Even if HLCC17 only would disclose the use of n-gram vectors on the 'filepath' of an URL to gain an entropy value, it would not take an inventive step to apply the general concept to other parts of the URL, including the second-level domain.

Claim amendment 11 lacks an inventive step

220. Claim amendment 11 is based on claim amendment 10. Additionally, the following feature is added after CA10-1.4.2.2.1.2.

wherein the polygram probability distribution is based on bigrams, trigrams, or higher-order polygrams of the leading label of the effective 2nd-level domain **[feature CA11-1.4.2.2.1.3**]

221. As stated in the context of claim amendment 9, a polygram per definition includes two or more consecutive letters and hence encompasses – *inter alia* – bigrams and trigrams. Since HLCC 17 already discloses trigrams, the added feature does not provide a further distinction.

Claim amendment 12 lacks an inventive step

222. Claim amendment 12 includes all feature additions of claim amendment (groups) 1

to 11. The combination of all features does not provide an inventive step. Accordingly, claim amendment 12 lacks an inventive step for the same reasons as the previous claim amendments.

E. INFRINGEMENT/LEGAL CONSEQUENCES

223. As result of the admissible and founded Counterclaim for revocation and the at least unfounded Application to amend the patent, European patent No. EP 3 652 914 B1 is to be entirely revoked in the territories of France and Germany.

224. Consequently, the admissible Infringement action regarding the French and German part of the patent-in-suit is unfounded and to be dismissed.

F. COSTS

225. The decision on the (recoverable) costs with regard to both the Infringement action and the Counterclaim for revocation is based on Art. 69 (1) UPCA, R. 118.5 RoP.

G. VALUE IN DISPUTE

226. The value of the dispute is set to 2.000.000 € (1.000.000 € for the Infringement action and 1.000.000 € for the Counterclaim for revocation) after having heard the parties.

DECISION:

A. The European patent EP 3 652 914 B1 is entirely revoked in the territories of France and Germany.

B. The Application to amend the patent is dismissed.

C. The Infringement action is dismissed.

D. Claimant has to bear the costs of the litigation.

E. The value in dispute for the Infringement action and the Counterclaim of revocation is set at 2.000.000 € (1.000.000 € each).

Delivered in Mannheim on 19 December 2025

NAMES AND SIGNATURES

Presiding judge Tochtermann	Peter Michael Dr. Tochtermann Digital unterschrieben von Peter Michael Dr. Tochtermann Datum: 2025.12.17 17:41:28 +01'00'
Legally qualified judge Sender	Tobias Sender Digital unterschrieben von Tobias Sender Datum: 2025.12.12 09:29:58 +01'00'
Legally qualified judge Härmand	Kai Härmand Digitally signed by Kai Härmand Date: 2025.12.13 11:16:45 +01'00'
Technically qualified judge Kitchen	STEVEN RICHARD KITCHEN Digitally signed by STEVEN RICHARD KITCHEN Date: 2025.12.13 00:02:36 +01'00'
For the Sub-Registrar: Kranz, Clerk LD Mannheim	ANDREAS MICHAEL Kranz Digital unterschrieben von ANDREAS MICHAEL Kranz Datum: 2025.12.17 17:45:29 +01'00'

Information about appeal

An appeal against the present Decision may be lodged at the Court of Appeal, by any party which has been unsuccessful, in whole or in part, in its submissions, within two months of the date of its notification (Art. 73(1) UPCA, R. 220.1(a), 224.1(a) RoP).

Information about enforcement (Art. 82 UPCA, Art. Art. 37(2) UPCS, R. 118.8, 158.2, 354, 355.4 RoP)

The decision has no enforcable content.