US 20210026954A1

(19) **United States**

(12) **Patent Application Publication** (10) **Pub. No.: US 2021/0026954 A1**

Murphy et al. (43) **Pub. Date: Jan. 28, 2021**

(54) **THREAT MITIGATION SYSTEM AND METHOD**

(71) Applicant: **ReliaQuest Holding, LLC**, Tampa, FL (US)

(72) Inventors: **Brian P. Murphy**, Tampa, FL (US); **Joe Partlow**, Tampa, FL (US); **Colin O'Connor**, Tampa, FL (US); **Jason Pfeiffer**, Tampa, FL (US); **Brian Philip Murphy**, St. Petersburg, FL (US)

(21) Appl. No.: **16/940,059**

(22) Filed: **Jul. 27, 2020**

Related U.S. Application Data

(60) Provisional application No. 62/879,105, filed on Jul. 26, 2019, provisional application No. 62/883,797, filed on Aug. 7, 2019.

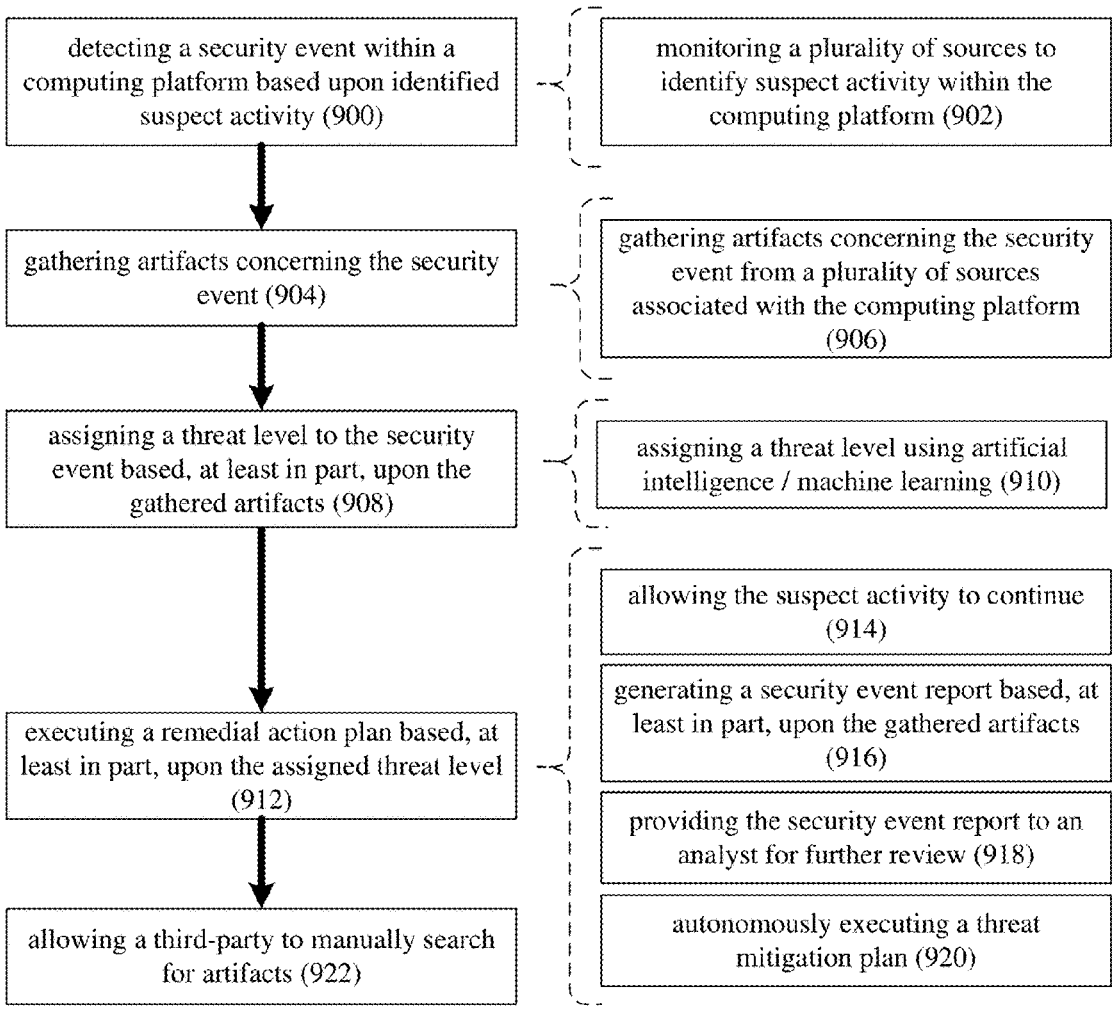
Publication Classification

(51) **Int. Cl.**
G06F 21/55 (2006.01)
G06F 3/0484 (2006.01)

(52) **U.S. Cl.**
CPC ***G06F 21/554*** (2013.01); *G06F 2221/034* (2013.01); ***G06F 3/04842*** (2013.01)

(57) **ABSTRACT**

A computer-implemented method, computer program product and computing system for: a computer-implemented method is executed on a computing device and includes: rendering a threat mitigation user interface that identifies objects within a computing platform in response to a security event; monitoring actions taken by a third-party when investigating the security event; and providing suggestions to the third-party concerning additional actions to be taken by the third-party concerning the investigation of the security event.



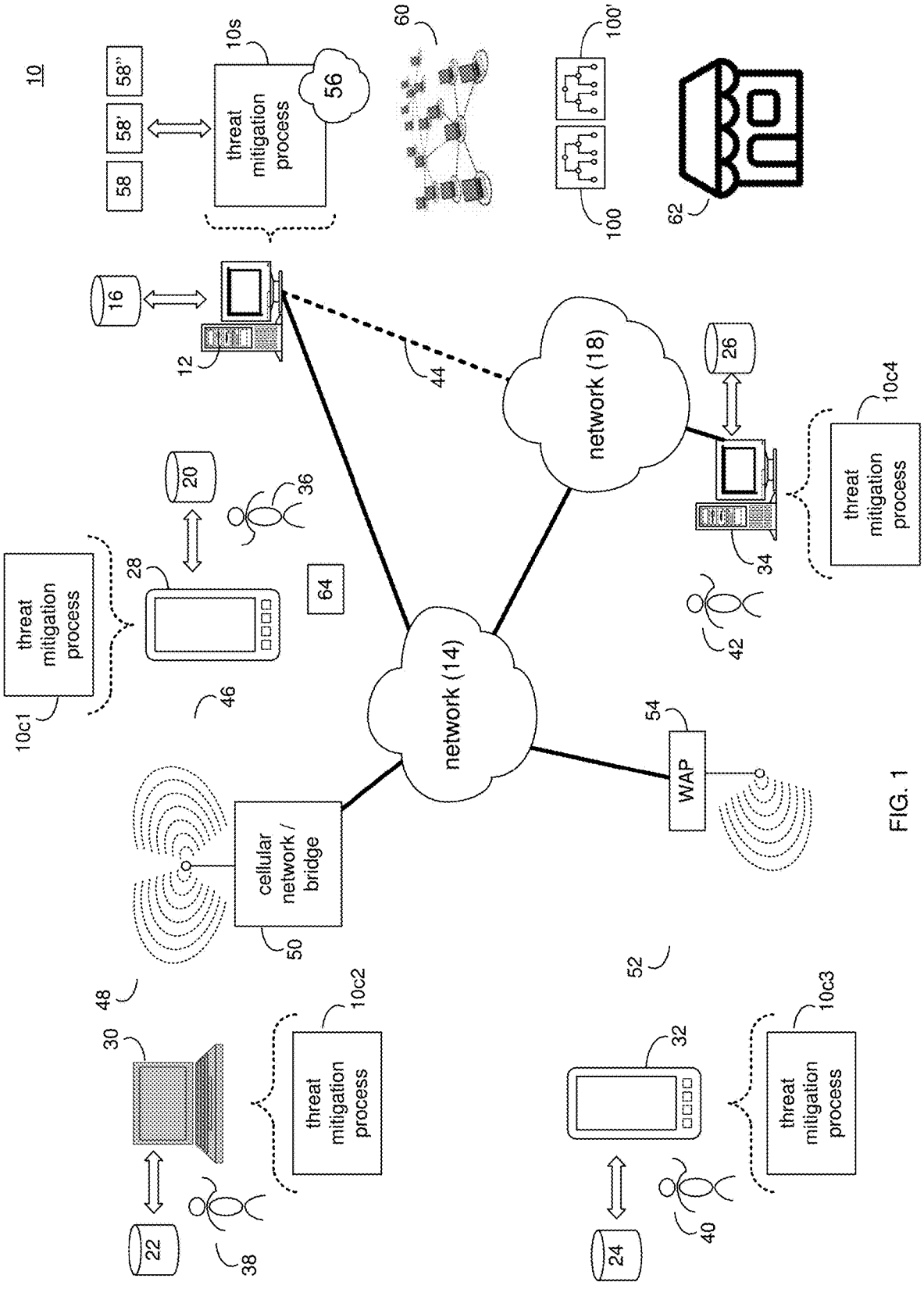

10
10s
threat mitigation process
56
58
58'
58''
60
100
100'
62
16
12
44
network (18)
26
10c4
threat mitigation process
20
36
28
64
10c1
threat mitigation process
46
34
42
network (14)
54
WAP
cellular network / bridge
50
52
48
30
10c2
threat mitigation process
22
38
32
10c3
threat mitigation process
24
40

FIG. 1

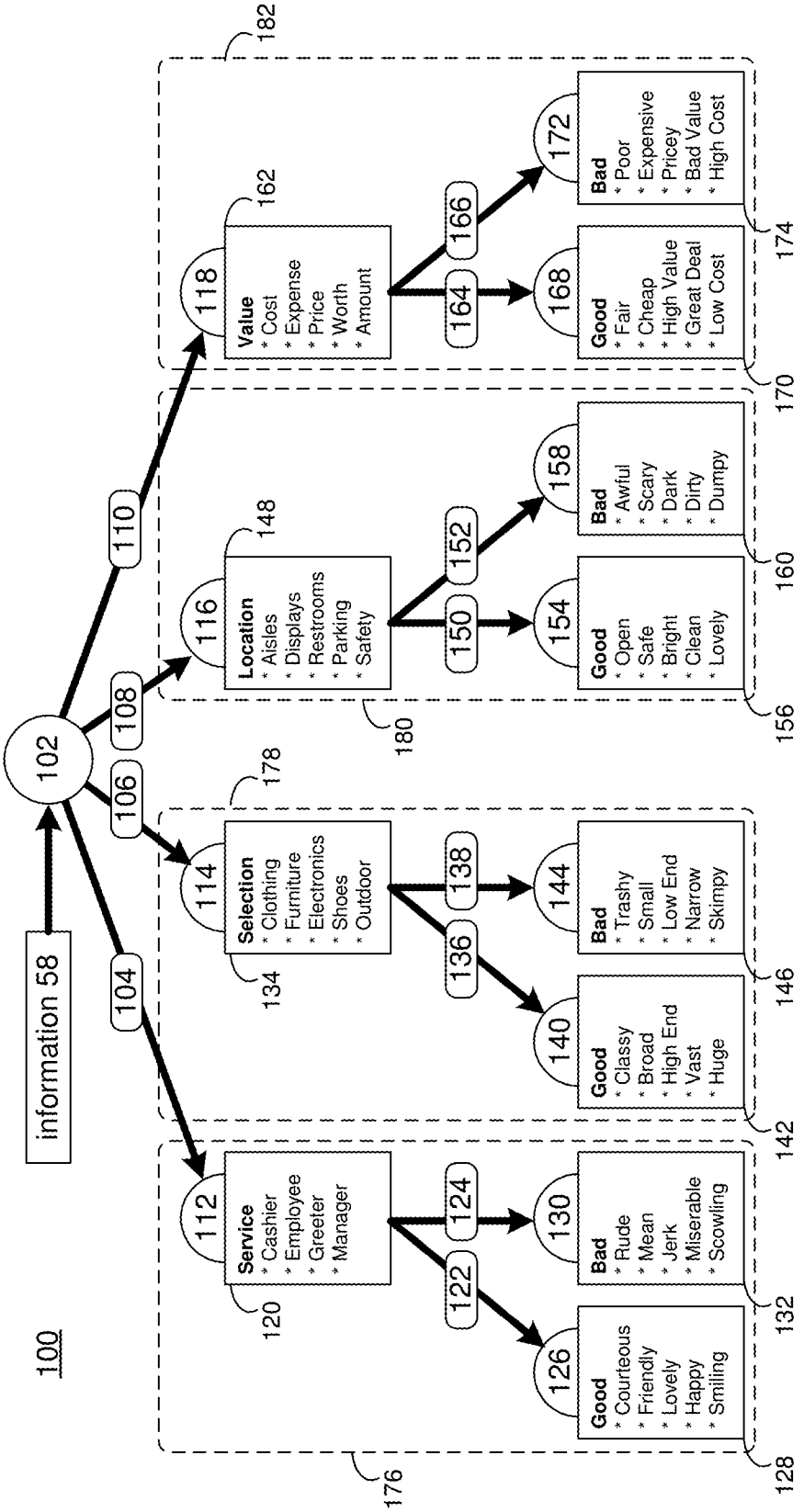

100
information 58
102
104
106
108
110
112
Service
* Cashier
* Employee
* Greeter
* Manager
120
114
Selection
* Clothing
* Furniture
* Electronics
* Shoes
* Outdoor
134
116
Location
* Aisles
* Displays
* Restrooms
* Parking
* Safety
148
180
118
Value
* Cost
* Expense
* Price
* Worth
* Amount
162
122
124
126
Good
* Courteous
* Friendly
* Lovely
* Happy
* Smiling
128
130
Bad
* Rude
* Mean
* Jerk
* Miserable
* Scowling
132
136
138
140
Good
* Classy
* Broad
* High End
* Vast
* Huge
142
144
Bad
* Trashy
* Small
* Low End
* Narrow
* Skimpy
146
150
152
154
Good
* Open
* Safe
* Bright
* Clean
* Lovely
156
158
Bad
* Awful
* Scary
* Dark
* Dirty
* Dumpy
160
164
166
168
Good
* Fair
* Cheap
* High Value
* Great Deal
* Low Cost
170
172
Bad
* Poor
* Expensive
* Pricey
* Bad Value
* High Cost
174
176
178
182

FIG. 2

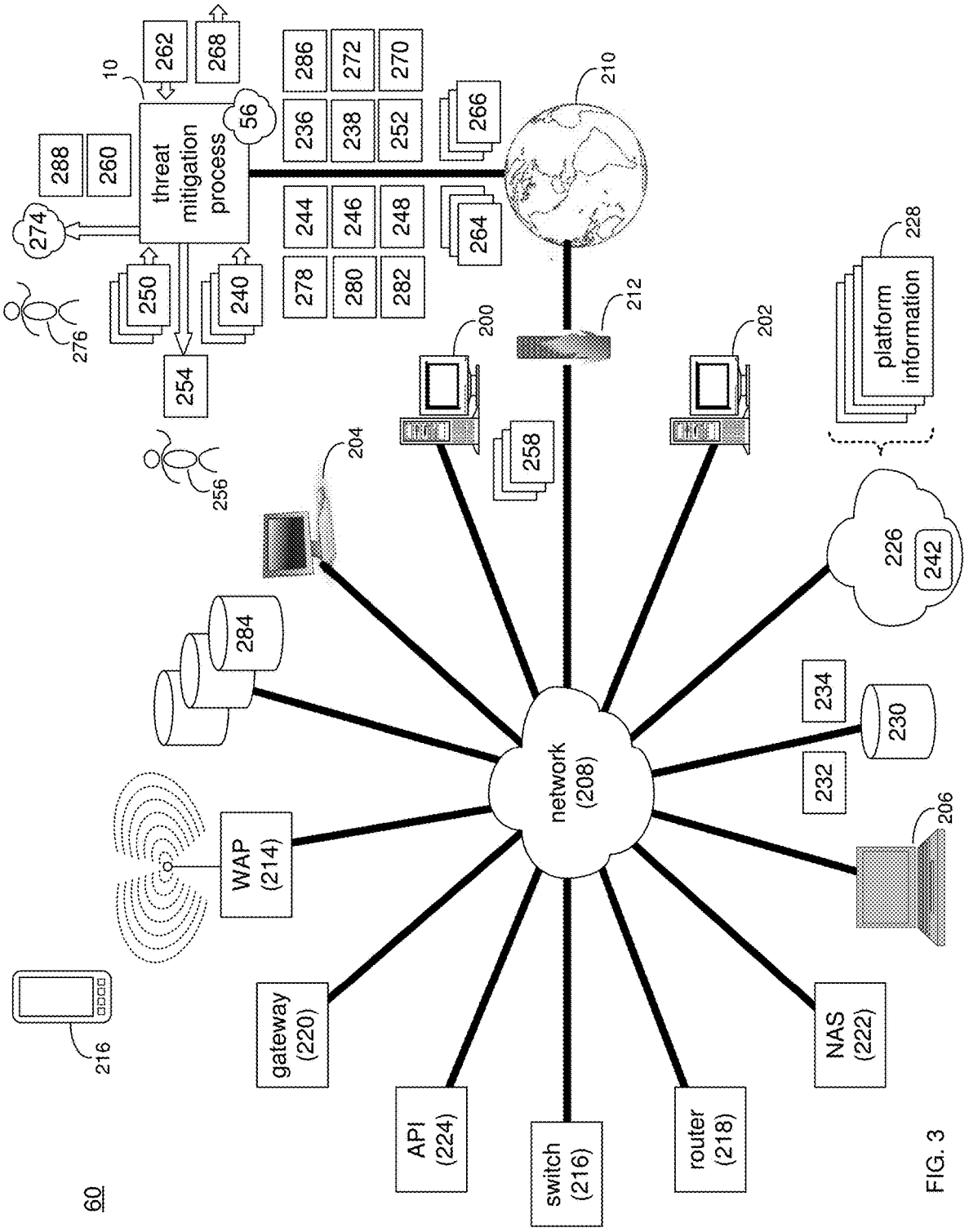

60
216
gateway (220)
WAP (214)
API (224)
switch (216)
router (218)
NAS (222)
network (208)
206
232
234
230
226
242
228
platform information
202
212
200
204
258
210
10
threat mitigation process
56
262
268
288
260
274
276
250
240
254
256
266
264
244
246
248
278
280
282
286
272
270
236
238
252
284

FIG. 3

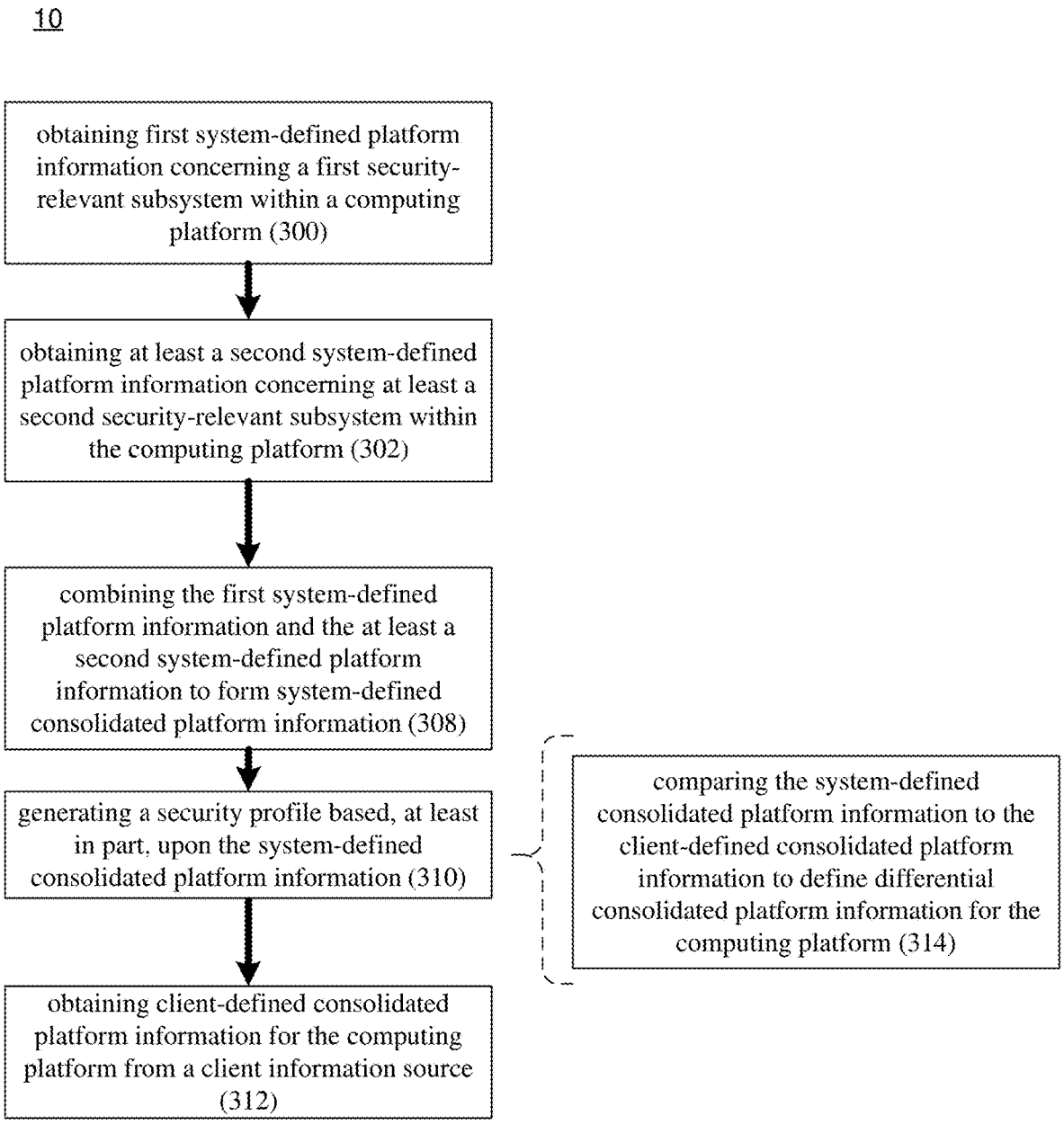

10
obtaining first system-defined platform information concerning a first security-relevant subsystem within a computing platform (300)
obtaining at least a second system-defined platform information concerning at least a second security-relevant subsystem within the computing platform (302)
combining the first system-defined platform information and the at least a second system-defined platform information to form system-defined consolidated platform information (308)
generating a security profile based, at least in part, upon the system-defined consolidated platform information (310)
obtaining client-defined consolidated platform information for the computing platform from a client information source (312)
comparing the system-defined consolidated platform information to the client-defined consolidated platform information to define differential consolidated platform information for the computing platform (314)

FIG. 4

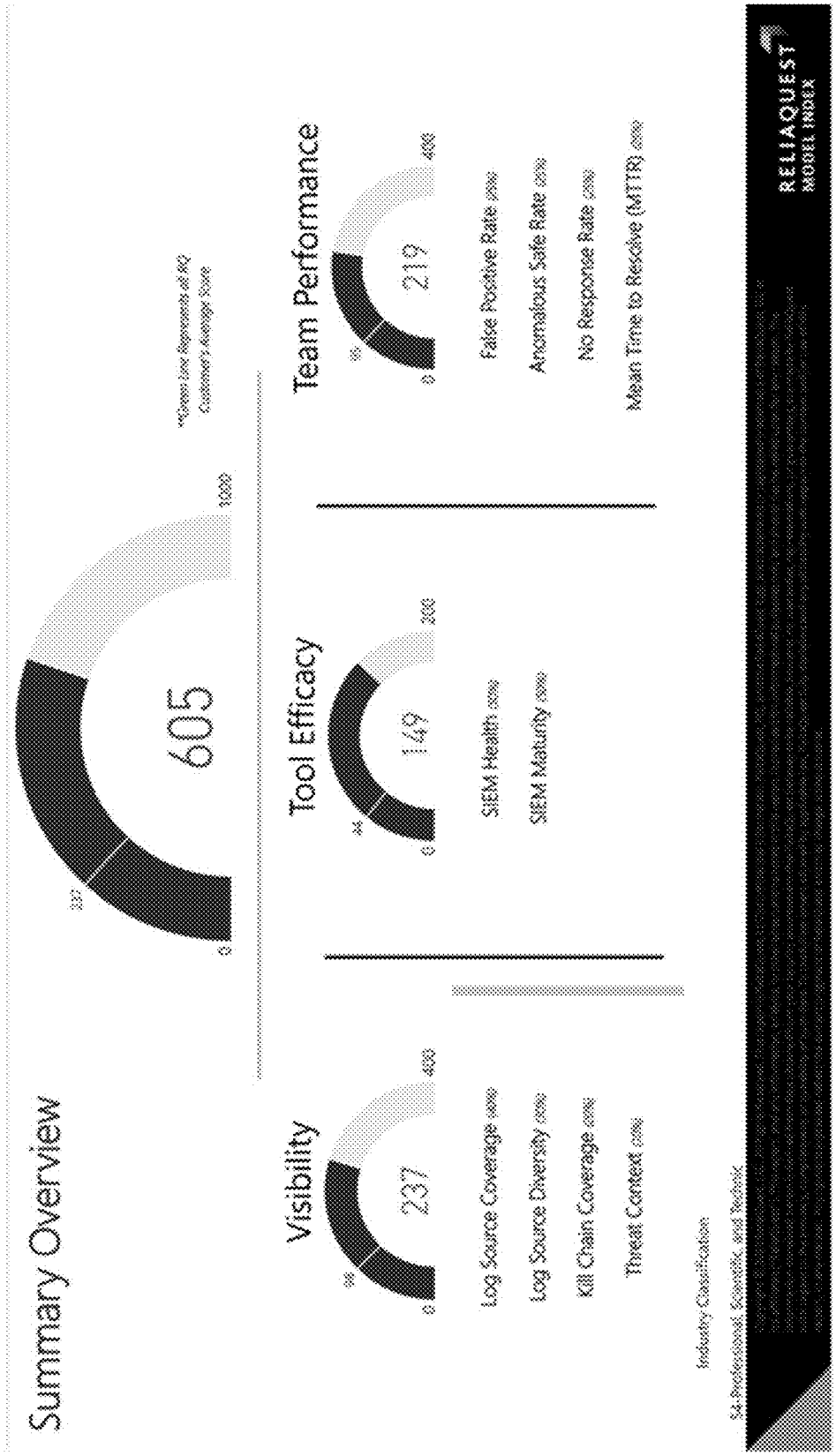

350
Summary Overview
605
1000
0
Visibility
237
400
0
Log Source Coverage ###
Log Source Diversity ###
Kill Chain Coverage ###
Threat Content ###
Tool Efficacy
149
200
0
SIEM Health ###
SIEM Maturity ###
Team Performance
219
400
0
False Positive Rate ###
Anomalous Safe Rate ###
No Response Rate ###
Mean Time to Resolve (MTTR) ###
Industry Classification
RELIAQUEST
MODEL INDEX

FIG. 5

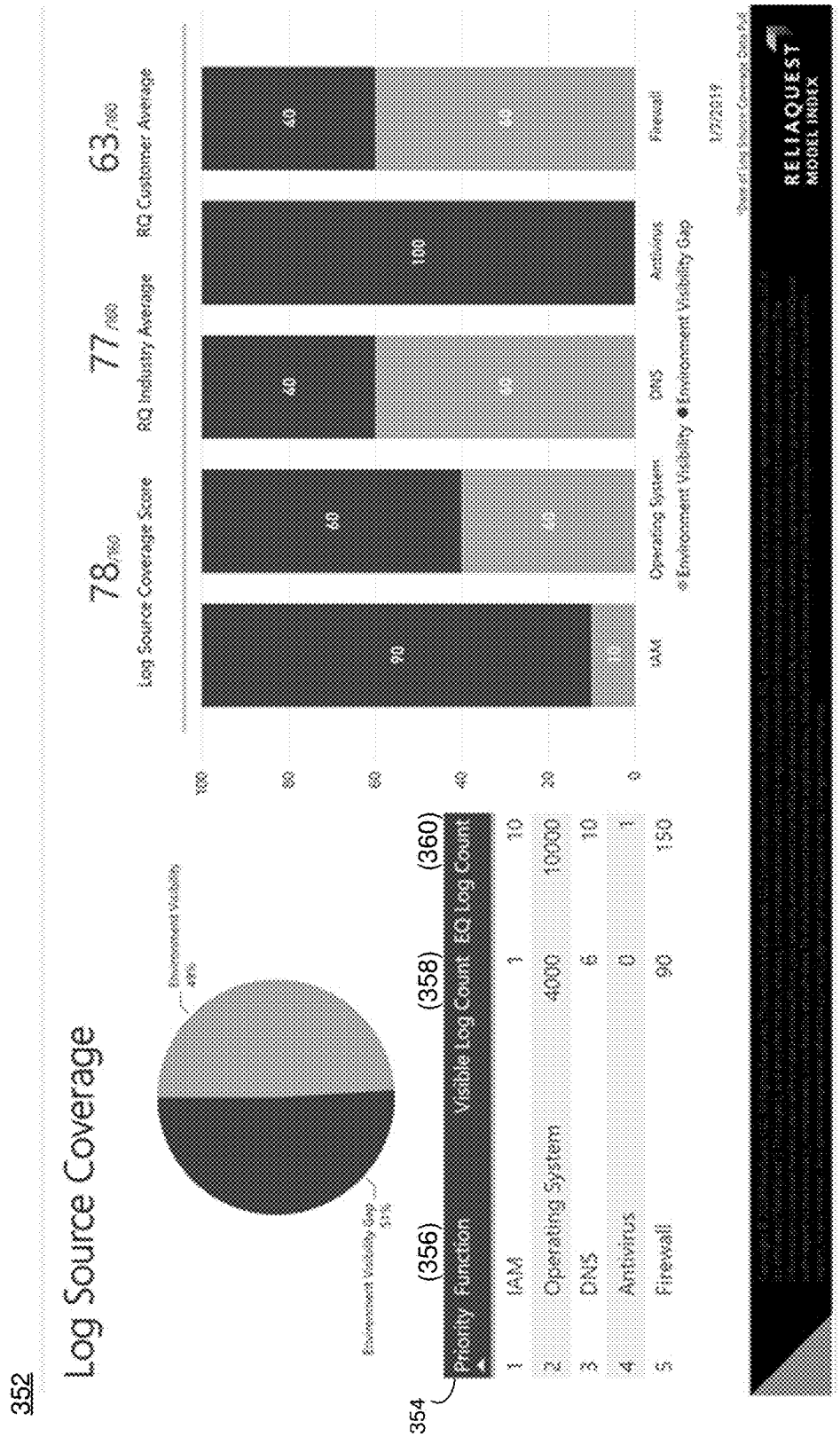

352
Log Source Coverage
Environment Visibility 40%
Environment Visibility Gap 60%
354
(356)
(358)
(360)
Priority
Function
Visible Log Count
EQ Log Count
1 IAM 1 10
2 Operating System 4000 10000
3 DNS 6 10
4 Antivirus 0 1
5 Firewall 90 150
78/100
Log Source Coverage Score
77/100
RQ Industry Average
63/100
RQ Customer Average
100
80
60
40
20
0
IAM
Operating System
DNS
Antivirus
Firewall
Environment Visibility
Environment Visibility Gap
1/7/2019
RELIAQUEST
MODEL INDEX

FIG. 6

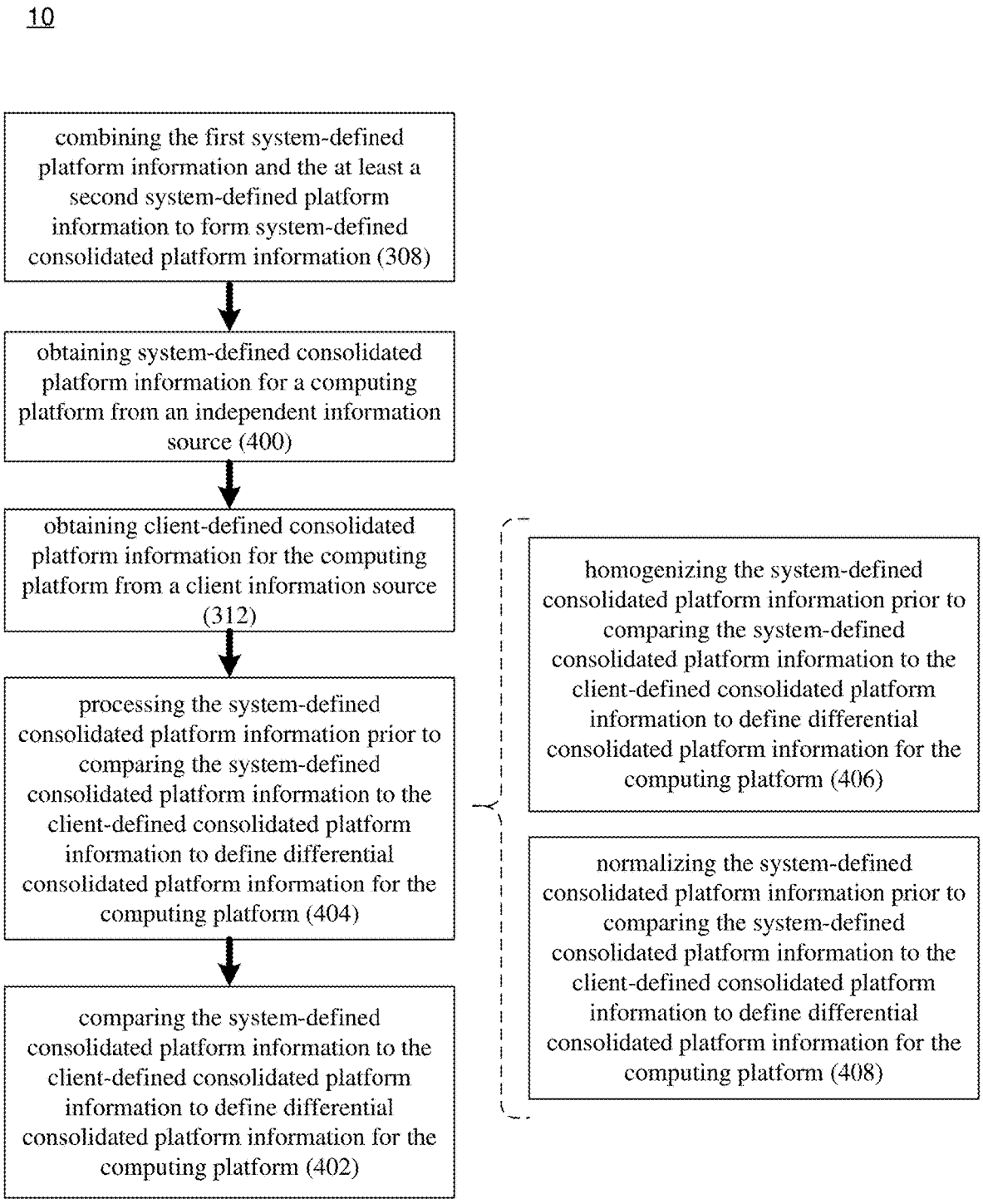

10
combining the first system-defined platform information and the at least a second system-defined platform information to form system-defined consolidated platform information (308)
obtaining system-defined consolidated platform information for a computing platform from an independent information source (400)
obtaining client-defined consolidated platform information for the computing platform from a client information source (312)
processing the system-defined consolidated platform information prior to comparing the system-defined consolidated platform information to the client-defined consolidated platform information to define differential consolidated platform information for the computing platform (404)
comparing the system-defined consolidated platform information to the client-defined consolidated platform information to define differential consolidated platform information for the computing platform (402)
homogenizing the system-defined consolidated platform information prior to comparing the system-defined consolidated platform information to the client-defined consolidated platform information to define differential consolidated platform information for the computing platform (406)
normalizing the system-defined consolidated platform information prior to comparing the system-defined consolidated platform information to the client-defined consolidated platform information to define differential consolidated platform information for the computing platform (408)

FIG. 7

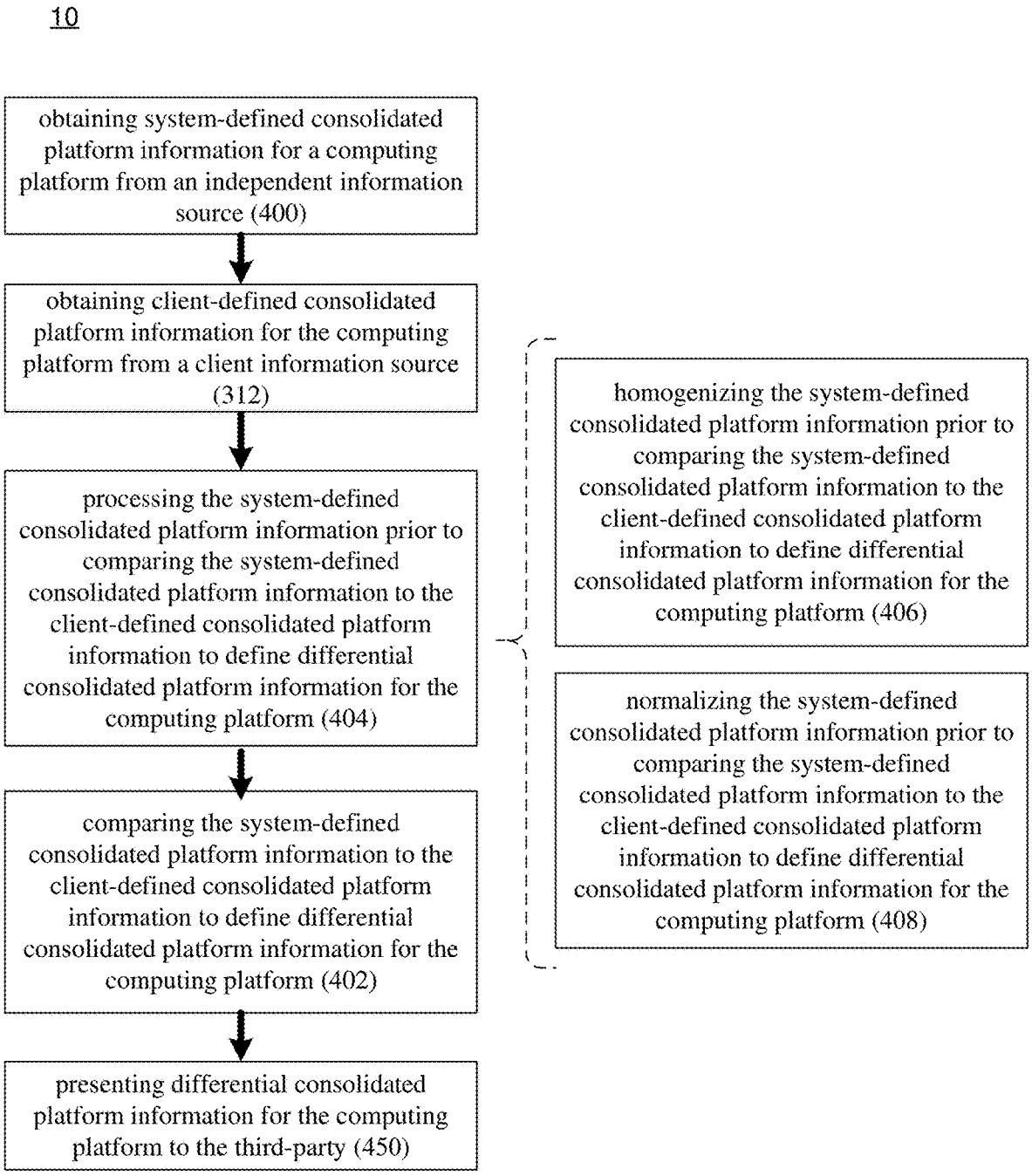

10
obtaining system-defined consolidated platform information for a computing platform from an independent information source (400)
obtaining client-defined consolidated platform information for the computing platform from a client information source (312)
processing the system-defined consolidated platform information prior to comparing the system-defined consolidated platform information to the client-defined consolidated platform information to define differential consolidated platform information for the computing platform (404)
comparing the system-defined consolidated platform information to the client-defined consolidated platform information to define differential consolidated platform information for the computing platform (402)
presenting differential consolidated platform information for the computing platform to the third-party (450)
homogenizing the system-defined consolidated platform information prior to comparing the system-defined consolidated platform information to the client-defined consolidated platform information to define differential consolidated platform information for the computing platform (406)
normalizing the system-defined consolidated platform information prior to comparing the system-defined consolidated platform information to the client-defined consolidated platform information to define differential consolidated platform information for the computing platform (408)

FIG. 8

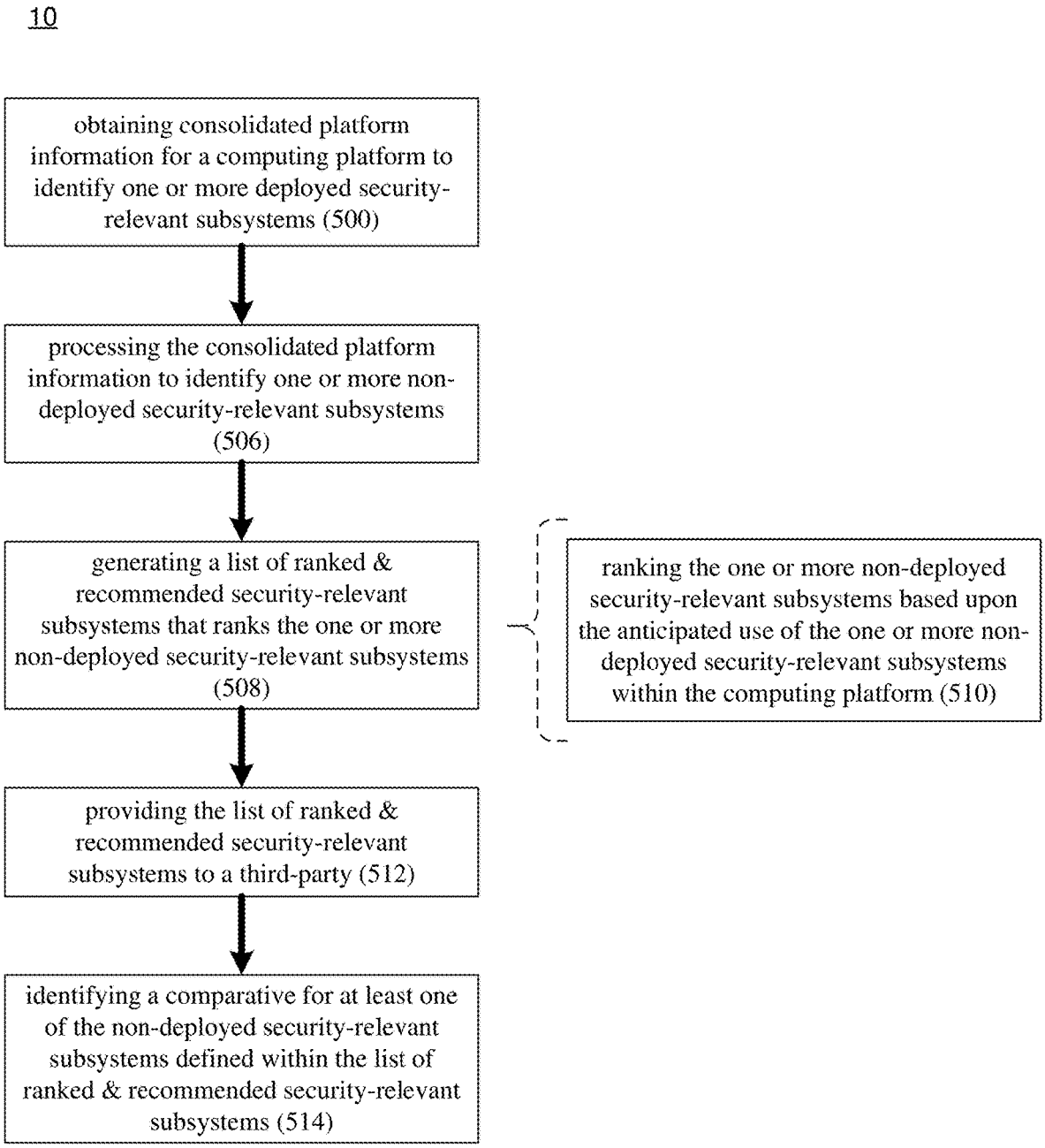

10
obtaining consolidated platform information for a computing platform to identify one or more deployed security-relevant subsystems (500)
processing the consolidated platform information to identify one or more non-deployed security-relevant subsystems (506)
generating a list of ranked & recommended security-relevant subsystems that ranks the one or more non-deployed security-relevant subsystems (508)
ranking the one or more non-deployed security-relevant subsystems based upon the anticipated use of the one or more non-deployed security-relevant subsystems within the computing platform (510)
providing the list of ranked & recommended security-relevant subsystems to a third-party (512)
identifying a comparative for at least one of the non-deployed security-relevant subsystems defined within the list of ranked & recommended security-relevant subsystems (514)

FIG. 9

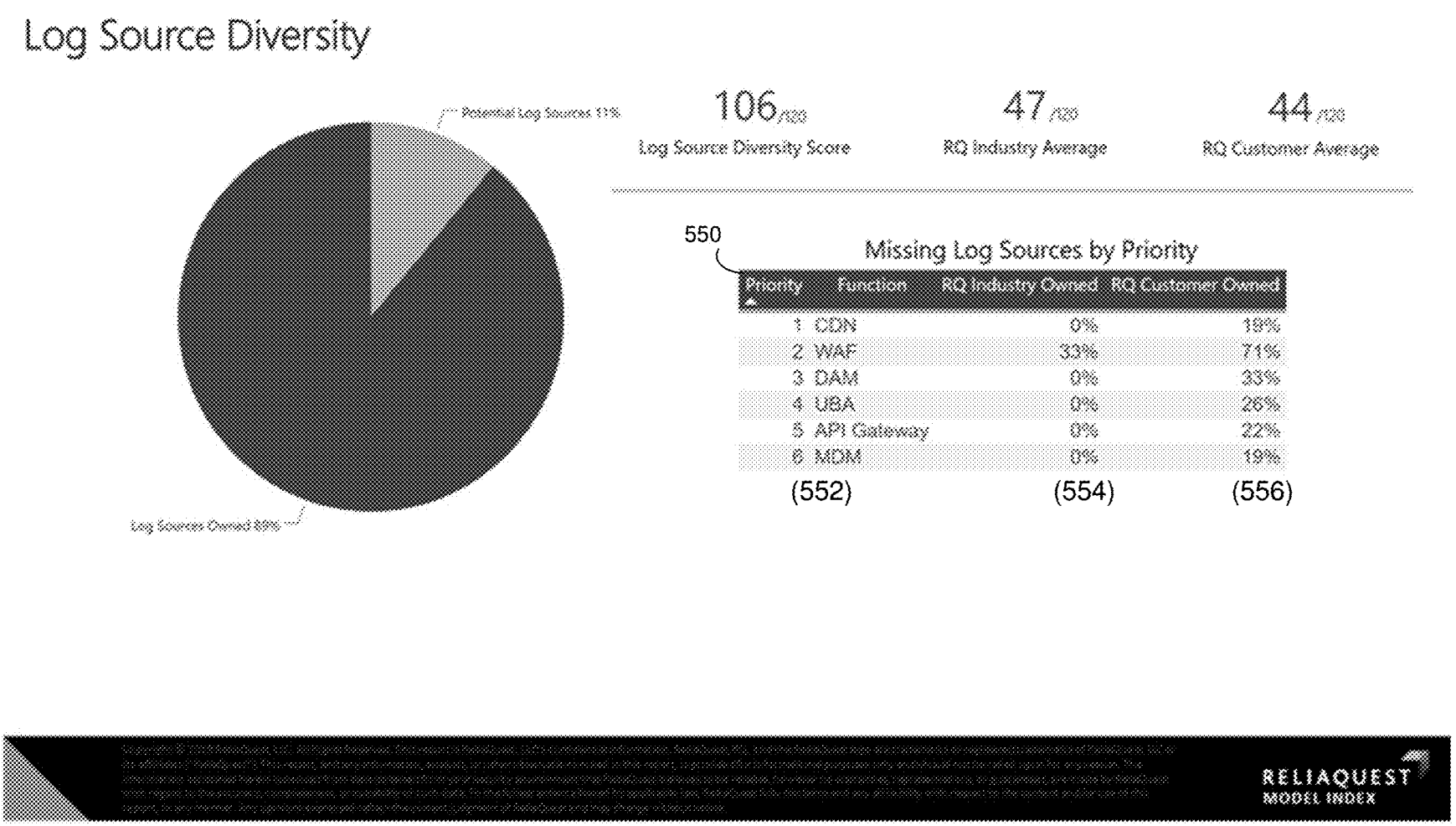

Log Source Diversity
Potential Log Sources 11%
Log Sources Owned 89%
106/100
Log Source Diversity Score
47/100
RQ Industry Average
44/100
RQ Customer Average
550
Missing Log Sources by Priority
Priority
Function
RQ Industry Owned
RQ Customer Owned
1 CDN 0% 18%
2 WAF 33% 71%
3 DAM 0% 33%
4 UBA 0% 26%
5 API Gateway 0% 22%
6 MDM 0% 19%
(552)
(554)
(556)
RELIAQUEST
MODEL INDEX

FIG. 10

10

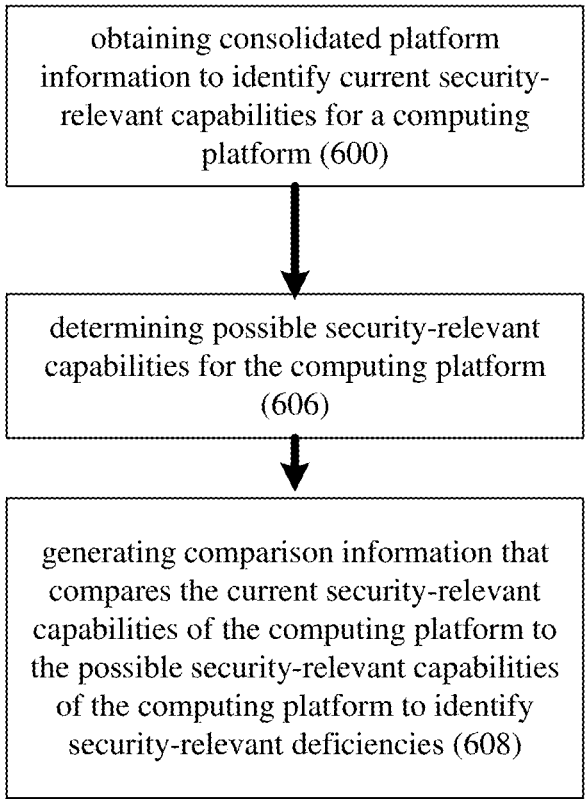

obtaining consolidated platform information to identify current security-relevant capabilities for a computing platform (600)
determining possible security-relevant capabilities for the computing platform (606)
generating comparison information that compares the current security-relevant capabilities of the computing platform to the possible security-relevant capabilities of the computing platform to identify security-relevant deficiencies (608)

FIG. 11

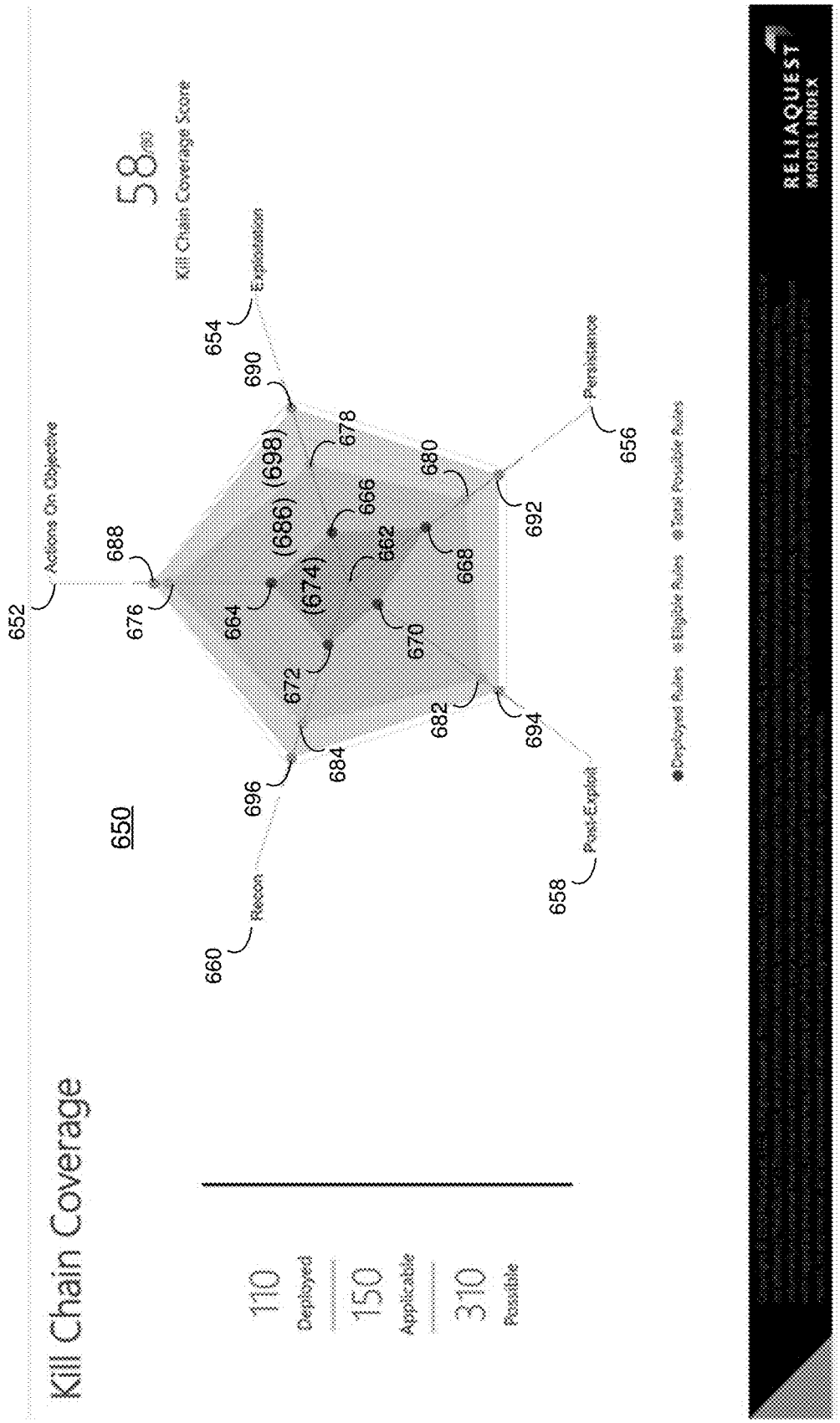

Kill Chain Coverage
650
58/100
Kill Chain Coverage Score
110
Deployed
150
Applicable
310
Possible
Actions On Objective
Exploitation
Persistence
Post-Exploit
Recon
652
654
656
658
660
662
664
666
668
670
672
674
676
678
680
682
684
686
688
690
692
694
696
698
Deployed Rules
Eligible Rules
Total Possible Rules
RELIAQUEST
MODEL INDEX

FIG. 12

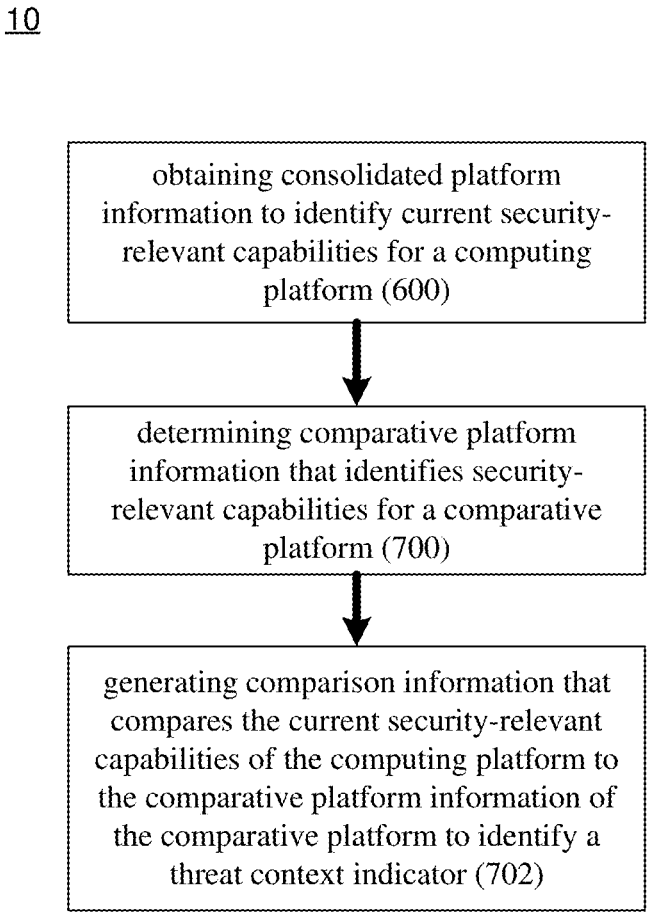

10
obtaining consolidated platform information to identify current security-relevant capabilities for a computing platform (600)
determining comparative platform information that identifies security-relevant capabilities for a comparative platform (700)
generating comparison information that compares the current security-relevant capabilities of the computing platform to the comparative platform information of the comparative platform to identify a threat context indicator (702)

FIG. 13

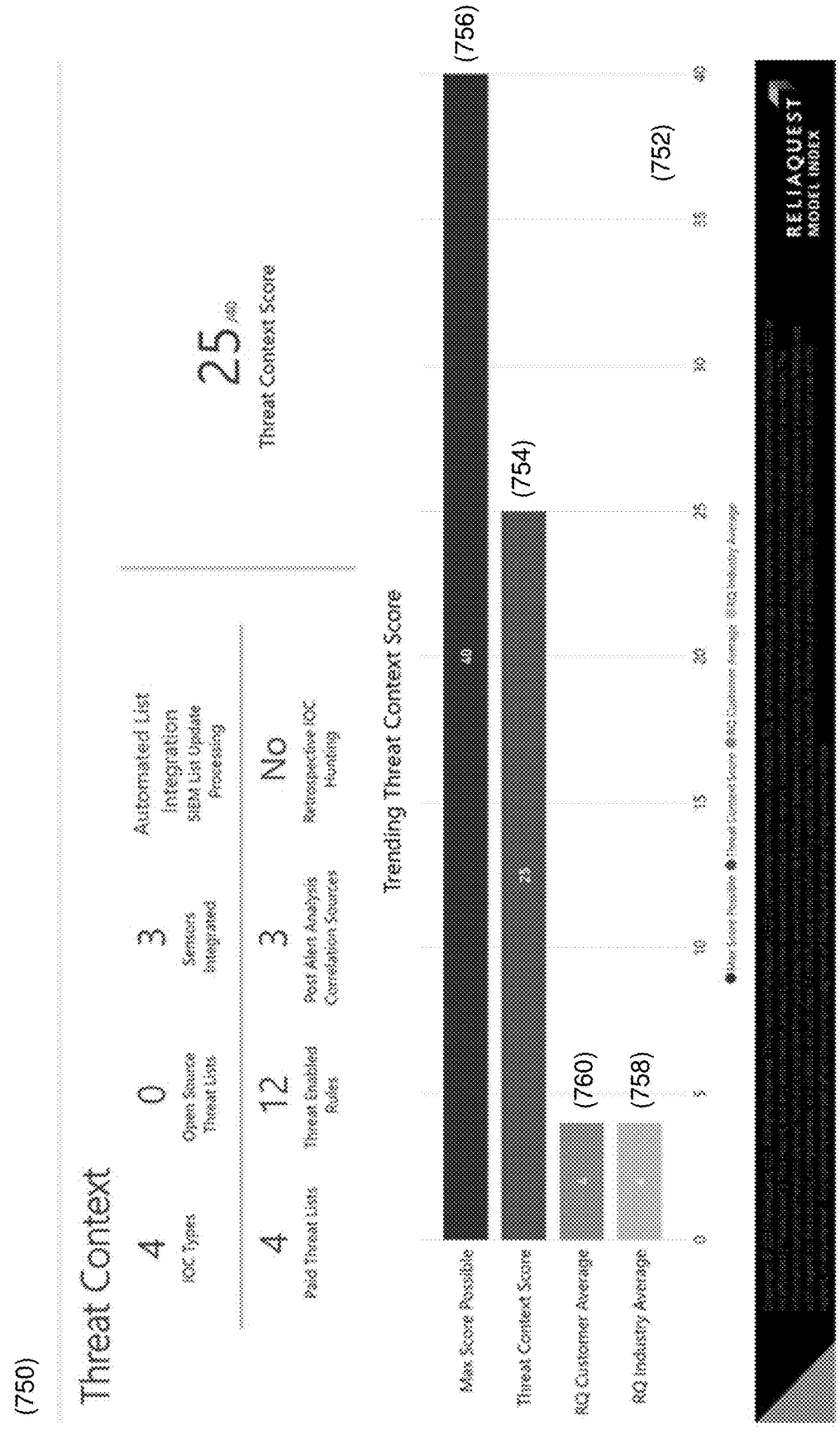

(750)
Threat Context
4
IOC Types
0
Open Source Threat Lists
3
Sensors Integrated
Automated List Integration SIEM List Update Processing
4
Paid Threat Lists
12
Threat Enabled Rules
3
Post Alert Analysis Correlation Sources
No
Retrospective IOC Hunting
25/40
Threat Context Score
Trending Threat Context Score
Max Score Possible
Threat Context Score
RQ Customer Average
RQ Industry Average
(760)
(758)
(754)
(752)
(756)
0
5
10
15
20
25
30
35
40
RELIAQUEST
MODEL INDEX

FIG. 14

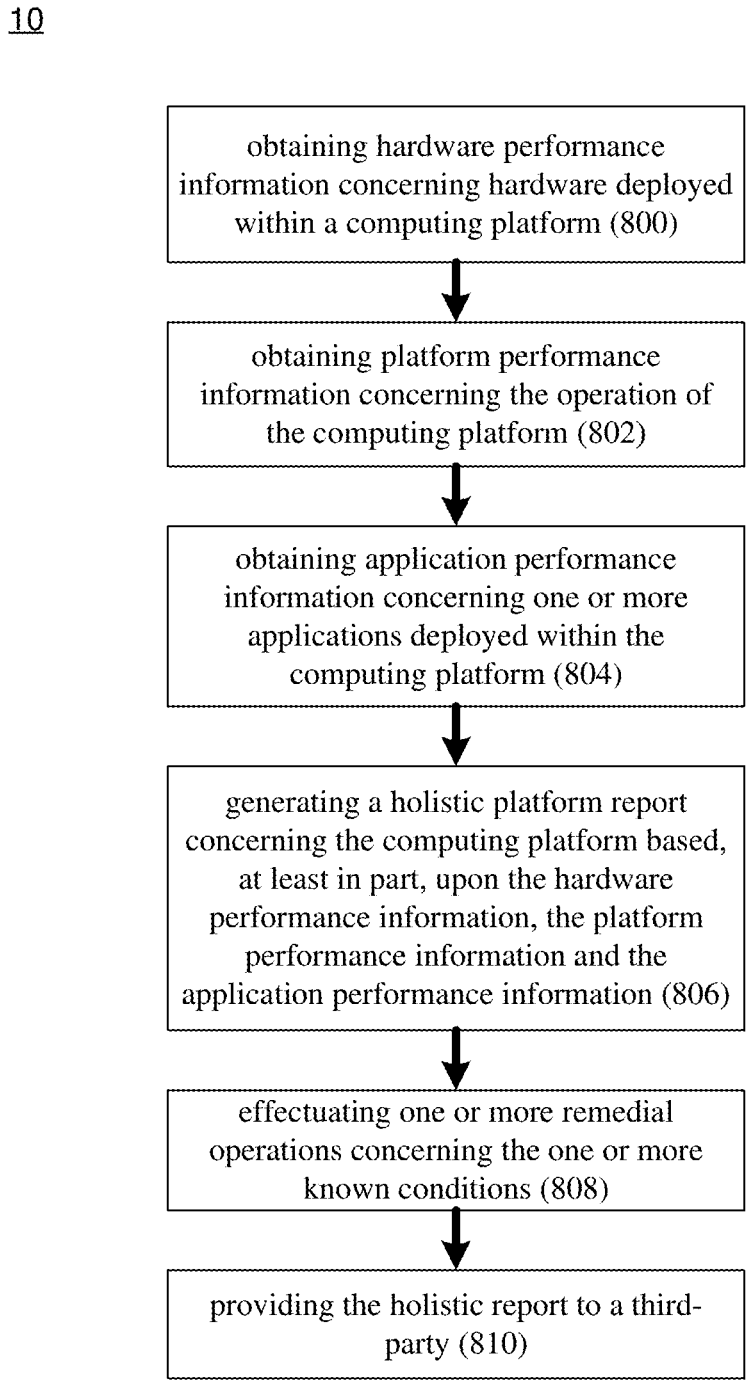
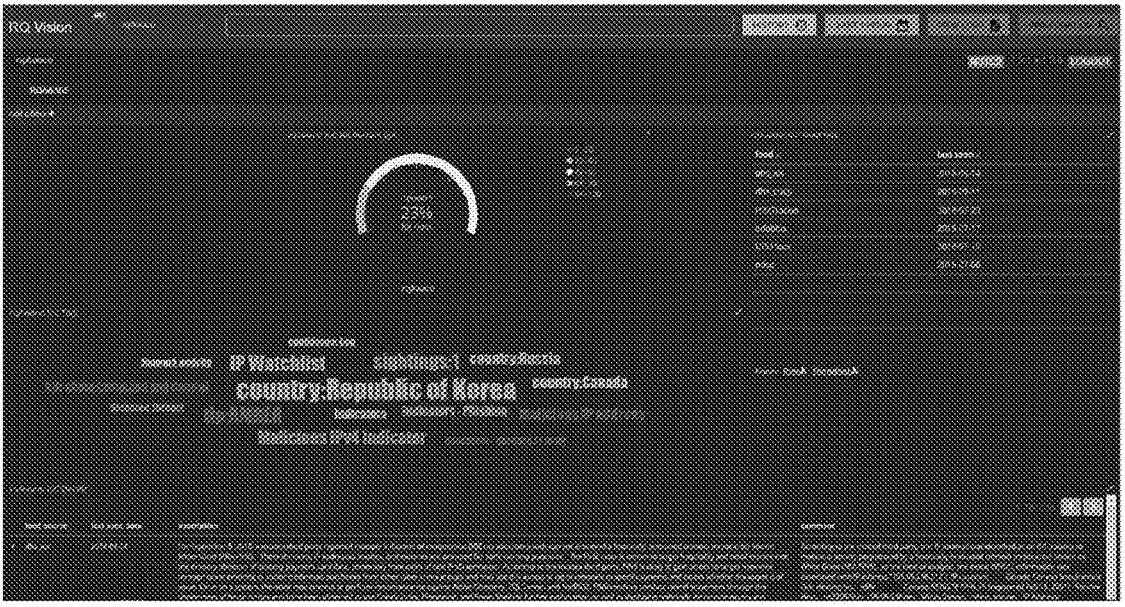

10
obtaining hardware performance information concerning hardware deployed within a computing platform (800)
obtaining platform performance information concerning the operation of the computing platform (802)
obtaining application performance information concerning one or more applications deployed within the computing platform (804)
generating a holistic platform report concerning the computing platform based, at least in part, upon the hardware performance information, the platform performance information and the application performance information (806)
effectuating one or more remedial operations concerning the one or more known conditions (808)
providing the holistic report to a third-party (810)

FIG. 15

352



352

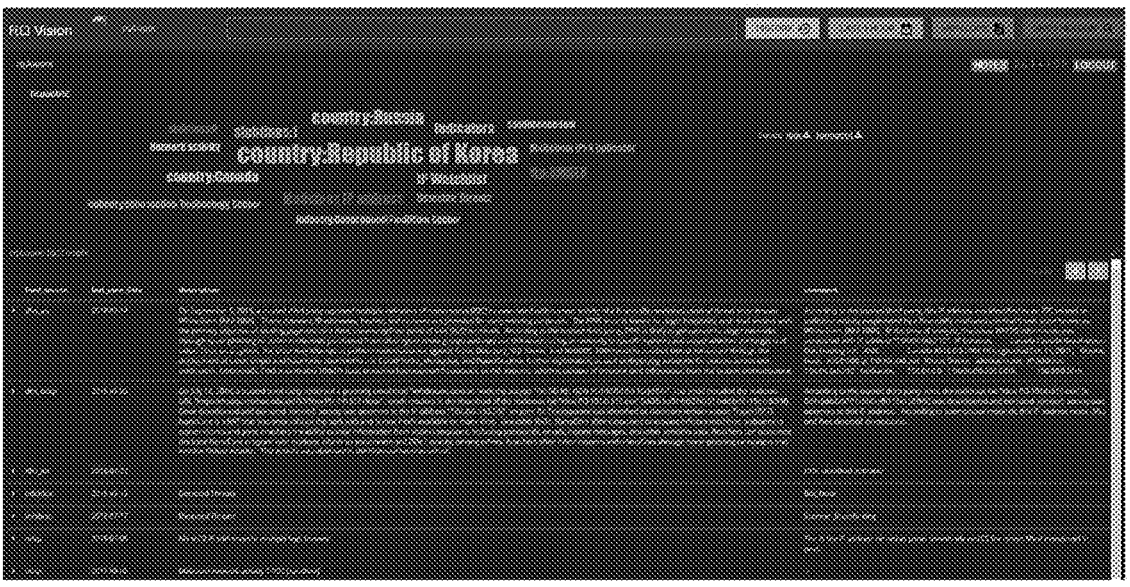


FIG. 16

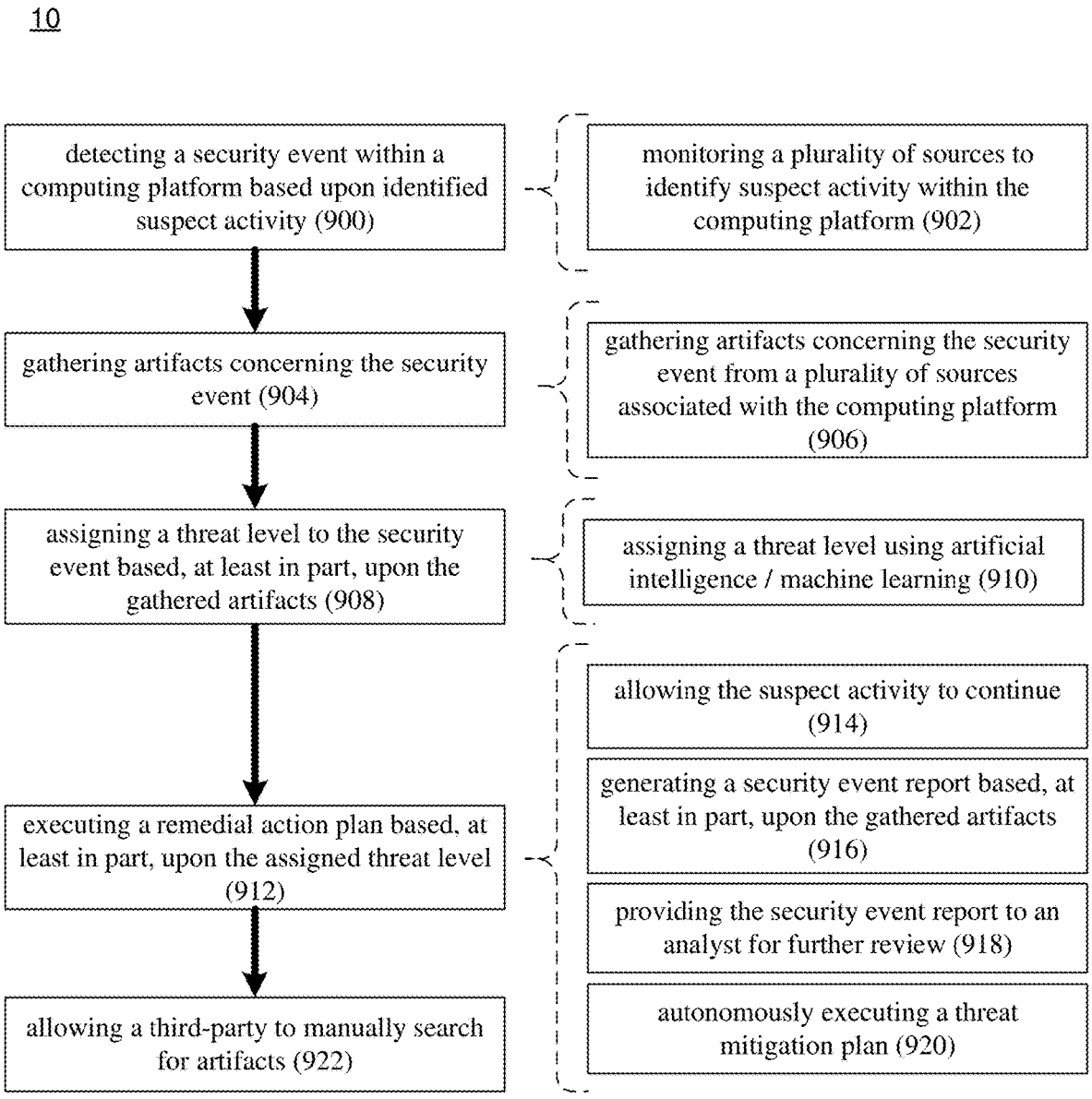

10
detecting a security event within a computing platform based upon identified suspect activity (900)
monitoring a plurality of sources to identify suspect activity within the computing platform (902)
gathering artifacts concerning the security event (904)
gathering artifacts concerning the security event from a plurality of sources associated with the computing platform (906)
assigning a threat level to the security event based, at least in part, upon the gathered artifacts (908)
assigning a threat level using artificial intelligence / machine learning (910)
executing a remedial action plan based, at least in part, upon the assigned threat level (912)
allowing the suspect activity to continue (914)
generating a security event report based, at least in part, upon the gathered artifacts (916)
providing the security event report to an analyst for further review (918)
autonomously executing a threat mitigation plan (920)
allowing a third-party to manually search for artifacts (922)

FIG. 17

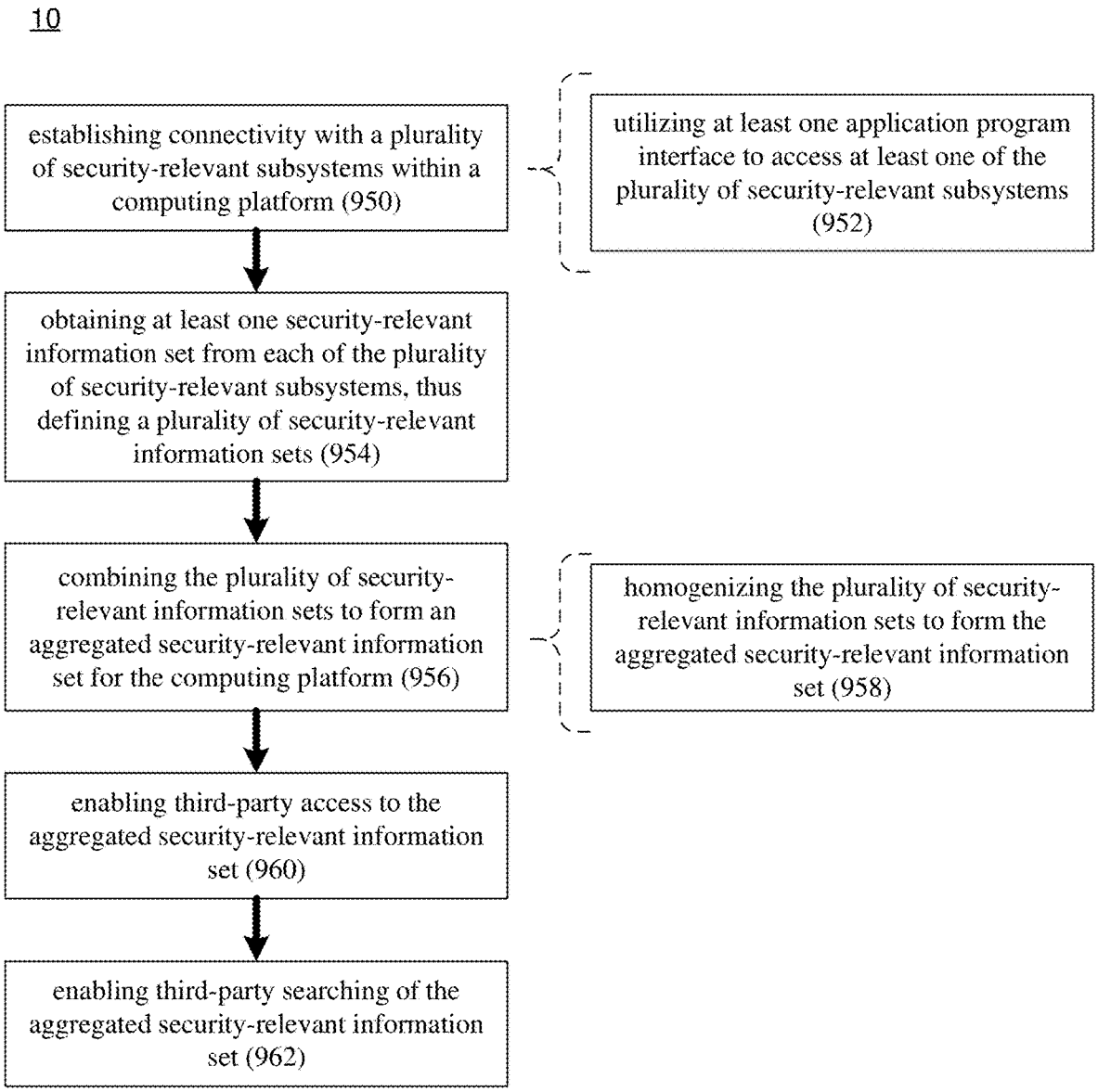

10
establishing connectivity with a plurality of security-relevant subsystems within a computing platform (950)
utilizing at least one application program interface to access at least one of the plurality of security-relevant subsystems (952)
obtaining at least one security-relevant information set from each of the plurality of security-relevant subsystems, thus defining a plurality of security-relevant information sets (954)
combining the plurality of security-relevant information sets to form an aggregated security-relevant information set for the computing platform (956)
homogenizing the plurality of security-relevant information sets to form the aggregated security-relevant information set (958)
enabling third-party access to the aggregated security-relevant information set (960)
enabling third-party searching of the aggregated security-relevant information set (962)

FIG. 18

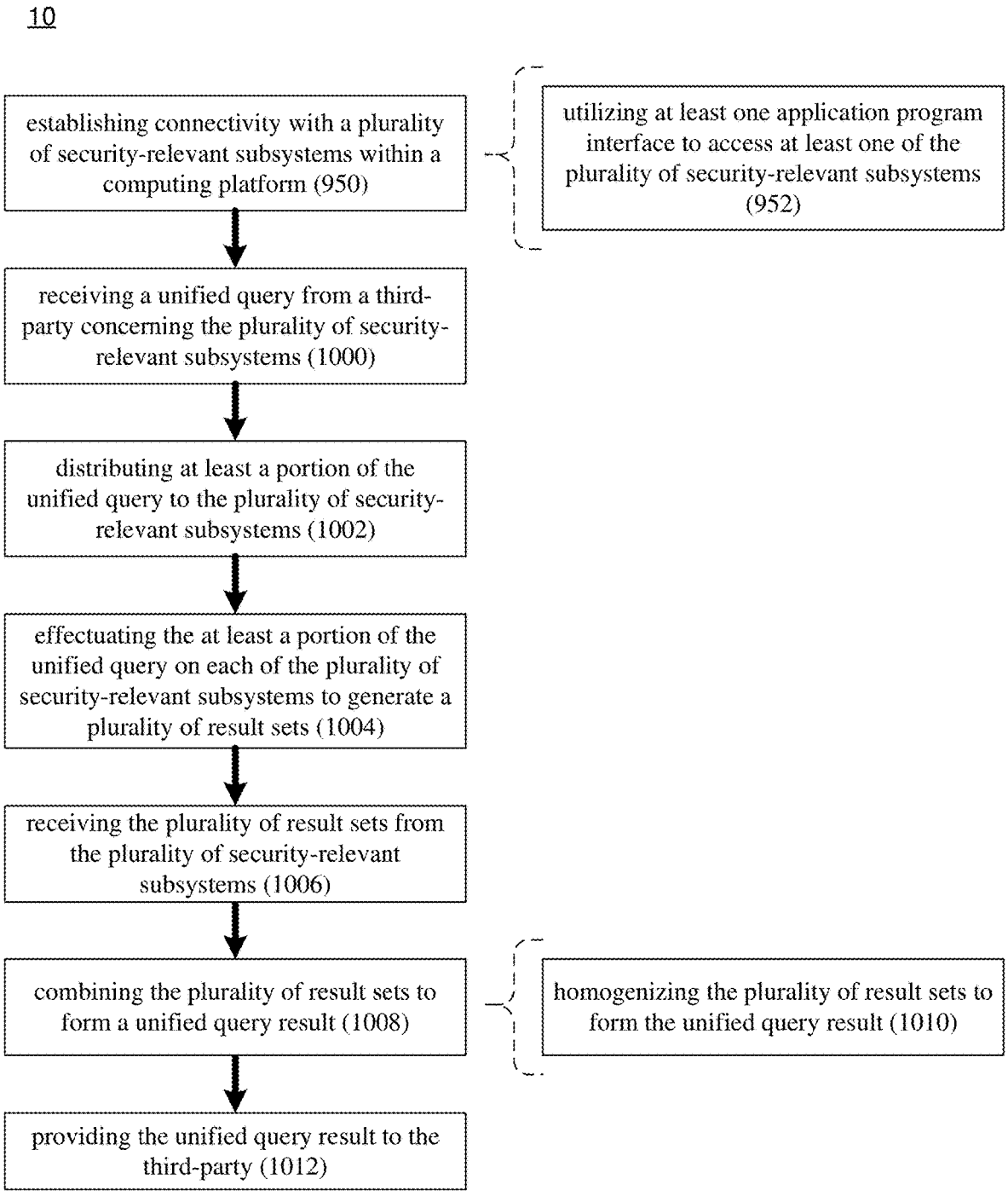

10
establishing connectivity with a plurality of security-relevant subsystems within a computing platform (950)
utilizing at least one application program interface to access at least one of the plurality of security-relevant subsystems (952)
receiving a unified query from a third-party concerning the plurality of security-relevant subsystems (1000)
distributing at least a portion of the unified query to the plurality of security-relevant subsystems (1002)
effectuating the at least a portion of the unified query on each of the plurality of security-relevant subsystems to generate a plurality of result sets (1004)
receiving the plurality of result sets from the plurality of security-relevant subsystems (1006)
combining the plurality of result sets to form a unified query result (1008)
homogenizing the plurality of result sets to form the unified query result (1010)
providing the unified query result to the third-party (1012)

FIG. 19

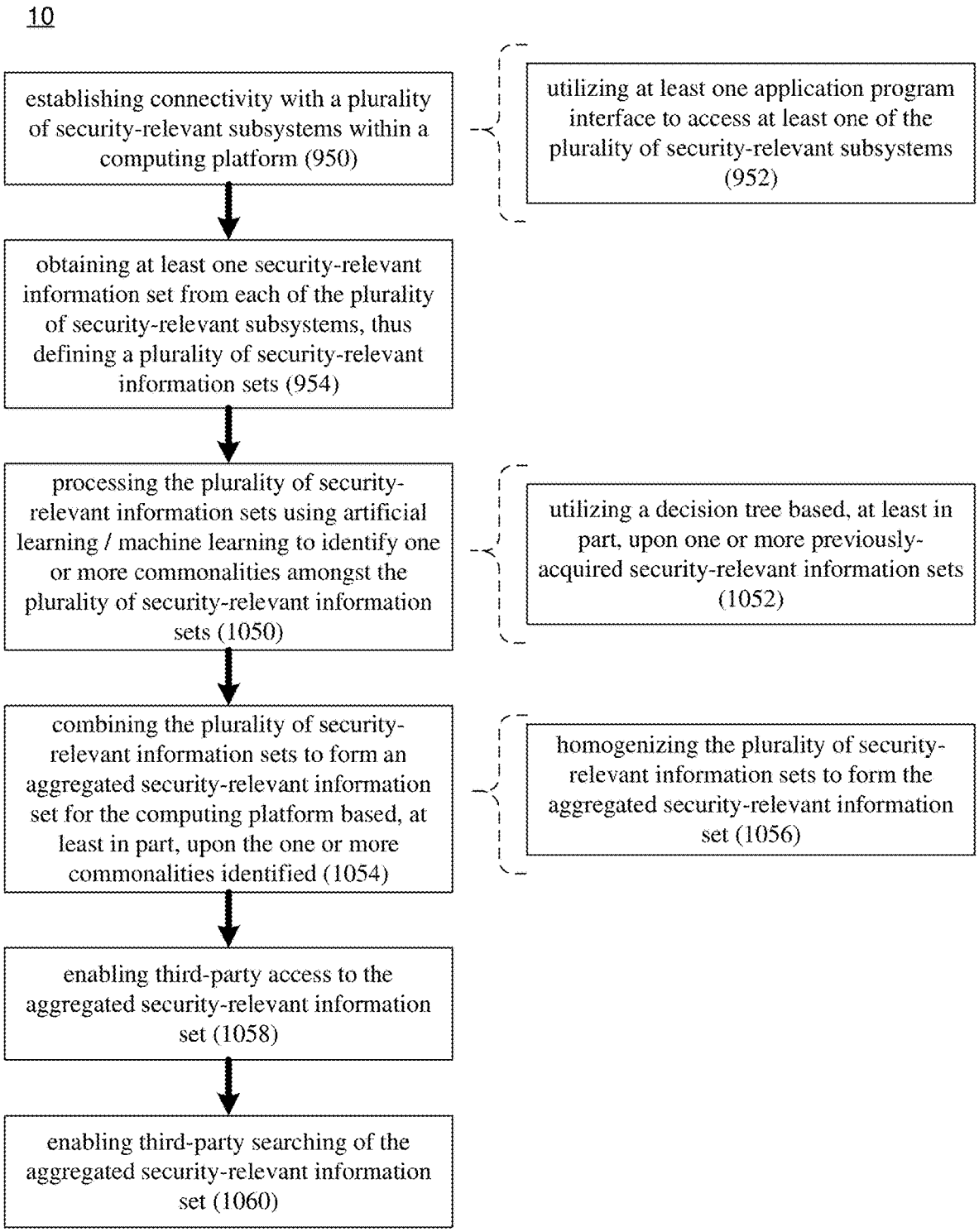

10
establishing connectivity with a plurality of security-relevant subsystems within a computing platform (950)
utilizing at least one application program interface to access at least one of the plurality of security-relevant subsystems (952)
obtaining at least one security-relevant information set from each of the plurality of security-relevant subsystems, thus defining a plurality of security-relevant information sets (954)
processing the plurality of security-relevant information sets using artificial learning / machine learning to identify one or more commonalities amongst the plurality of security-relevant information sets (1050)
utilizing a decision tree based, at least in part, upon one or more previously-acquired security-relevant information sets (1052)
combining the plurality of security-relevant information sets to form an aggregated security-relevant information set for the computing platform based, at least in part, upon the one or more commonalities identified (1054)
homogenizing the plurality of security-relevant information sets to form the aggregated security-relevant information set (1056)
enabling third-party access to the aggregated security-relevant information set (1058)
enabling third-party searching of the aggregated security-relevant information set (1060)

FIG. 20

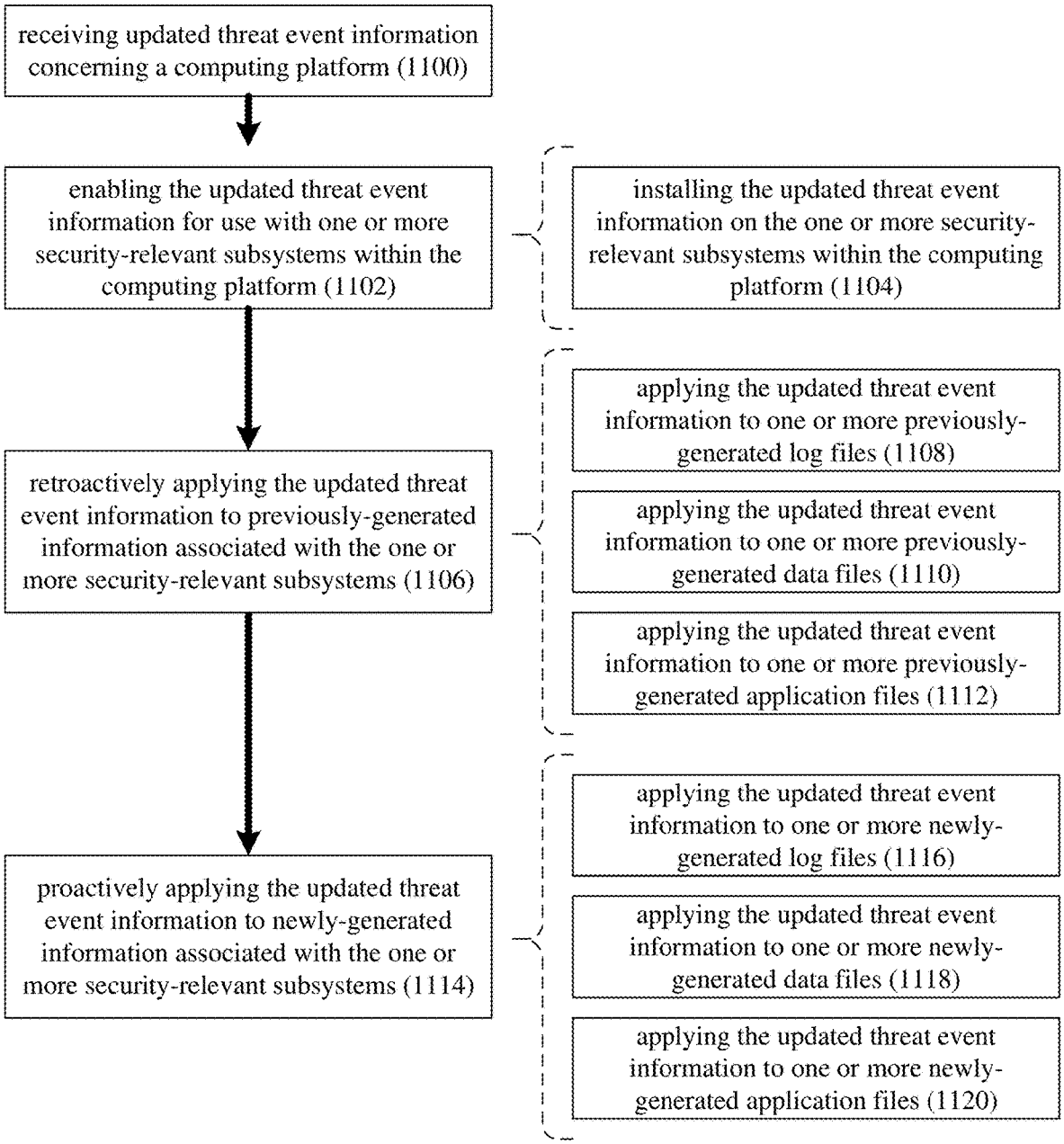

10
receiving updated threat event information concerning a computing platform (1100)
enabling the updated threat event information for use with one or more security-relevant subsystems within the computing platform (1102)
installing the updated threat event information on the one or more security-relevant subsystems within the computing platform (1104)
retroactively applying the updated threat event information to previously-generated information associated with the one or more security-relevant subsystems (1106)
applying the updated threat event information to one or more previously-generated log files (1108)
applying the updated threat event information to one or more previously-generated data files (1110)
applying the updated threat event information to one or more previously-generated application files (1112)
proactively applying the updated threat event information to newly-generated information associated with the one or more security-relevant subsystems (1114)
applying the updated threat event information to one or more newly-generated log files (1116)
applying the updated threat event information to one or more newly-generated data files (1118)
applying the updated threat event information to one or more newly-generated application files (1120)

FIG. 21

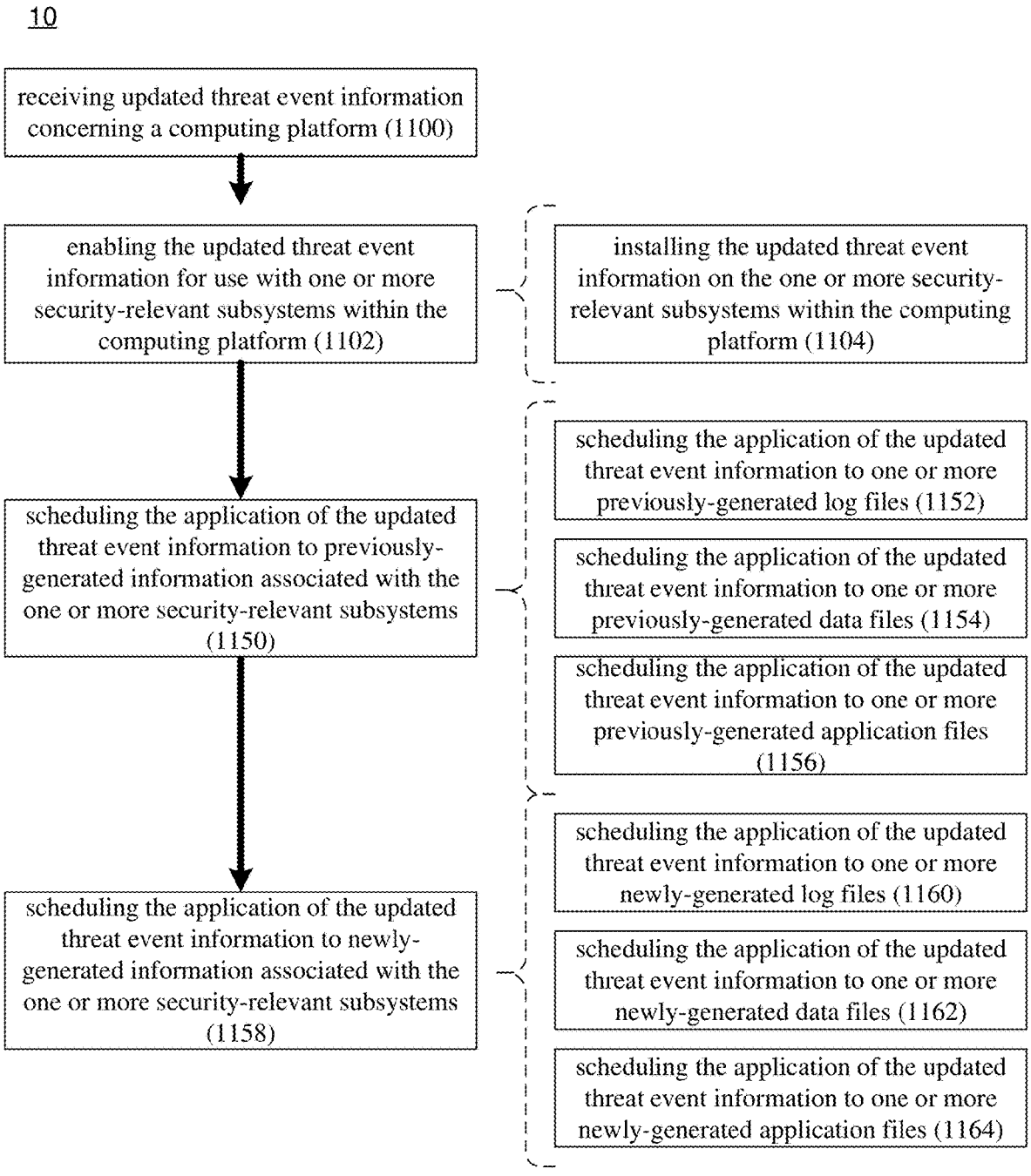

10
receiving updated threat event information concerning a computing platform (1100)
enabling the updated threat event information for use with one or more security-relevant subsystems within the computing platform (1102)
installing the updated threat event information on the one or more security-relevant subsystems within the computing platform (1104)
scheduling the application of the updated threat event information to previously-generated information associated with the one or more security-relevant subsystems (1150)
scheduling the application of the updated threat event information to one or more previously-generated log files (1152)
scheduling the application of the updated threat event information to one or more previously-generated data files (1154)
scheduling the application of the updated threat event information to one or more previously-generated application files (1156)
scheduling the application of the updated threat event information to newly-generated information associated with the one or more security-relevant subsystems (1158)
scheduling the application of the updated threat event information to one or more newly-generated log files (1160)
scheduling the application of the updated threat event information to one or more newly-generated data files (1162)
scheduling the application of the updated threat event information to one or more newly-generated application files (1164)

FIG. 22

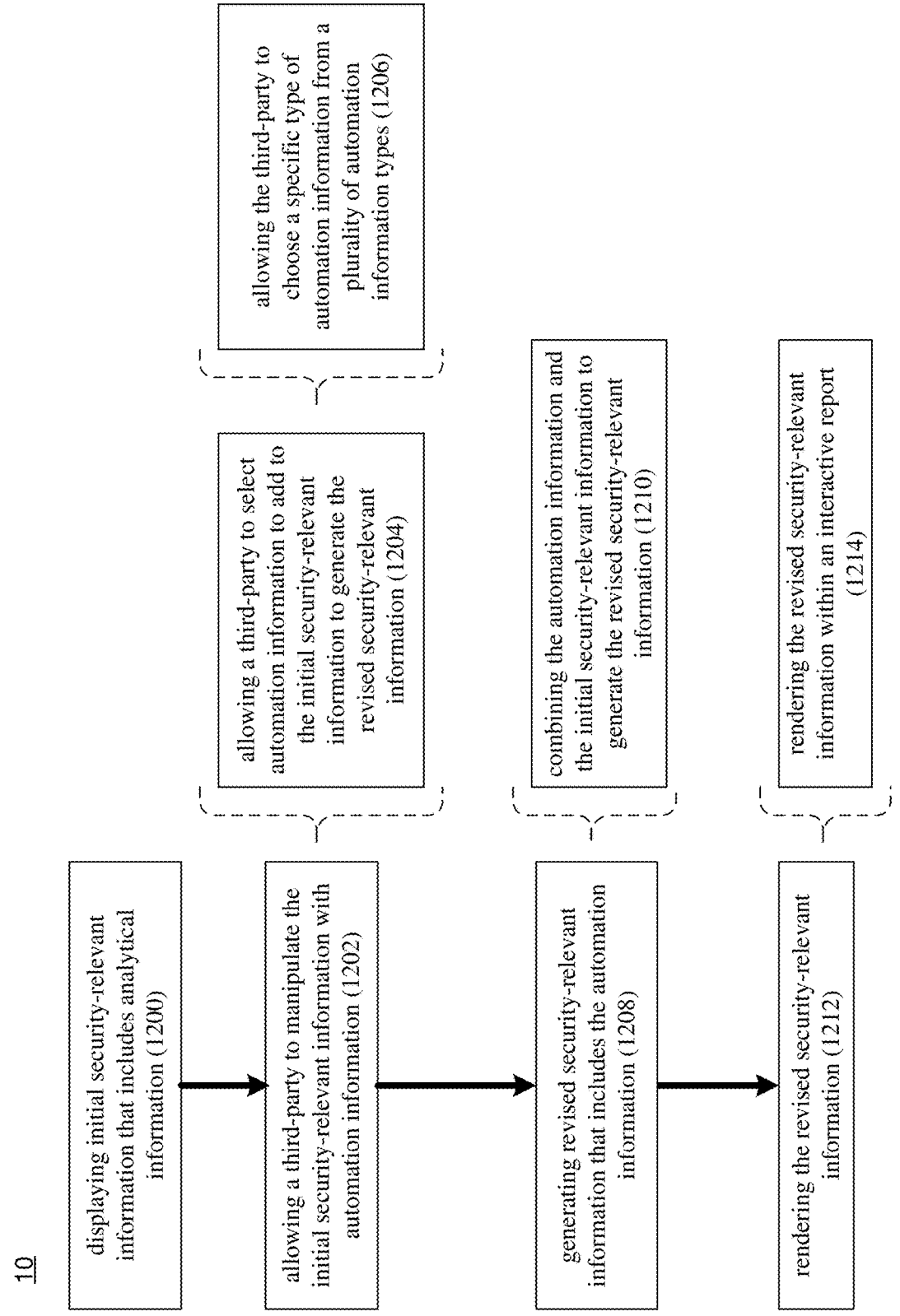

10
displaying initial security-relevant information that includes analytical information (1200)
allowing a third-party to manipulate the initial security-relevant information with automation information (1202)
allowing a third-party to select automation information to add to the initial security-relevant information to generate the revised security-relevant information (1204)
allowing the third-party to choose a specific type of automation information from a plurality of automation information types (1206)
generating revised security-relevant information that includes the automation information (1208)
combining the automation information and the initial security-relevant information to generate the revised security-relevant information (1210)
rendering the revised security-relevant information (1212)
rendering the revised security-relevant information within an interactive report (1214)

FIG. 23

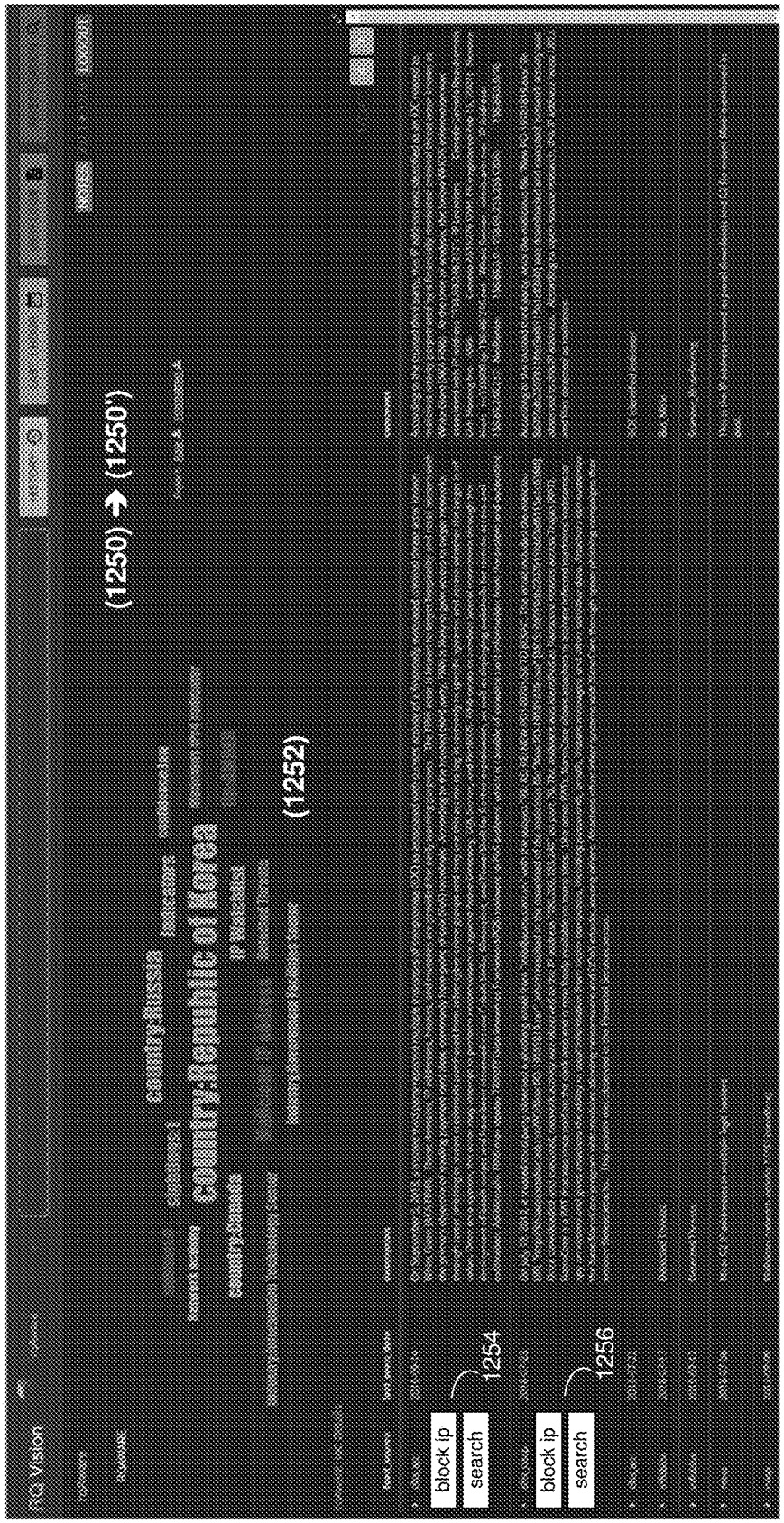

1258
(1250) → (1250')
(1252)
block ip
search
1254
block ip
search
1256

FIG. 24

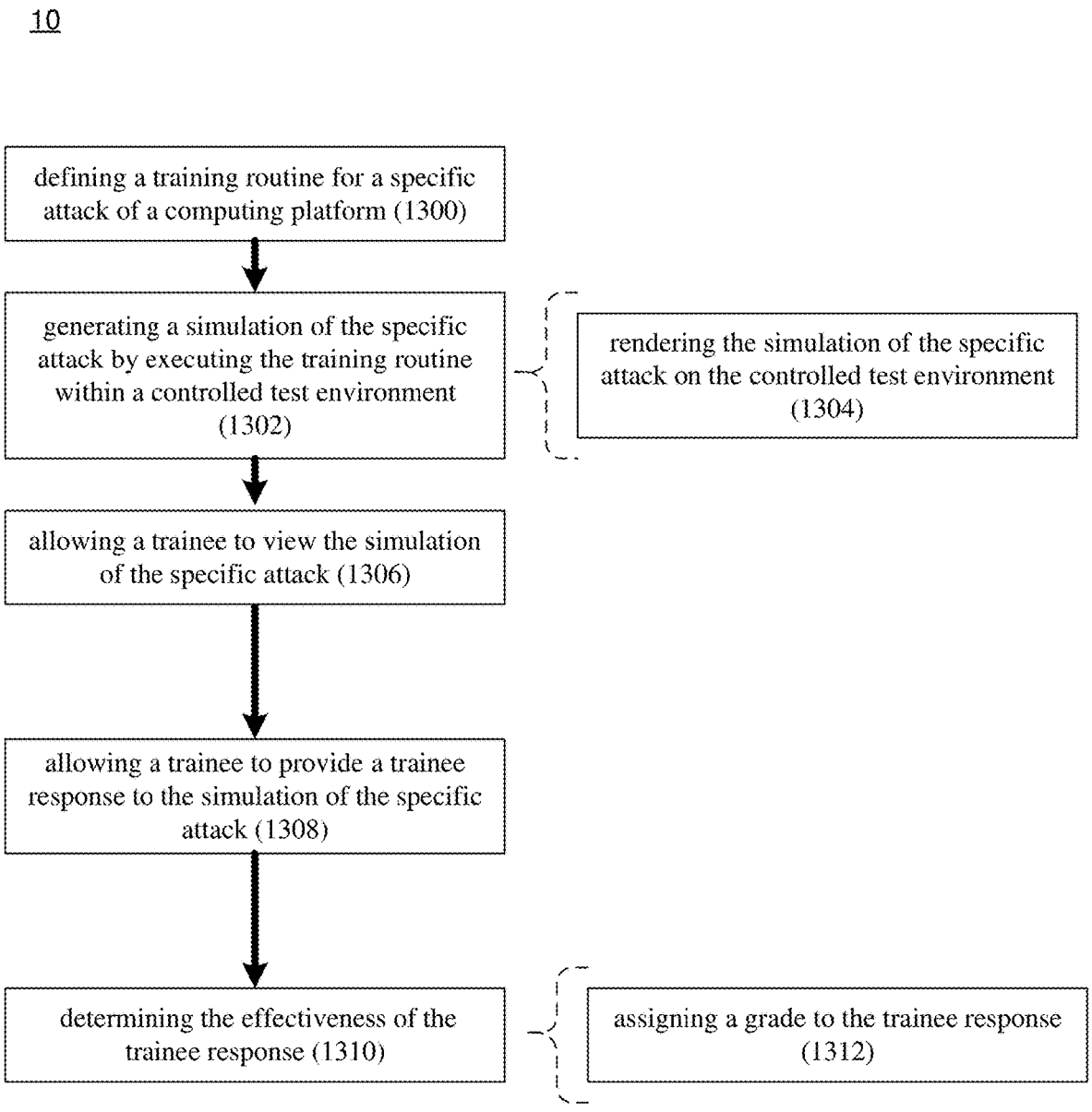

10
defining a training routine for a specific attack of a computing platform (1300)
generating a simulation of the specific attack by executing the training routine within a controlled test environment (1302)
rendering the simulation of the specific attack on the controlled test environment (1304)
allowing a trainee to view the simulation of the specific attack (1306)
allowing a trainee to provide a trainee response to the simulation of the specific attack (1308)
determining the effectiveness of the trainee response (1310)
assigning a grade to the trainee response (1312)

FIG. 25

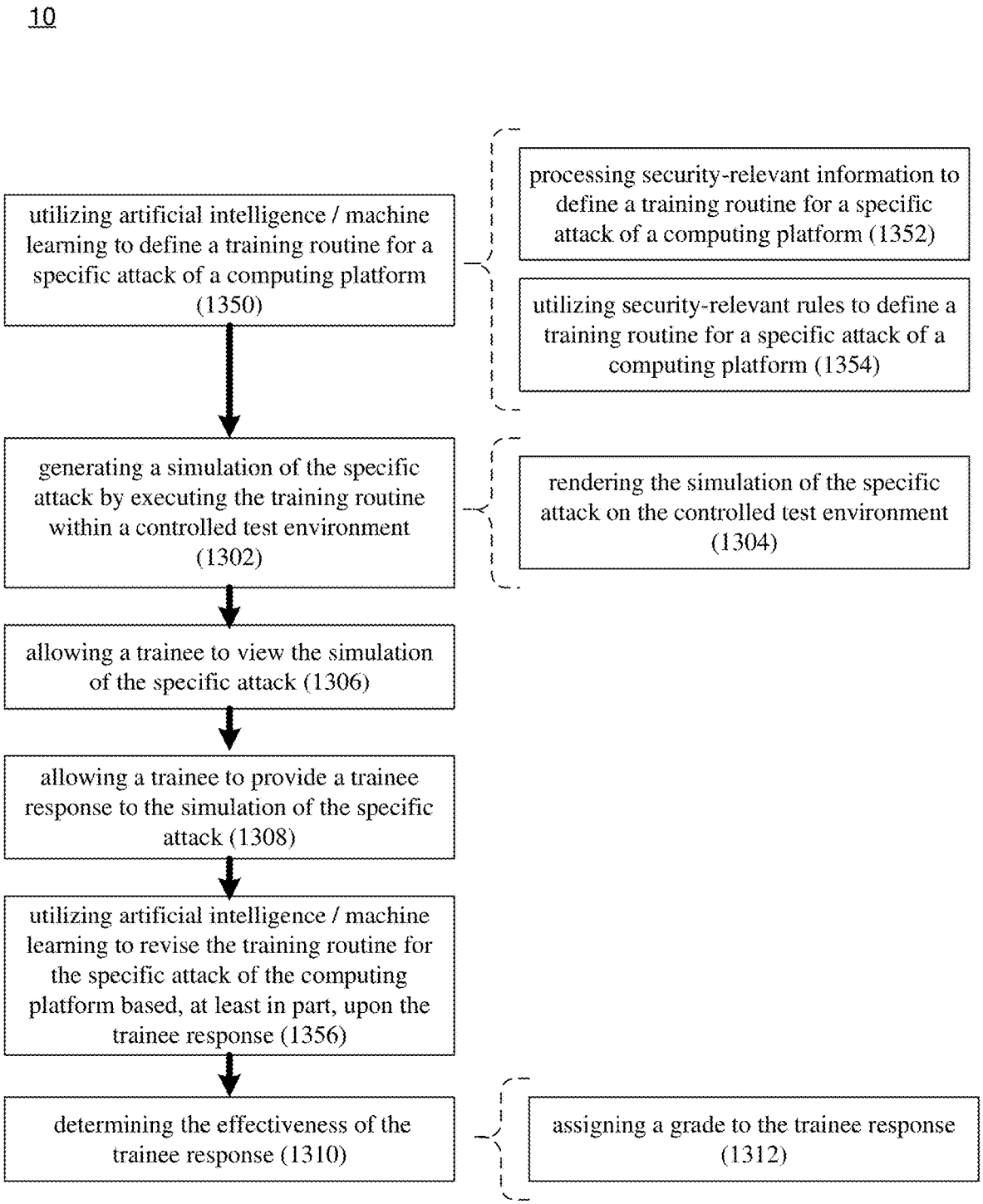

10
utilizing artificial intelligence / machine learning to define a training routine for a specific attack of a computing platform (1350)
processing security-relevant information to define a training routine for a specific attack of a computing platform (1352)
utilizing security-relevant rules to define a training routine for a specific attack of a computing platform (1354)
generating a simulation of the specific attack by executing the training routine within a controlled test environment (1302)
rendering the simulation of the specific attack on the controlled test environment (1304)
allowing a trainee to view the simulation of the specific attack (1306)
allowing a trainee to provide a trainee response to the simulation of the specific attack (1308)
utilizing artificial intelligence / machine learning to revise the training routine for the specific attack of the computing platform based, at least in part, upon the trainee response (1356)
determining the effectiveness of the trainee response (1310)
assigning a grade to the trainee response (1312)

FIG. 26

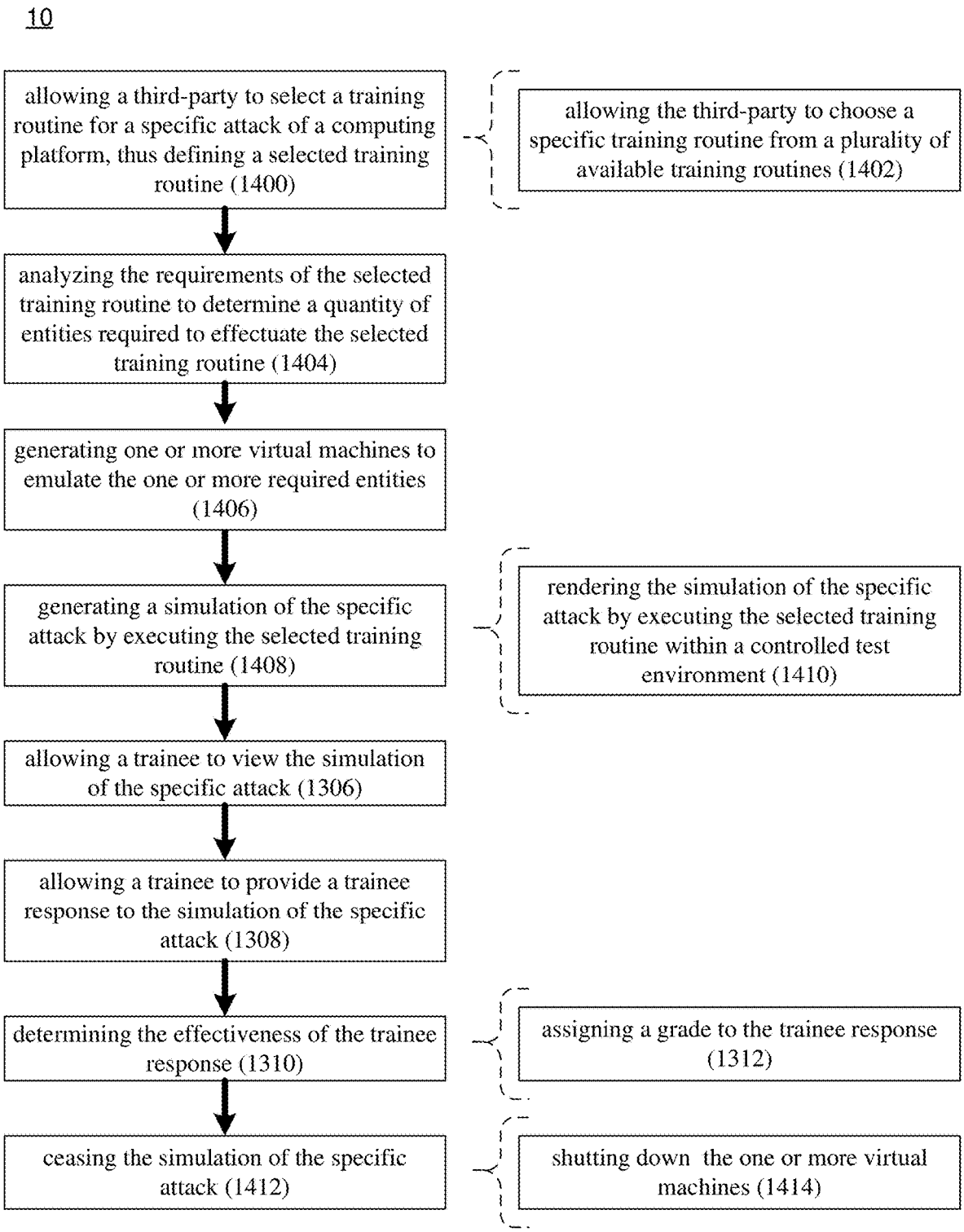

10
allowing a third-party to select a training routine for a specific attack of a computing platform, thus defining a selected training routine (1400)
allowing the third-party to choose a specific training routine from a plurality of available training routines (1402)
analyzing the requirements of the selected training routine to determine a quantity of entities required to effectuate the selected training routine (1404)
generating one or more virtual machines to emulate the one or more required entities (1406)
generating a simulation of the specific attack by executing the selected training routine (1408)
rendering the simulation of the specific attack by executing the selected training routine within a controlled test environment (1410)
allowing a trainee to view the simulation of the specific attack (1306)
allowing a trainee to provide a trainee response to the simulation of the specific attack (1308)
determining the effectiveness of the trainee response (1310)
assigning a grade to the trainee response (1312)
ceasing the simulation of the specific attack (1412)
shutting down the one or more virtual machines (1414)

FIG. 27

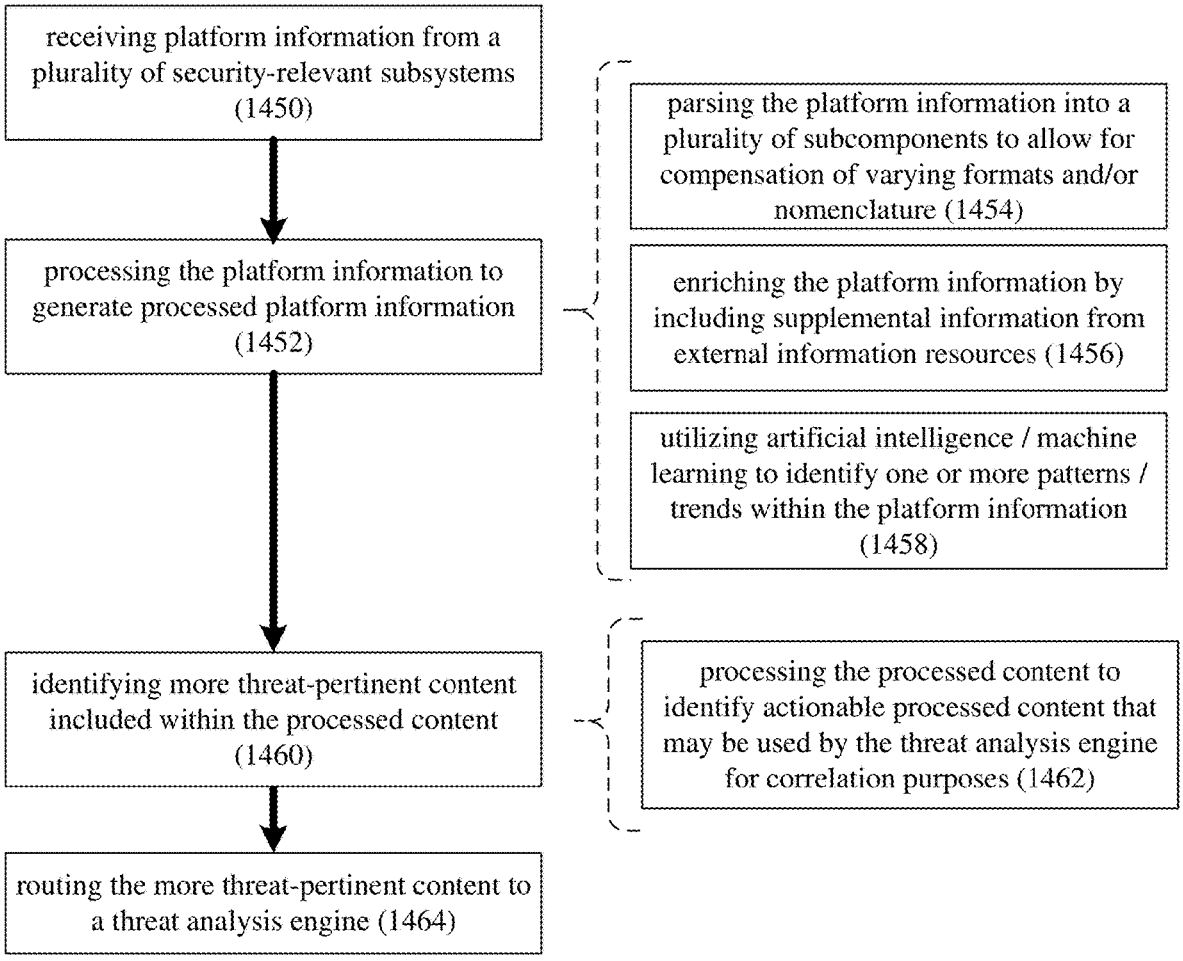

10
receiving platform information from a plurality of security-relevant subsystems (1450)
processing the platform information to generate processed platform information (1452)
identifying more threat-pertinent content included within the processed content (1460)
routing the more threat-pertinent content to a threat analysis engine (1464)
parsing the platform information into a plurality of subcomponents to allow for compensation of varying formats and/or nomenclature (1454)
enriching the platform information by including supplemental information from external information resources (1456)
utilizing artificial intelligence / machine learning to identify one or more patterns / trends within the platform information (1458)
processing the processed content to identify actionable processed content that may be used by the threat analysis engine for correlation purposes (1462)

FIG. 28

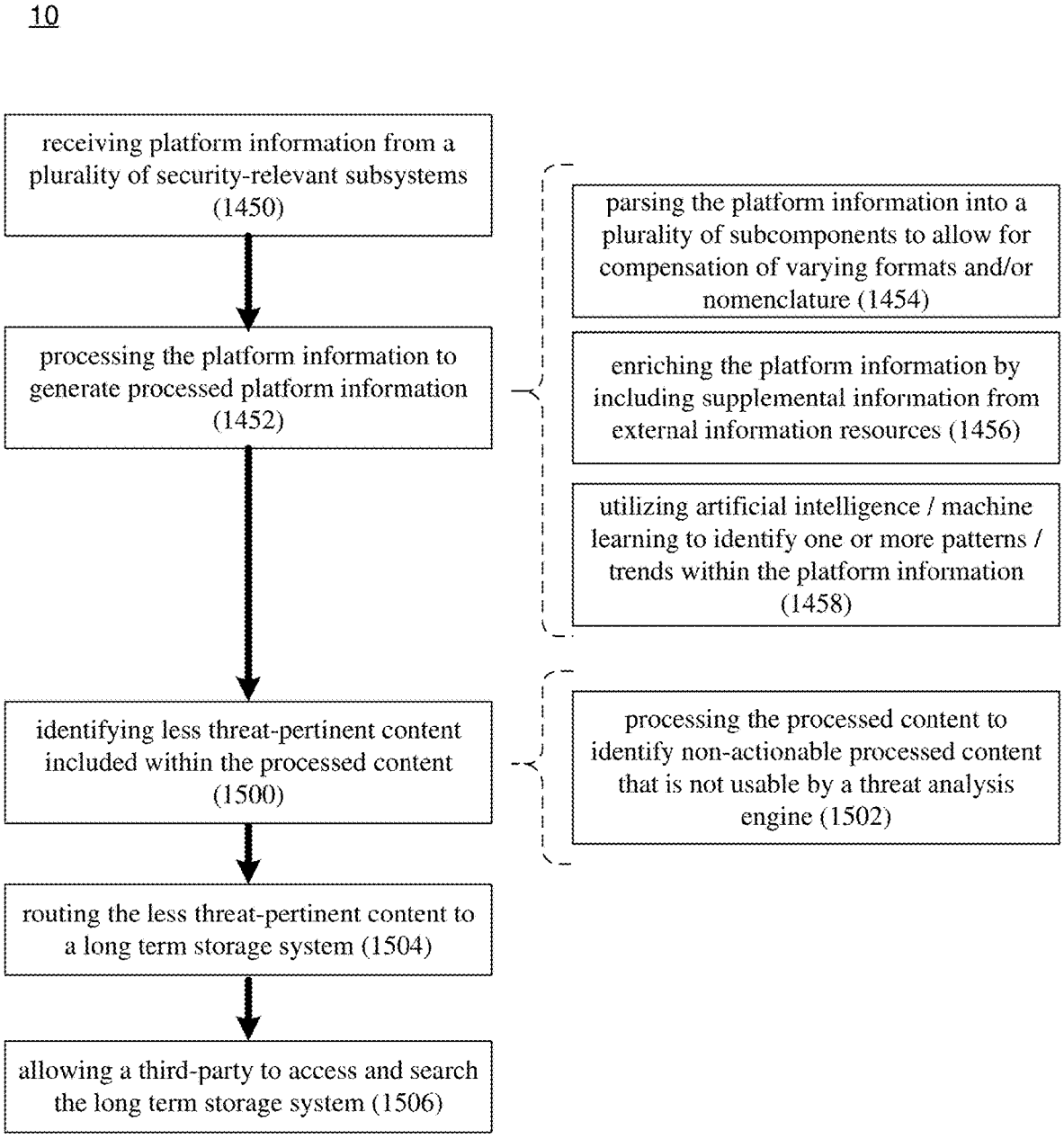

10
receiving platform information from a plurality of security-relevant subsystems (1450)
processing the platform information to generate processed platform information (1452)
parsing the platform information into a plurality of subcomponents to allow for compensation of varying formats and/or nomenclature (1454)
enriching the platform information by including supplemental information from external information resources (1456)
utilizing artificial intelligence / machine learning to identify one or more patterns / trends within the platform information (1458)
identifying less threat-pertinent content included within the processed content (1500)
processing the processed content to identify non-actionable processed content that is not usable by a threat analysis engine (1502)
routing the less threat-pertinent content to a long term storage system (1504)
allowing a third-party to access and search the long term storage system (1506)

FIG. 29

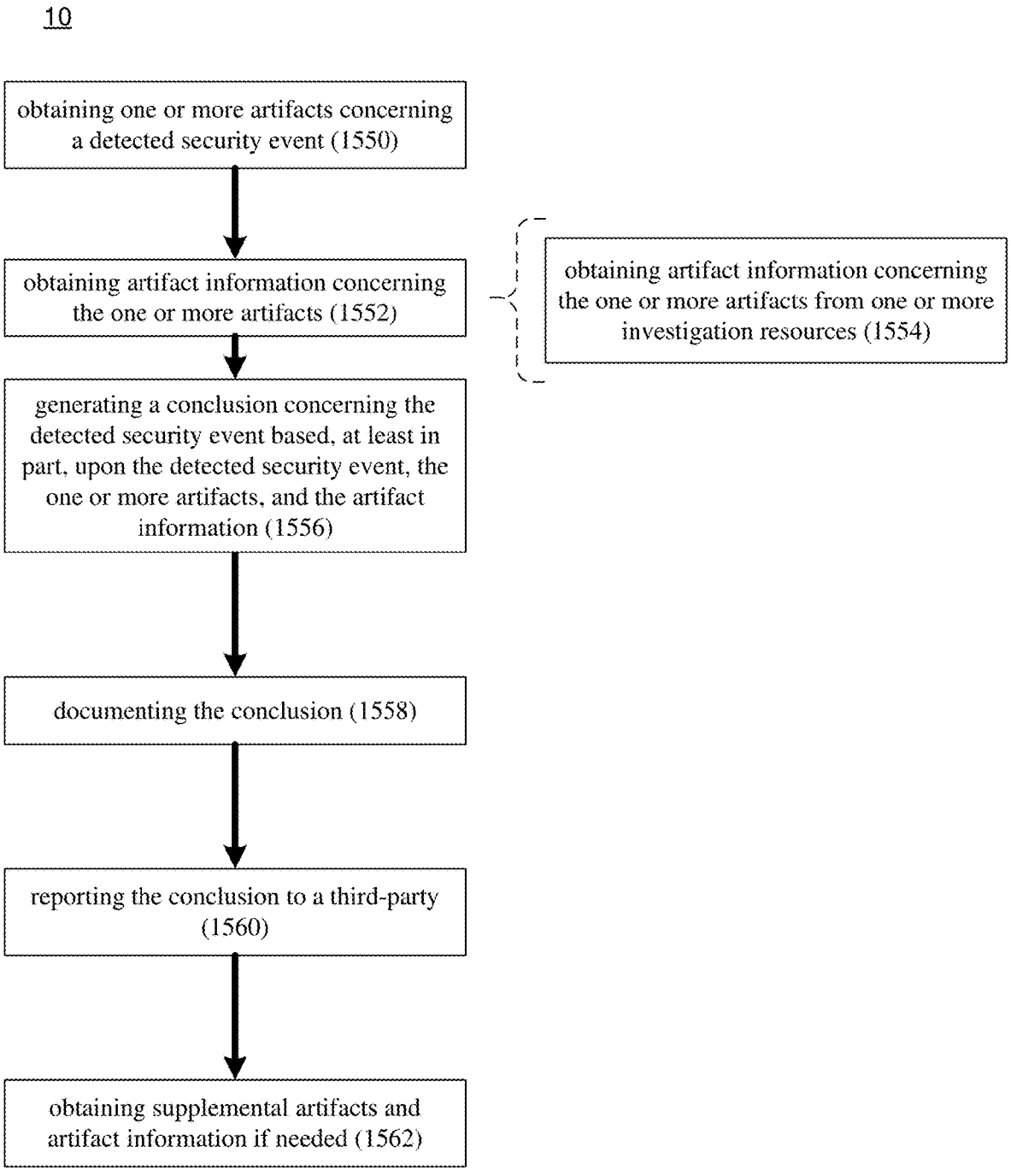

10
obtaining one or more artifacts concerning a detected security event (1550)
obtaining artifact information concerning the one or more artifacts (1552)
obtaining artifact information concerning the one or more artifacts from one or more investigation resources (1554)
generating a conclusion concerning the detected security event based, at least in part, upon the detected security event, the one or more artifacts, and the artifact information (1556)
documenting the conclusion (1558)
reporting the conclusion to a third-party (1560)
obtaining supplemental artifacts and artifact information if needed (1562)

FIG. 30

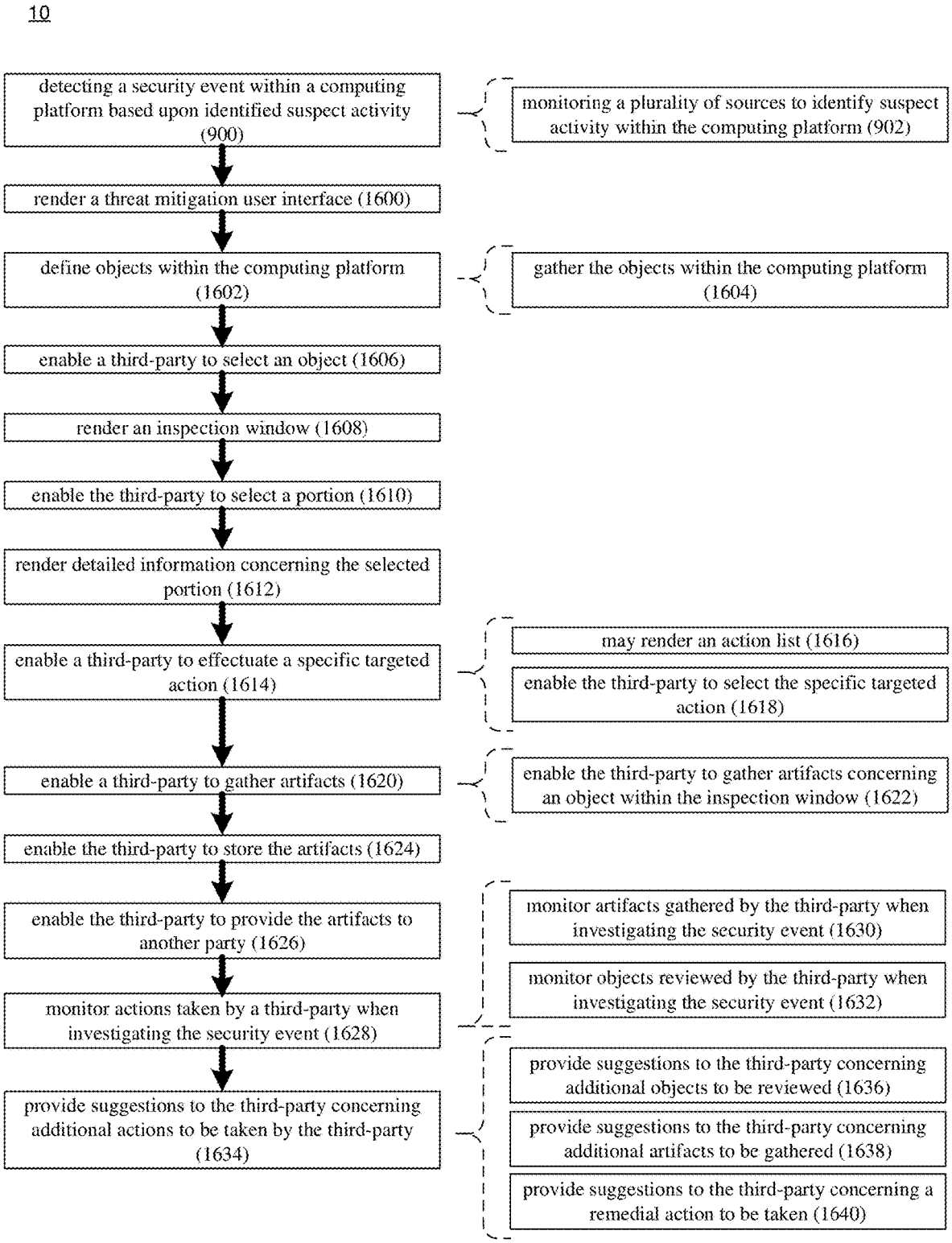

10
detecting a security event within a computing platform based upon identified suspect activity (900)
monitoring a plurality of sources to identify suspect activity within the computing platform (902)
render a threat mitigation user interface (1600)
define objects within the computing platform (1602)
gather the objects within the computing platform (1604)
enable a third-party to select an object (1606)
render an inspection window (1608)
enable the third-party to select a portion (1610)
render detailed information concerning the selected portion (1612)
enable a third-party to effectuate a specific targeted action (1614)
may render an action list (1616)
enable the third-party to select the specific targeted action (1618)
enable a third-party to gather artifacts (1620)
enable the third-party to gather artifacts concerning an object within the inspection window (1622)
enable the third-party to store the artifacts (1624)
enable the third-party to provide the artifacts to another party (1626)
monitor actions taken by a third-party when investigating the security event (1628)
monitor artifacts gathered by the third-party when investigating the security event (1630)
monitor objects reviewed by the third-party when investigating the security event (1632)
provide suggestions to the third-party concerning additional actions to be taken by the third-party (1634)
provide suggestions to the third-party concerning additional objects to be reviewed (1636)
provide suggestions to the third-party concerning additional artifacts to be gathered (1638)
provide suggestions to the third-party concerning a remedial action to be taken (1640)

FIG. 31

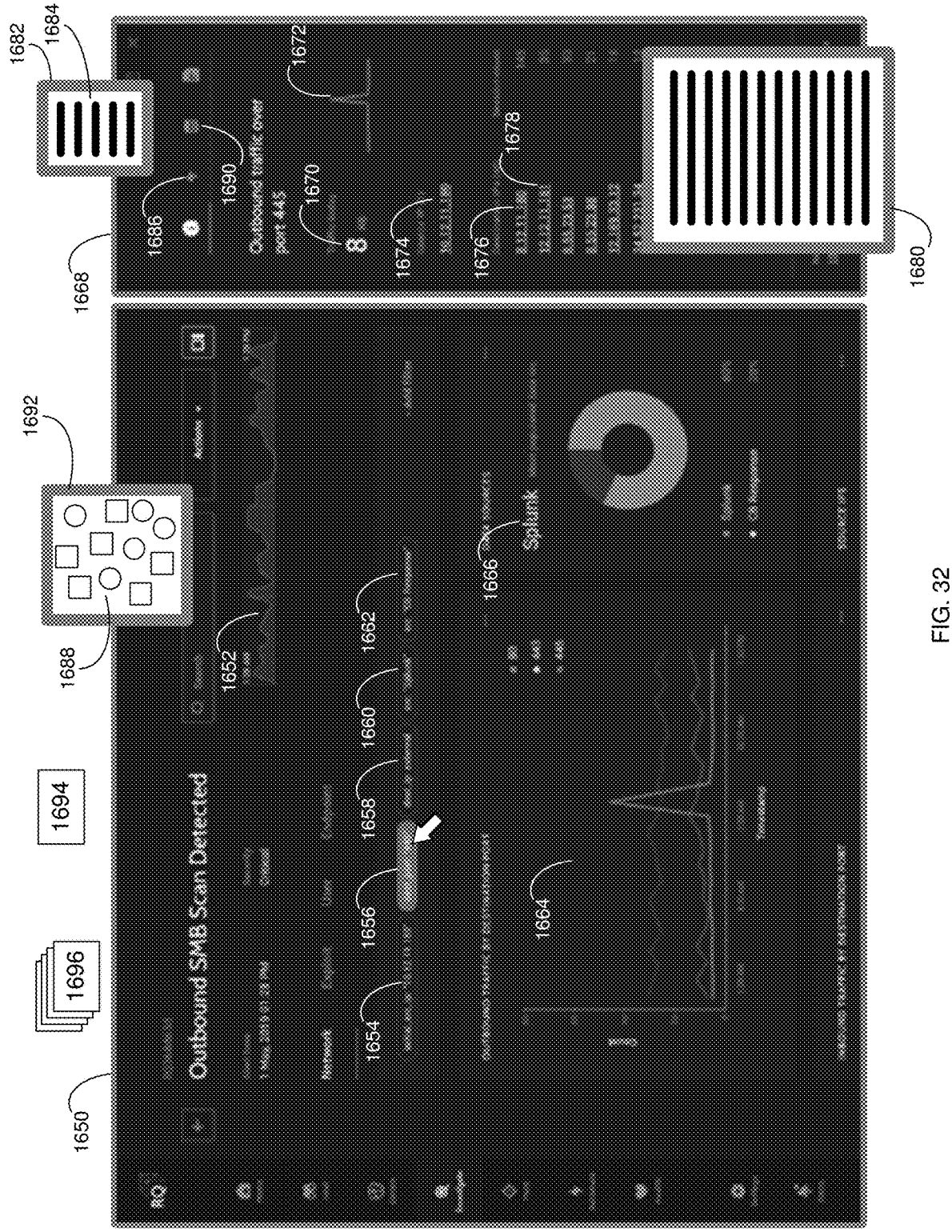

Outbound SMB Scan Detected
1650
1652
1654
1656
1658
1660
1662
1664
1666
1668
1670
1672
1674
1676
1678
1680
1682
1684
1686
1688
1690
1692
1694
1696

FIG. 32

THREAT MITIGATION SYSTEM AND METHOD

RELATED APPLICATION(S)

[0001] This application claims the benefit of the following U.S. Provisional Application Nos.: 62/879,105, filed on 26 Jul. 2019; and 62/883,797, filed on 7 Aug. 2019, their entire contents of which are herein incorporated by reference.

TECHNICAL FIELD

[0002] This disclosure relates to threat mitigation systems and, more particularly, to threat mitigation systems that utilize Artificial Intelligence (AI) and Machine Learning (ML).

BACKGROUND

[0003] In the computer world, there is a constant battle occurring between bad actors that want to attack computing platforms and good actors who try to prevent the same. Unfortunately, the complexity of such computer attacks in constantly increasing, so technology needs to be employed that understands the complexity of these attacks and is capable of addressing the same. Additionally, the use of Artificial Intelligence (AI) and Machine Learning (ML) has revolutionized the manner in which large quantities of content may be processed so that information may be extracted that is not readily discernible to a human user. Accordingly and though the use of AI/ML, the good actors may gain the upper hand in this never ending battle.

SUMMARY OF DISCLOSURE

Concept 4

[0004] In one implementation, a computer-implemented method is executed on a computing device and includes: rendering a threat mitigation user interface that identifies objects within a computing platform in response to a security event; monitoring actions taken by a third-party when investigating the security event; and providing suggestions to the third-party concerning additional actions to be taken by the third-party concerning the investigation of the security event.

[0005] One or more of the following features may be included. Monitoring actions taken by a third-party when investigating the security event may include monitoring artifacts gathered by the third-party when investigating the security event. The artifacts may include one or more of: raw data; screen shots; graphics; notes; annotations; audio recordings; and video recordings. Monitoring actions taken by a third-party when investigating the security event may include monitoring objects reviewed by the third-party when investigating the security event. Providing suggestions to the third-party concerning additional actions to be taken by the third-party concerning the investigation of the security event may include providing suggestions to the third-party concerning additional objects to be reviewed by the third-party when investigating the security event. Providing suggestions to the third-party concerning additional actions to be taken by the third-party concerning the investigation of the security event may include providing suggestions to the third-party concerning additional artifacts to be gathered by the third-party when investigating the security event. Providing suggestions to the third-party concerning additional actions to be taken by the third-party concerning the investigation of the security event may include providing suggestions to the third-party concerning a remedial action to be taken by the third-party when investigating the security event. The third-party may be enabled to select an object within the threat mitigation user interface, thus defining a selected object. An inspection window may be rendered that defines object information concerning the selected object. The inspection window may be a popup inspection window. The inspection window may be a slide out inspection window.

[0006] In another implementation, a computer program product resides on a computer readable medium and has a plurality of instructions stored on it. When executed by a processor, the instructions cause the processor to perform operations including: rendering a threat mitigation user interface that identifies objects within a computing platform in response to a security event; enabling a third-party to select an object within the threat mitigation user interface, thus defining a selected object; and rendering an inspection window that defines object information concerning the selected object.

[0007] One or more of the following features may be included. Monitoring actions taken by a third-party when investigating the security event may include monitoring artifacts gathered by the third-party when investigating the security event. The artifacts may include one or more of: raw data; screen shots; graphics; notes; annotations; audio recordings; and video recordings. Monitoring actions taken by a third-party when investigating the security event may include monitoring objects reviewed by the third-party when investigating the security event. Providing suggestions to the third-party concerning additional actions to be taken by the third-party concerning the investigation of the security event may include providing suggestions to the third-party concerning additional objects to be reviewed by the third-party when investigating the security event. Providing suggestions to the third-party concerning additional actions to be taken by the third-party concerning the investigation of the security event may include providing suggestions to the third-party concerning additional artifacts to be gathered by the third-party when investigating the security event. Providing suggestions to the third-party concerning additional actions to be taken by the third-party concerning the investigation of the security event may include providing suggestions to the third-party concerning a remedial action to be taken by the third-party when investigating the security event. The third-party may be enabled to select an object within the threat mitigation user interface, thus defining a selected object. An inspection window may be rendered that defines object information concerning the selected object. The inspection window may be a popup inspection window. The inspection window may be a slide out inspection window.

[0008] In another implementation, a computing system includes a processor and memory is configured to perform operations including: rendering a threat mitigation user interface that identifies objects within a computing platform in response to a security event; enabling a third-party to select an object within the threat mitigation user interface, thus defining a selected object; and rendering an inspection window that defines object information concerning the selected object.

[0009] One or more of the following features may be included. Monitoring actions taken by a third-party when investigating the security event may include monitoring

artifacts gathered by the third-party when investigating the security event. The artifacts may include one or more of: raw data; screen shots; graphics; notes; annotations; audio recordings; and video recordings. Monitoring actions taken by a third-party when investigating the security event may include monitoring objects reviewed by the third-party when investigating the security event. Providing suggestions to the third-party concerning additional actions to be taken by the third-party concerning the investigation of the security event may include providing suggestions to the third-party concerning additional objects to be reviewed by the third-party when investigating the security event. Providing suggestions to the third-party concerning additional actions to be taken by the third-party concerning the investigation of the security event may include providing suggestions to the third-party concerning additional artifacts to be gathered by the third-party when investigating the security event. Providing suggestions to the third-party concerning additional actions to be taken by the third-party concerning the investigation of the security event may include providing suggestions to the third-party concerning a remedial action to be taken by the third-party when investigating the security event. The third-party may be enabled to select an object within the threat mitigation user interface, thus defining a selected object. An inspection window may be rendered that defines object information concerning the selected object. The inspection window may be a popup inspection window. The inspection window may be a slide out inspection window.

[0010] The details of one or more implementations are set forth in the accompanying drawings and the description below. Other features and advantages will become apparent from the description, the drawings, and the claims.

BRIEF DESCRIPTION OF THE DRAWINGS

[0011] FIG. **1** is a diagrammatic view of a distributed computing network including a computing device that executes a threat mitigation process according to an embodiment of the present disclosure;

[0012] FIG. **2** is a diagrammatic view of an exemplary probabilistic model rendered by a probabilistic process of the threat mitigation process of FIG. **1** according to an embodiment of the present disclosure;

[0013] FIG. **3** is a diagrammatic view of the computing platform of FIG. **1** according to an embodiment of the present disclosure;

[0014] FIG. **4** is a flowchart of an implementation of the threat mitigation process of FIG. **1** according to an embodiment of the present disclosure;

[0015] FIGS. **5-6** are diagrammatic views of screens rendered by the threat mitigation process of FIG. **1** according to an embodiment of the present disclosure;

[0016] FIGS. **7-9** are flowcharts of other implementations of the threat mitigation process of FIG. **1** according to an embodiment of the present disclosure;

[0017] FIG. **10** is a diagrammatic view of a screen rendered by the threat mitigation process of FIG. **1** according to an embodiment of the present disclosure;

[0018] FIG. **11** is a flowchart of another implementation of the threat mitigation process of FIG. **1** according to an embodiment of the present disclosure;

[0019] FIG. **12** is a diagrammatic view of a screen rendered by the threat mitigation process of FIG. **1** according to an embodiment of the present disclosure;

[0020] FIG. **13** is a flowchart of another implementation of the threat mitigation process of FIG. **1** according to an embodiment of the present disclosure;

[0021] FIG. **14** is a diagrammatic view of a screen rendered by the threat mitigation process of FIG. **1** according to an embodiment of the present disclosure;

[0022] FIG. **15** is a flowchart of another implementation of the threat mitigation process of FIG. **1** according to an embodiment of the present disclosure;

[0023] FIG. **16** is a diagrammatic view of screens rendered by the threat mitigation process of FIG. **1** according to an embodiment of the present disclosure;

[0024] FIGS. **17-23** are flowcharts of other implementations of the threat mitigation process of FIG. **1** according to an embodiment of the present disclosure;

[0025] FIG. **24** is a diagrammatic view of a screen rendered by the threat mitigation process of FIG. **1** according to an embodiment of the present disclosure;

[0026] FIGS. **25-31** are flowcharts of other implementations of the threat mitigation process of FIG. **1** according to an embodiment of the present disclosure; and

[0027] FIG. **32** is a diagrammatic view of a screen rendered by the threat mitigation process of FIG. **1** according to an embodiment of the present disclosure.

[0028] Like reference symbols in the various drawings indicate like elements.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

[0029] System Overview

[0030] Referring to FIG. **1**, there is shown threat mitigation process **10**. Threat mitigation process **10** may be implemented as a server-side process, a client-side process, or a hybrid server-side/client-side process. For example, threat mitigation process **10** may be implemented as a purely server-side process via threat mitigation process **10***s*. Alternatively, threat mitigation process **10** may be implemented as a purely client-side process via one or more of threat mitigation process **10***c***1**, threat mitigation process **10***c***2**, threat mitigation process **10***c***3**, and threat mitigation process **10***c***4**. Alternatively still, threat mitigation process **10** may be implemented as a hybrid server-side/client-side process via threat mitigation process **10***s* in combination with one or more of threat mitigation process **10***c***1**, threat mitigation process **10***c***2**, threat mitigation process **10***c***3**, and threat mitigation process **10***c***4**. Accordingly, threat mitigation process **10** as used in this disclosure may include any combination of threat mitigation process **10***s*, threat mitigation process **10***c***1**, threat mitigation process **10***c***2**, threat mitigation process, and threat mitigation process **10***c***4**.

[0031] Threat mitigation process **10***s* may be a server application and may reside on and may be executed by computing device **12**, which may be connected to network **14** (e.g., the Internet or a local area network). Examples of computing device **12** may include, but are not limited to: a personal computer, a laptop computer, a personal digital assistant, a data-enabled cellular telephone, a notebook computer, a television with one or more processors embedded therein or coupled thereto, a cable/satellite receiver with one or more processors embedded therein or coupled thereto, a server computer, a series of server computers, a mini computer, a mainframe computer, or a cloud-based computing network.

[0032] The instruction sets and subroutines of threat mitigation process **10***s*, which may be stored on storage device **16** coupled to computing device **12**, may be executed by one or more processors (not shown) and one or more memory architectures (not shown) included within computing device **12**. Examples of storage device **16** may include but are not limited to: a hard disk drive; a RAID device; a random access memory (RAM); a read-only memory (ROM); and all forms of flash memory storage devices.

[0033] Network **14** may be connected to one or more secondary networks (e.g., network **18**), examples of which may include but are not limited to: a local area network; a wide area network; or an intranet, for example.

[0034] Examples of threat mitigation processes **10***c***1**, **10***c***2**, **10***c***3**, **10***c***4** may include but are not limited to a client application, a web browser, a game console user interface, or a specialized application (e.g., an application running on e.g., the Android™ platform or the iOS platform). The instruction sets and subroutines of threat mitigation processes **10***c***1**, **10***c***2**, **10***c***3**, **10***c***4**, which may be stored on storage devices **20**, **22**, **24**, **26** (respectively) coupled to client electronic devices **28**, **30**, **32**, **34** (respectively), may be executed by one or more processors (not shown) and one or more memory architectures (not shown) incorporated into client electronic devices **28**, **30**, **32**, **34** (respectively). Examples of storage device **16** may include but are not limited to: a hard disk drive; a RAID device; a random access memory (RAM); a read-only memory (ROM); and all forms of flash memory storage devices.

[0035] Examples of client electronic devices **28**, **30**, **32**, **34** may include, but are not limited to, data-enabled, cellular telephone **28**, laptop computer **30**, personal digital assistant **32**, personal computer **34**, a notebook computer (not shown), a server computer (not shown), a gaming console (not shown), a smart television (not shown), and a dedicated network device (not shown). Client electronic devices **28**, **30**, **32**, **34** may each execute an operating system, examples of which may include but are not limited to Microsoft Windows™, Android™, WebOS™, iOS™, Redhat Linux™, or a custom operating system.

[0036] Users **36**, **38**, **40**, **42** may access threat mitigation process **10** directly through network **14** or through secondary network **18**. Further, threat mitigation process **10** may be connected to network **14** through secondary network **18**, as illustrated with link line **44**.

[0037] The various client electronic devices (e.g., client electronic devices **28**, **30**, **32**, **34**) may be directly or indirectly coupled to network **14** (or network **18**). For example, data-enabled, cellular telephone **28** and laptop computer **30** are shown wirelessly coupled to network **14** via wireless communication channels **46**, **48** (respectively) established between data-enabled, cellular telephone **28**, laptop computer **30** (respectively) and cellular network/bridge **50**, which is shown directly coupled to network **14**. Further, personal digital assistant **32** is shown wirelessly coupled to network **14** via wireless communication channel **52** established between personal digital assistant **32** and wireless access point (i.e., WAP) **54**, which is shown directly coupled to network **14**. Additionally, personal computer **34** is shown directly coupled to network **18** via a hardwired network connection.

[0038] WAP **54** may be, for example, an IEEE 802.11a, 802.11b, 802.11g, 802.11n, Wi-Fi, and/or Bluetooth device that is capable of establishing wireless communication channel **52** between personal digital assistant **32** and WAP **54**. As is known in the art, IEEE 802.11x specifications may use Ethernet protocol and carrier sense multiple access with collision avoidance (i.e., CSMA/CA) for path sharing. The various 802.11x specifications may use phase-shift keying (i.e., PSK) modulation or complementary code keying (i.e., CCK) modulation, for example. As is known in the art, Bluetooth is a telecommunications industry specification that allows e.g., mobile phones, computers, and personal digital assistants to be interconnected using a short-range wireless connection.

Artificial Intelligence/Machines Learning Overview:

[0039] Assume for illustrative purposes that threat mitigation process **10** includes probabilistic process **56** (e.g., an artificial intelligence/machine learning process) that is configured to process information (e.g., information **58**). As will be discussed below in greater detail, examples of information **58** may include but are not limited to platform information (e.g., structured or unstructured content) being scanned to detect security events (e.g., access auditing; anomalies; authentication; denial of services; exploitation; malware; phishing; spamming; reconnaissance; and/or web attack) within a monitored computing platform (e.g., computing platform **60**).

[0040] As is known in the art, structured content may be content that is separated into independent portions (e.g., fields, columns, features) and, therefore, may have a pre-defined data model and/or is organized in a pre-defined manner. For example, if the structured content concerns an employee list: a first field, column or feature may define the first name of the employee; a second field, column or feature may define the last name of the employee; a third field, column or feature may define the home address of the employee; and a fourth field, column or feature may define the hire date of the employee.

[0041] Further and as is known in the art, unstructured content may be content that is not separated into independent portions (e.g., fields, columns, features) and, therefore, may not have a pre-defined data model and/or is not organized in a pre-defined manner. For example, if the unstructured content concerns the same employee list: the first name of the employee, the last name of the employee, the home address of the employee, and the hire date of the employee may all be combined into one field, column or feature.

[0042] For the following illustrative example, assume that information **58** is unstructured content, an example of which may include but is not limited to unstructured user feedback received by a company (e.g., text-based feedback such as text-messages, social media posts, and email messages; and transcribed voice-based feedback such as transcribed voice mail, and transcribed voice messages).

[0043] When processing information **58**, probabilistic process **56** may use probabilistic modeling to accomplish such processing, wherein examples of such probabilistic modeling may include but are not limited to discriminative modeling, generative modeling, or combinations thereof.

[0044] As is known in the art, probabilistic modeling may be used within modern artificial intelligence systems (e.g., probabilistic process **56**), in that these probabilistic models may provide artificial intelligence systems with the tools required to autonomously analyze vast quantities of data (e.g., information **58**).

[0045] Examples of the tasks for which probabilistic modeling may be utilized may include but are not limited to:

[0046] predicting media (music, movies, books) that a user may like or enjoy based upon media that the user has liked or enjoyed in the past;

[0047] transcribing words spoken by a user into editable text;

[0048] grouping genes into gene clusters;

[0049] identifying recurring patterns within vast data sets;

[0050] filtering email that is believed to be spam from a user's inbox;

[0051] generating clean (i.e., non-noisy) data from a noisy data set;

[0052] analyzing (voice-based or text-based) customer feedback; and

[0053] diagnosing various medical conditions and diseases.

[0054] For each of the above-described applications of probabilistic modeling, an initial probabilistic model may be defined, wherein this initial probabilistic model may be subsequently (e.g., iteratively or continuously) modified and revised, thus allowing the probabilistic models and the artificial intelligence systems (e.g., probabilistic process **56**) to "learn" so that future probabilistic models may be more precise and may explain more complex data sets.

[0055] Accordingly, probabilistic process **56** may define an initial probabilistic model for accomplishing a defined task (e.g., the analyzing of information **58**). For the illustrative example, assume that this defined task is analyzing customer feedback (e.g., information **58**) that is received from customers of e.g., store **62** via an automated feedback phone line. For this example, assume that information **58** is initially voice-based content that is processed via e.g., a speech-to-text process that results in unstructured text-based customer feedback (e.g., information **58**).

[0056] With respect to probabilistic process **56**, a probabilistic model may be utilized to go from initial observations about information **58** (e.g., as represented by the initial branches of a probabilistic model) to conclusions about information **58** (e.g., as represented by the leaves of a probabilistic model).

[0057] As used in this disclosure, the term "branch" may refer to the existence (or non-existence) of a component (e.g., a sub-model) of (or included within) a model. Examples of such a branch may include but are not limited to: an execution branch of a probabilistic program or other generative model, a part (or parts) of a probabilistic graphical model, and/or a component neural network that may (or may not) have been previously trained.

[0058] While the following discussion provides a detailed example of a probabilistic model, this is for illustrative purposes only and is not intended to be a limitation of this disclosure, as other configurations are possible and are considered to be within the scope of this disclosure. For example, the following discussion may concern any type of model (e.g., be it probabilistic or other) and, therefore, the below-described probabilistic model is merely intended to be one illustrative example of a type of model and is not intended to limit this disclosure to probabilistic models.

[0059] Additionally, while the following discussion concerns word-based routing of messages through a probabilistic model, this is for illustrative purposes only and is not intended to be a limitation of this disclosure, as other configurations are possible and are considered to be within the scope of this disclosure. Examples of other types of information that may be used to route messages through a probabilistic model may include: the order of the words within a message; and the punctuation interspersed throughout the message.

[0060] For example and referring also to FIG. **2**, there is shown one simplified example of a probabilistic model (e.g., probabilistic model **100**) that may be utilized to analyze information **58** (e.g. unstructured text-based customer feedback) concerning store **62**. The manner in which probabilistic model **100** may be automatically-generated by probabilistic process **56** will be discussed below in detail. In this particular example, probabilistic model **100** may receive information **58** (e.g. unstructured text-based customer feedback) at branching node **102** for processing. Assume that probabilistic model **100** includes four branches off of branching node **102**, namely: service branch **104**; selection branch **106**; location branch **108**; and value branch **110** that respectively lead to service node **112**, selection node **114**, location node **116**, and value node **118**.

[0061] As stated above, service branch **104** may lead to service node **112**, which may be configured to process the portion of information **58** (e.g. unstructured text-based customer feedback) that concerns (in whole or in part) feedback concerning the customer service of store **62**. For example, service node **112** may define service word list **120** that may include e.g., the word service, as well as synonyms of (and words related to) the word service (e.g., cashier, employee, greeter and manager). Accordingly and in the event that a portion of information **58** (e.g., a text-based customer feedback message) includes the word cashier, employee, greeter and/or manager, that portion of information **58** may be considered to be text-based customer feedback concerning the service received at store **62** and (therefore) may be routed to service node **112** of probabilistic model **100** for further processing. Assume for this illustrative example that probabilistic model **100** includes two branches off of service node **112**, namely: good service branch **122** and bad service branch **124**.

[0062] Good service branch **122** may lead to good service node **126**, which may be configured to process the portion of information **58** (e.g. unstructured text-based customer feedback) that concerns (in whole or in part) good feedback concerning the customer service of store **62**. For example, good service node **126** may define good service word list **128** that may include e.g., the word good, as well as synonyms of (and words related to) the word good (e.g., courteous, friendly, lovely, happy, and smiling). Accordingly and in the event that a portion of information **58** (e.g., a text-based customer feedback message) that was routed to service node **112** includes the word good, courteous, friendly, lovely, happy, and/or smiling, that portion of information **58** may be considered to be text-based customer feedback indicative of good service received at store **62** (and, therefore, may be routed to good service node **126**).

[0063] Bad service branch **124** may lead to bad service node **130**, which may be configured to process the portion of information **58** (e.g. unstructured text-based customer feedback) that concerns (in whole or in part) bad feedback concerning the customer service of store **62**. For example, bad service node **130** may define bad service word list **132** that may include e.g., the word bad, as well as synonyms of (and words related to) the word bad (e.g., rude, mean, jerk,

miserable, and scowling). Accordingly and in the event that a portion of information **58** (e.g., a text-based customer feedback message) that was routed to service node **112** includes the word bad, rude, mean, jerk, miserable, and/or scowling, that portion of information **58** may be considered to be text-based customer feedback indicative of bad service received at store **62** (and, therefore, may be routed to bad service node **130**).

[0064] As stated above, selection branch **106** may lead to selection node **114**, which may be configured to process the portion of information **58** (e.g. unstructured text-based customer feedback) that concerns (in whole or in part) feedback concerning the selection available at store **62**. For example, selection node **114** may define selection word list **134** that may include e.g., words indicative of the selection available at store **62**. Accordingly and in the event that a portion of information **58** (e.g., a text-based customer feedback message) includes any of the words defined within selection word list **134**, that portion of information **58** may be considered to be text-based customer feedback concerning the selection available at store **62** and (therefore) may be routed to selection node **114** of probabilistic model **100** for further processing. Assume for this illustrative example that probabilistic model **100** includes two branches off of selection node **114**, namely: good selection branch **136** and bad selection branch **138**.

[0065] Good selection branch **136** may lead to good selection node **140**, which may be configured to process the portion of information **58** (e.g. unstructured text-based customer feedback) that concerns (in whole or in part) good feedback concerning the selection available at store **62**. For example, good selection node **140** may define good selection word list **142** that may include words indicative of a good selection at store **62**. Accordingly and in the event that a portion of information **58** (e.g., a text-based customer feedback message) that was routed to selection node **114** includes any of the words defined within good selection word list **142**, that portion of information **58** may be considered to be text-based customer feedback indicative of a good selection available at store **62** (and, therefore, may be routed to good selection node **140**).

[0066] Bad selection branch **138** may lead to bad selection node **144**, which may be configured to process the portion of information **58** (e.g. unstructured text-based customer feedback) that concerns (in whole or in part) bad feedback concerning the selection available at store **62**. For example, bad selection node **144** may define bad selection word list **146** that may include words indicative of a bad selection at store **62**. Accordingly and in the event that a portion of information **58** (e.g., a text-based customer feedback message) that was routed to selection node **114** includes any of the words defined within bad selection word list **146**, that portion of information **58** may be considered to be text-based customer feedback indicative of a bad selection being available at store **62** (and, therefore, may be routed to bad selection node **144**).

[0067] As stated above, location branch **108** may lead to location node **116**, which may be configured to process the portion of information **58** (e.g. unstructured text-based customer feedback) that concerns (in whole or in part) feedback concerning the location of store **62**. For example, location node **116** may define location word list **148** that may include e.g., words indicative of the location of store **62**. Accordingly and in the event that a portion of information **58** (e.g., a text-based customer feedback message) includes any of the words defined within location word list **148**, that portion of information **58** may be considered to be text-based customer feedback concerning the location of store **62** and (therefore) may be routed to location node **116** of probabilistic model **100** for further processing. Assume for this illustrative example that probabilistic model **100** includes two branches off of location node **116**, namely: good location branch **150** and bad location branch **152**.

[0068] Good location branch **150** may lead to good location node **154**, which may be configured to process the portion of information **58** (e.g. unstructured text-based customer feedback) that concerns (in whole or in part) good feedback concerning the location of store **62**. For example, good location node **154** may define good location word list **156** that may include words indicative of store **62** being in a good location. Accordingly and in the event that a portion of information **58** (e.g., a text-based customer feedback message) that was routed to location node **116** includes any of the words defined within good location word list **156**, that portion of information **58** may be considered to be text-based customer feedback indicative of store **62** being in a good location (and, therefore, may be routed to good location node **154**).

[0069] Bad location branch **152** may lead to bad location node **158**, which may be configured to process the portion of information **58** (e.g. unstructured text-based customer feedback) that concerns (in whole or in part) bad feedback concerning the location of store **62**. For example, bad location node **158** may define bad location word list **160** that may include words indicative of store **62** being in a bad location. Accordingly and in the event that a portion of information **58** (e.g., a text-based customer feedback message) that was routed to location node **116** includes any of the words defined within bad location word list **160**, that portion of information **58** may be considered to be text-based customer feedback indicative of store **62** being in a bad location (and, therefore, may be routed to bad location node **158**).

[0070] As stated above, value branch **110** may lead to value node **118**, which may be configured to process the portion of information **58** (e.g. unstructured text-based customer feedback) that concerns (in whole or in part) feedback concerning the value received at store **62**. For example, value node **118** may define value word list **162** that may include e.g., words indicative of the value received at store **62**. Accordingly and in the event that a portion of information **58** (e.g., a text-based customer feedback message) includes any of the words defined within value word list **162**, that portion of information **58** may be considered to be text-based customer feedback concerning the value received at store **62** and (therefore) may be routed to value node **118** of probabilistic model **100** for further processing. Assume for this illustrative example that probabilistic model **100** includes two branches off of value node **118**, namely: good value branch **164** and bad value branch **166**.

[0071] Good value branch **164** may lead to good value node **168**, which may be configured to process the portion of information **58** (e.g. unstructured text-based customer feedback) that concerns (in whole or in part) good value being received at store **62**. For example, good value node **168** may define good value word list **170** that may include words indicative of receiving good value at store **62**. Accordingly and in the event that a portion of information **58** (e.g., a

text-based customer feedback message) that was routed to value node **118** includes any of the words defined within good value word list **170**, that portion of information **58** may be considered to be text-based customer feedback indicative of good value being received at store **62** (and, therefore, may be routed to good value node **168**).

[0072] Bad value branch **166** may lead to bad value node **172**, which may be configured to process the portion of information **58** (e.g. unstructured text-based customer feedback) that concerns (in whole or in part) bad value being received at store **62**. For example, bad value node **172** may define bad value word list **174** that may include words indicative of receiving bad value at store **62**. Accordingly and in the event that a portion of information **58** (e.g., a text-based customer feedback message) that was routed to value node **118** includes any of the words defined within bad value word list **174**, that portion of information **58** may be considered to be text-based customer feedback indicative of bad value being received at store **62** (and, therefore, may be routed to bad value node **172**).

[0073] Once it is established that good or bad customer feedback was received concerning store **62** (i.e., with respect to the service, the selection, the location or the value), representatives and/or agents of store **62** may address the provider of such good or bad feedback via e.g., social media postings, text-messages and/or personal contact.

[0074] Assume for illustrative purposes that user **36** uses data-enabled, cellular telephone **28** to provide feedback **64** (e.g., a portion of information **58**) to an automated feedback phone line concerning store **62**. Upon receiving feedback **64** for analysis, probabilistic process **56** may identify any pertinent content that is included within feedback **64**.

[0075] For illustrative purposes, assume that user **36** was not happy with their experience at store **62** and that feedback **64** provided by user **36** was "my cashier was rude and the weather was rainy". Accordingly and for this example, probabilistic process **56** may identify the pertinent content (included within feedback **64**) as the phrase "my cashier was rude" and may ignore/remove the irrelevant content "the weather was rainy". As (in this example) feedback **64** includes the word "cashier", probabilistic process **56** may route feedback **64** to service node **112** via service branch **104**. Further, as feedback **64** also includes the word "rude", probabilistic process **56** may route feedback **64** to bad service node **130** via bad service branch **124** and may consider feedback **64** to be text-based customer feedback indicative of bad service being received at store **62**.

[0076] For further illustrative purposes, assume that user **36** was happy with their experience at store **62** and that feedback **64** provided by user **36** was "the clothing I purchased was classy but my cab got stuck in traffic". Accordingly and for this example, probabilistic process **56** may identify the pertinent content (included within feedback **64**) as the phrase "the clothing I purchased was classy" and may ignore/remove the irrelevant content "my cab got stuck in traffic". As (in this example) feedback **64** includes the word "clothing", probabilistic process **56** may route feedback **64** to selection node **114** via selection branch **106**. Further, as feedback **64** also includes the word "classy", probabilistic process **56** may route feedback **64** to good selection node **140** via good selection branch **136** and may consider feedback **64** to be text-based customer feedback indicative of a good selection being available at store **62**.

Model Generation Overview:

[0077] While the following discussion concerns the automated generation of a probabilistic model, this is for illustrative purposes only and is not intended to be a limitation of this disclosure, as other configurations are possible and are considered to be within the scope of this disclosure. For example, the following discussion of automated generation may be utilized on any type of model. For example, the following discussion may be applicable to any other form of probabilistic model or any form of generic model (such as Dempster Shaffer theory or fuzzy logic).

[0078] As discussed above, probabilistic model **100** may be utilized to categorize information **58**, thus allowing the various messages included within information **58** to be routed to (in this simplified example) one of eight nodes (e.g., good service node **126**, bad service node **130**, good selection node **140**, bad selection node **144**, good location node **154**, bad location node **158**, good value node **168**, and bad value node **172**). For the following example, assume that store **62** is a long-standing and well established shopping establishment. Further, assume that information **58** is a very large quantity of voice mail messages (>10,000 messages) that were left by customers of store **62** on a voice-based customer feedback line. Additionally, assume that this very large quantity of voice mail messages (>10,000) have been transcribed into a very large quantity of text-based messages (>10,000).

[0079] Probabilistic process **56** may be configured to automatically define probabilistic model **100** based upon information **58**. Accordingly, probabilistic process **56** may receive content (e.g., a very large quantity of text-based messages) and may be configured to define one or more probabilistic model variables for probabilistic model **100**. For example, probabilistic process **56** may be configured to allow a user to specify such probabilistic model variables. Another example of such variables may include but is not limited to values and/or ranges of values for a data flow variable. For the following discussion and for this disclosure, examples of a "variable" may include but are not limited to variables, parameters, ranges, branches and nodes.

[0080] Specifically and for this example, assume that probabilistic process **56** defines the initial number of branches (i.e., the number of branches off of branching node **102**) within probabilistic model **100** as four (i.e., service branch **104**, selection branch **106**, location branch **108** and value branch **110**). The defining of the initial number of branches (i.e., the number of branches off of branching node **102**) within probabilistic model **100** as four may be effectuated in various ways (e.g., manually or algorithmically). Further and when defining probabilistic model **100** based, at least in part, upon information **58** and the one or more model variables (i.e., defining the number of branches off of branching node **102** as four), probabilistic process **56** may process information **58** to identify the pertinent content included within information **58**. As discussed above, probabilistic process **56** may identify the pertinent content (included within information **58**) and may ignore/remove the irrelevant content.

[0081] This type of processing of information **58** may continue for all of the very large quantity of text-based messages (>10,000) included within information **58**. And using the probabilistic modeling technique described above, probabilistic process **56** may define a first version of the probabilistic model (e.g., probabilistic model **100**) based, at

least in part, upon pertinent content found within information **58**. Accordingly, a first text-based message included within information **58** may be processed to extract pertinent information from that first message, wherein this pertinent information may be grouped in a manner to correspond (at least temporarily) with the requirement that four branches originate from branching node **102** (as defined above).

[0082] As probabilistic process **56** continues to process information **58** to identify pertinent content included within information **58**, probabilistic process **56** may identify patterns within these text-based message included within information **58**. For example, the messages may all concern one or more of the service, the selection, the location and/or the value of store **62**. Further and e.g., using the probabilistic modeling technique described above, probabilistic process **56** may process information **58** to e.g.: a) sort text-based messages concerning the service into positive or negative service messages; b) sort text-based messages concerning the selection into positive or negative selection messages; c) sort text-based messages concerning the location into positive or negative location messages; and/or d) sort text-based messages concerning the value into positive or negative service messages. For example, probabilistic process **56** may define various lists (e.g., lists **128**, **132**, **142**, **146**, **156**, **160**, **170**, **174**) by starting with a root word (e.g., good or bad) and may then determine synonyms for these words and use those words and synonyms to populate lists **128**, **132**, **142**, **146**, **156**, **160**, **170**, **174**.

[0083] Continuing with the above-stated example, once information **58** (or a portion thereof) is processed by probabilistic process **56**, probabilistic process **56** may define a first version of the probabilistic model (e.g., probabilistic model **100**) based, at least in part, upon pertinent content found within information **58**. Probabilistic process **56** may compare the first version of the probabilistic model (e.g., probabilistic model **100**) to information **58** to determine if the first version of the probabilistic model (e.g., probabilistic model **100**) is a good explanation of the content.

[0084] When determining if the first version of the probabilistic model (e.g., probabilistic model **100**) is a good explanation of the content, probabilistic process **56** may use an ML algorithm to fit the first version of the probabilistic model (e.g., probabilistic model **100**) to the content, wherein examples of such an ML algorithm may include but are not limited to one or more of: an inferencing algorithm, a learning algorithm, an optimization algorithm, and a statistical algorithm.

[0085] For example and as is known in the art, probabilistic model **100** may be used to generate messages (in addition to analyzing them). For example and when defining a first version of the probabilistic model (e.g., probabilistic model **100**) based, at least in part, upon pertinent content found within information **58**, probabilistic process **56** may define a weight for each branch within probabilistic model **100** based upon information **58**. For example, threat mitigation process **10** may equally weight each of branches **104**, **106**, **108**, **110** at 25%. Alternatively, if e.g., a larger percentage of information **58** concerned the service received at store **62**, threat mitigation process **10** may equally weight each of branches **106**, **108**, **110** at 20%, while more heavily weighting branch **104** at 40%.

[0086] Accordingly and when probabilistic process **56** compares the first version of the probabilistic model (e.g., probabilistic model **100**) to information **58** to determine if the first version of the probabilistic model (e.g., probabilistic model **100**) is a good explanation of the content, probabilistic process **56** may generate a very large quantity of messages e.g., by auto-generating messages using the above-described probabilities, the above-described nodes & node types, and the words defined in the above-described lists (e.g., lists **128**, **132**, **142**, **146**, **156**, **160**, **170**, **174**), thus resulting in generated information **58'**. Generated information **58'** may then be compared to information **58** to determine if the first version of the probabilistic model (e.g., probabilistic model **100**) is a good explanation of the content. For example, if generated information **58'** exceeds a threshold level of similarity to information **58**, the first version of the probabilistic model (e.g., probabilistic model **100**) may be deemed a good explanation of the content. Conversely, if generated information **58'** does not exceed a threshold level of similarity to information **58**, the first version of the probabilistic model (e.g., probabilistic model **100**) may be deemed not a good explanation of the content.

[0087] If the first version of the probabilistic model (e.g., probabilistic model **100**) is not a good explanation of the content, probabilistic process **56** may define a revised version of the probabilistic model (e.g., revised probabilistic model **100'**). When defining revised probabilistic model **100'**, probabilistic process **56** may e.g., adjust weighting, adjust probabilities, adjust node counts, adjust node types, and/or adjust branch counts to define the revised version of the probabilistic model (e.g., revised probabilistic model **100'**). Once defined, the above-described process of auto-generating messages (this time using revised probabilistic model **100'**) may be repeated and this newly-generated content (e.g., generated information **58"**) may be compared to information **58** to determine if e.g., revised probabilistic model **100'** is a good explanation of the content. If revised probabilistic model **100'** is not a good explanation of the content, the above-described process may be repeated until a proper probabilistic model is defined.

[0088] The Threat Mitigation Process

[0089] As discussed above, threat mitigation process **10** may include probabilistic process **56** (e.g., an artificial intelligence/machine learning process) that may be configured to process information (e.g., information **58**), wherein examples of information **58** may include but are not limited to platform information (e.g., structured or unstructured content) that may be scanned to detect security events (e.g., access auditing; anomalies; authentication; denial of services; exploitation; malware; phishing; spamming; reconnaissance; and/or web attack) within a monitored computing platform (e.g., computing platform **60**).

[0090] Referring also to FIG. **3**, the monitored computing platform (e.g., computing platform **60**) utilized by business today may be a highly complex, multi-location computing system/network that may span multiple buildings/locations/countries. For this illustrative example, the monitored computing platform (e.g., computing platform **60**) is shown to include many discrete computing devices, examples of which may include but are not limited to: server computers (e.g., server computers **200**, **202**), desktop computers (e.g., desktop computer **204**), and laptop computers (e.g., laptop computer **206**), all of which may be coupled together via a network (e.g., network **208**), such as an Ethernet network. Computing platform **60** may be coupled to an external network (e.g., Internet **210**) through WAF (i.e., Web Application Firewall) **212**. A wireless access point (e.g., WAP

214) may be configured to allow wireless devices (e.g., smartphone **216**) to access computing platform **60**. Computing platform **60** may include various connectivity devices that enable the coupling of devices within computing platform **60**, examples of which may include but are not limited to: switch **216**, router **218** and gateway **220**. Computing platform **60** may also include various storage devices (e.g., NAS **222**), as well as functionality (e.g., API Gateway **224**) that allows software applications to gain access to one or more resources within computing platform **60**.

[0091] In addition to the devices and functionality discussed above, other technology (e.g., security-relevant subsystems **226**) may be deployed within computing platform **60** to monitor the operation of (and the activity within) computing platform **60**. Examples of security-relevant subsystems **226** may include but are not limited to: CDN (i.e., Content Delivery Network) systems; DAM (i.e., Database Activity Monitoring) systems; UBA (i.e., User Behavior Analytics) systems; MDM (i.e., Mobile Device Management) systems; IAM (i.e., Identity and Access Management) systems; DNS (i.e., Domain Name Server) systems, antivirus systems, operating systems, data lakes; data logs; security-relevant software applications; security-relevant hardware systems; and resources external to the computing platform.

[0092] Each of security-relevant subsystems **226** may monitor and log their activity with respect to computing platform **60**, resulting in the generation of platform information **228**. For example, platform information **228** associated with a client-defined MDM (i.e., Mobile Device Management) system may monitor and log the mobile devices that were allowed access to computing platform **60**.

[0093] Further, SEIM (i.e., Security Information and Event Management) system **230** may be deployed within computing platform **60**. As is known in the art, SIEM system **230** is an approach to security management that combines SIM (security information management) functionality and SEM (security event management) functionality into one security management system. The underlying principles of a SIEM system is to aggregate relevant data from multiple sources, identify deviations from the norm and take appropriate action. For example, when a security event is detected, STEM system **230** might log additional information, generate an alert and instruct other security controls to mitigate the security event. Accordingly, STEM system **230** may be configured to monitor and log the activity of security-relevant subsystems **226** (e.g., CDN (i.e., Content Delivery Network) systems; DAM (i.e., Database Activity Monitoring) systems; UBA (i.e., User Behavior Analytics) systems; MDM (i.e., Mobile Device Management) systems; IAM (i.e., Identity and Access Management) systems; DNS (i.e., Domain Name Server) systems, antivirus systems, operating systems, data lakes; data logs; security-relevant software applications; security-relevant hardware systems; and resources external to the computing platform).

[0094] Computing Platform Analysis & Reporting

[0095] As will be discussed below in greater detail, threat mitigation process **10** may be configured to e.g., analyze computing platform **60** and provide reports to third-parties concerning the same.

[0096] Referring also to FIGS. **4-6**, threat mitigation process **10** may be configured to obtain and combine information from multiple security-relevant subsystem to generate a security profile for computing platform **60**. For example, threat mitigation process **10** may obtain **300** first system-defined platform information (e.g., system-defined platform information **232**) concerning a first security-relevant subsystem (e.g., the number of operating systems deployed) within computing platform **60** and may obtain **302** at least a second system-defined platform information (e.g., system-defined platform information **234**) concerning at least a second security-relevant subsystem (e.g., the number of antivirus systems deployed) within computing platform **60**.

[0097] The first system-defined platform information (e.g., system-defined platform information **232**) and the at least a second system-defined platform information (e.g., system-defined platform information **234**) may be obtained from one or more log files defined for computing platform **60**.

[0098] Specifically, system-defined platform information **232** and/or system-defined platform information **234** may be obtained from SIEM system **230**, wherein (and as discussed above) STEM system **230** may be configured to monitor and log the activity of security-relevant subsystems **226** (e.g., CDN (i.e., Content Delivery Network) systems; DAM (i.e., Database Activity Monitoring) systems; UBA (i.e., User Behavior Analytics) systems; MDM (i.e., Mobile Device Management) systems; IAM (i.e., Identity and Access Management) systems; DNS (i.e., Domain Name Server) systems, antivirus systems, operating systems, data lakes; data logs; security-relevant software applications; security-relevant hardware systems; and resources external to the computing platform).

[0099] Alternatively, the first system-defined platform information (e.g., system-defined platform information **232**) and the at least a second system-defined platform information (e.g., system-defined platform information **234**) may be obtained from the first security-relevant subsystem (e.g., the operating systems themselves) and the at least a second security-relevant subsystem (e.g., the antivirus systems themselves). Specifically, system-defined platform information **232** and/or system-defined platform information **234** may be obtained directly from the security-relevant subsystems (e.g., the operating systems and/or the antivirus systems), which (as discussed above) may be configured to self-document their activity.

[0100] Threat mitigation process **10** may combine **308** the first system-defined platform information (e.g., system-defined platform information **232**) and the at least a second system-defined platform information (e.g., system-defined platform information **234**) to form system-defined consolidated platform information **236**. Accordingly and in this example, system-defined consolidated platform information **236** may independently define the security-relevant subsystems (e.g., security-relevant subsystems **226**) present on computing platform **60**.

[0101] Threat mitigation process **10** may generate **310** a security profile (e.g., security profile **350**) based, at least in part, upon system-defined consolidated platform information **236**. Through the use of security profile (e.g., security profile **350**), the user/owner/operator of computing platform **60** may be able to see that e.g., they have a security score of 605 out of a possible score of 1,000, wherein the average customer has a security score of 237. While security profile **350** in shown in the example to include several indicators that may enable a user to compare (in this example) computing platform **60** to other computing platforms, this is for illustrative purposes only and is not intended to be a limi-

tation of this disclosure, as it is understood that other configurations are possible and are considered to be within the scope of this disclosure.

[0102] Naturally, the format, appearance and content of security profile **350** may be varied greatly depending upon the design criteria and anticipated performance/use of threat mitigation process **10**. Accordingly, the appearance, format, completeness and content of security profile **350** is for illustrative purposes only and is not intended to be a limitation of this disclosure, as other configurations are possible and are considered to be within the scope of this disclosure. For example, content may be added to security profile **350**, removed from security profile **350**, and/or reformatted within security profile **350**.

[0103] Additionally, threat mitigation process **10** may obtain **312** client-defined consolidated platform information **238** for computing platform **60** from a client information source, examples of which may include but are not limited to one or more client-completed questionnaires (e.g., questionnaires **240**) and/or one or more client-deployed platform monitors (e.g., client-deployed platform monitor **242**, which may be configured to effectuate SIEM functionality). Accordingly and in this example, client-defined consolidated platform information **238** may define the security-relevant subsystems (e.g., security-relevant subsystems **226**) that the client believes are present on computing platform **60**.

[0104] When generating **310** a security profile (e.g., security profile **350**) based, at least in part, upon system-defined consolidated platform information **236**, threat mitigation process **10** may compare **314** the system-defined consolidated platform information (e.g., system-defined consolidated platform information **236**) to the client-defined consolidated platform information (e.g., client-defined consolidated platform information **238**) to define differential consolidated platform information **352** for computing platform **60**.

[0105] Differential consolidated platform information **352** may include comparison table **354** that e.g., compares computing platform **60** to other computing platforms. For example and in this particular implementation of differential consolidated platform information **352**, comparison table **354** is shown to include three columns, namely: security-relevant subsystem column **356** (that identifies the security-relevant subsystems in question); system-defined consolidated platform information column **358** (that is based upon system-defined consolidated platform information **236** and independently defines what security-relevant subsystems are present on computing platform **60**); and client-defined consolidated platform column **360** (that is based upon client-defined platform information **238** and defines what security-relevant subsystems the client believes are present on computing platform **60**). As shown within comparison table **354**, there are considerable differences between that is actually present on computing platform **60** and what is believed to be present on computing platform **60** (e.g., 1 IAM system vs. 10 IAM systems; 4,000 operating systems vs. 10,000 operating systems, 6 DNS systems vs. 10 DNS systems; 0 antivirus systems vs. 1 antivirus system, and 90 firewalls vs. 150 firewalls).

[0106] Naturally, the format, appearance and content of differential consolidated platform information **352** may be varied greatly depending upon the design criteria and anticipated performance/use of threat mitigation process **10**. Accordingly, the appearance, format, completeness and content of differential consolidated platform information **352** is for illustrative purposes only and is not intended to be a limitation of this disclosure, as other configurations are possible and are considered to be within the scope of this disclosure. For example, content may be added to differential consolidated platform information **352**, removed from differential consolidated platform information **352**, and/or reformatted within differential consolidated platform information **352**.

[0107] Referring also to FIG. **7**, threat mitigation process **10** may be configured to compare what security relevant subsystems are actually included within computing platform **60** versus what security relevant subsystems were believed to be included within computing platform **60**. As discussed above, threat mitigation process **10** may combine **308** the first system-defined platform information (e.g., system-defined platform information **232**) and the at least a second system-defined platform information (e.g., system-defined platform information **234**) to form system-defined consolidated platform information **236**.

[0108] Threat mitigation process **10** may obtain **400** system-defined consolidated platform information **236** for computing platform **60** from an independent information source, examples of which may include but are not limited to: one or more log files defined for computing platform **60** (e.g., such as those maintained by STEM system **230**); and two or more security-relevant subsystems (e.g., directly from the operating system security-relevant subsystem and the antivirus security-relevant subsystem) deployed within computing platform **60**.

[0109] Further and as discussed above, threat mitigation process **10** may obtain **312** client-defined consolidated platform information **238** for computing platform **60** from a client information source, examples of which may include but are not limited to one or more client-completed questionnaires (e.g., questionnaires **240**) and/or one or more client-deployed platform monitors (e.g., client-deployed platform monitor **242**, which may be configured to effectuate STEM functionality).

[0110] Additionally and as discussed above, threat mitigation process **10** may compare **402** system-defined consolidated platform information **236** to client-defined consolidated platform information **238** to define differential consolidated platform information **352** for computing platform **60**, wherein differential consolidated platform information **352** may include comparison table **354** that e.g., compares computing platform **60** to other computing platforms.

[0111] Threat mitigation process **10** may process **404** system-defined consolidated platform information **236** prior to comparing **402** system-defined consolidated platform information **236** to client-defined consolidated platform information **238** to define differential consolidated platform information **352** for computing platform **60**. Specifically, threat mitigation process **10** may process **404** system-defined consolidated platform information **236** so that it is comparable to client-defined consolidated platform information **238**.

[0112] For example and when processing **404** system-defined consolidated platform information **236**, threat mitigation process **10** may homogenize **406** system-defined consolidated platform information **236** prior to comparing **402** system-defined consolidated platform information **236**

to client-defined consolidated platform information **238** to define differential consolidated platform information **352** for computing platform **60**. Such homogenization **406** may result in system-defined consolidated platform information **236** and client-defined consolidated platform information **238** being comparable to each other (e.g., to accommodate for differing data nomenclatures/headers).

[0113] Further and when processing **404** system-defined consolidated platform information **236**, threat mitigation process **10** may normalize **408** system-defined consolidated platform information **236** prior to comparing **402** system-defined consolidated platform information **236** to client-defined consolidated platform information **238** to define differential consolidated platform information **352** for computing platform **60** (e.g., to accommodate for data differing scales/ranges).

[0114] Referring also to FIG. **8**, threat mitigation process **10** may be configured to compare what security relevant subsystems are actually included within computing platform **60** versus what security relevant subsystems were believed to be included within computing platform **60**.

[0115] As discussed above, threat mitigation process **10** may obtain **400** system-defined consolidated platform information **236** for computing platform **60** from an independent information source, examples of which may include but are not limited to: one or more log files defined for computing platform **60** (e.g., such as those maintained by SIEM system **230**); and two or more security-relevant subsystems (e.g., directly from the operating system security-relevant subsystem and the antivirus security-relevant subsystem) deployed within computing platform **60**

[0116] Further and as discussed above, threat mitigation process **10** may obtain **312** client-defined consolidated platform information **238** for computing platform **60** from a client information source, examples of which may include but are not limited to one or more client-completed questionnaires (e.g., questionnaires **240**) and/or one or more client-deployed platform monitors (e.g., client-deployed platform monitor **242**, which may be configured to effectuate SIEM functionality).

[0117] Threat mitigation process **10** may present **450** differential consolidated platform information **352** for computing platform **60** to a third-party, examples of which may include but are not limited to the user/owner/operator of computing platform **60**.

[0118] Additionally and as discussed above, threat mitigation process **10** may compare **402** system-defined consolidated platform information **236** to client-defined consolidated platform information **238** to define differential consolidated platform information **352** for computing platform **60**, wherein differential consolidated platform information **352** may include comparison table **354** that e.g., compares computing platform **60** to other computing platforms, wherein (and as discussed above) threat mitigation process **10** may process **404** (e.g., via homogenizing **406** and/or normalizing **408**) system-defined consolidated platform information **236** prior to comparing **402** system-defined consolidated platform information **236** to client-defined consolidated platform information **236** to define differential consolidated platform information **352** for computing platform **60**.

[0119] Computing Platform Analysis & Recommendation

[0120] As will be discussed below in greater detail, threat mitigation process **10** may be configured to e.g., analyze & display the vulnerabilities of computing platform **60**.

[0121] Referring also to FIG. **9**, threat mitigation process **10** may be configured to make recommendations concerning security relevant subsystems that are missing from computing platform **60**. As discussed above, threat mitigation process **10** may obtain **500** consolidated platform information for computing platform **60** to identify one or more deployed security-relevant subsystems **226** (e.g., CDN (i.e., Content Delivery Network) systems; DAM (i.e., Database Activity Monitoring) systems; UBA (i.e., User Behavior Analytics) systems; MDM (i.e., Mobile Device Management) systems; IAM (i.e., Identity and Access Management) systems; DNS (i.e., Domain Name Server) systems, antivirus systems, operating systems, data lakes; data logs; security-relevant software applications; security-relevant hardware systems; and resources external to the computing platform). This consolidated platform information may be obtained from an independent information source (e.g., such as SIEM system **230** that may provide system-defined consolidated platform information **236**) and/or may be obtained from a client information source (e.g., such as questionnaires **240** that may provide client-defined consolidated platform information **238**).

[0122] Referring also to FIG. **10**, threat mitigation process **10** may process **506** the consolidated platform information (e.g., system-defined consolidated platform information **236** and/or client-defined consolidated platform information **238**) to identify one or more non-deployed security-relevant subsystems (within computing platform **60**) and may then generate **508** a list of ranked & recommended security-relevant subsystems (e.g., non-deployed security-relevant subsystem list **550**) that ranks the one or more non-deployed security-relevant subsystems.

[0123] For this particular illustrative example, non-deployed security-relevant subsystem list **550** is shown to include column **552** that identifies six non-deployed security-relevant subsystems, namely: a CDN subsystem, a WAF subsystem, a DAM subsystem; a UBA subsystem; a API subsystem, and an MDM subsystem.

[0124] When generating **508** a list of ranked & recommended security-relevant subsystems (e.g., non-deployed security-relevant subsystem list **550**) that ranks the one or more non-deployed security-relevant subsystems, threat mitigation process **10** may rank **510** the one or more non-deployed security-relevant subsystems (e.g., a CDN subsystem, a WAF subsystem, a DAM subsystem; a UBA subsystem; a API subsystem, and an MDM subsystem) based upon the anticipated use of the one or more non-deployed security-relevant subsystems within computing platform **60**. This ranking **510** of the non-deployed security-relevant subsystems (e.g., a CDN subsystem, a WAF subsystem, a DAM subsystem; a UBA subsystem; a API subsystem, and an MDM subsystem) may be agnostic in nature and may be based on the functionality/effectiveness of the non-deployed security-relevant subsystems and the anticipated manner in which their implementation may impact the functionality/security of computing platform **60**.

[0125] Threat mitigation process **10** may provide **512** the list of ranked & recommended security-relevant subsystems (e.g., non-deployed security-relevant subsystem list **550**) to a third-party, examples of which may include but are not limited to a user/owner/operator of computing platform **60**.

[0126] Additionally, threat mitigation process **10** may identify **514** a comparative for at least one of the non-deployed security-relevant subsystems (e.g., a CDN subsystem, a WAF subsystem, a DAM subsystem; a UBA subsystem; a API subsystem, and an MDM subsystem) defined within the list of ranked & recommended security-relevant subsystems (e.g., non-deployed security-relevant subsystem list **550**). This comparative may include vendor customers in a specific industry comparative and/or vendor customers in any industry comparative.

[0127] For example and in addition to column **552**, non-deployed security-relevant subsystem list **550** may include columns **554**, **556** for defining the comparatives for the six non-deployed security-relevant subsystems, namely: a CDN subsystem, a WAF subsystem, a DAM subsystem; a UBA subsystem; a API subsystem, and an MDM subsystem. Specifically, column **554** is shown to define comparatives concerning vendor customers that own the non-deployed security-relevant subsystems in a specific industry (i.e., the same industry as the user/owner/operator of computing platform **60**). Additionally, column **556** is shown to define comparatives concerning vendor customers that own the non-deployed security-relevant subsystems in any industry (i.e., not necessarily the same industry as the user/owner/operator of computing platform **60**). For example and concerning the comparatives of the WAF subsystem: 33% of the vendor customers in the same industry as the user/owner/operator of computing platform **60** deploy a WAF subsystem; while 71% of the vendor customers in any industry deploy a WAF subsystem.

[0128] Naturally, the format, appearance and content of non-deployed security-relevant subsystem list **550** may be varied greatly depending upon the design criteria and anticipated performance/use of threat mitigation process **10**. Accordingly, the appearance, format, completeness and content of non-deployed security-relevant subsystem list **550** is for illustrative purposes only and is not intended to be a limitation of this disclosure, as other configurations are possible and are considered to be within the scope of this disclosure. For example, content may be added to non-deployed security-relevant subsystem list **550**, removed from non-deployed security-relevant subsystem list **550**, and/or reformatted within non-deployed security-relevant subsystem list **550**.

[0129] Referring also to FIG. **11**, threat mitigation process **10** may be configured to compare the current capabilities to the possible capabilities of computing platform **60**. As discussed above, threat mitigation process **10** may obtain **600** consolidated platform information to identify current security-relevant capabilities for computing platform **60**. This consolidated platform information may be obtained from an independent information source (e.g., such as STEM system **230** that may provide system-defined consolidated platform information **236**) and/or may be obtained from a client information source (e.g., such as questionnaires **240** that may provide client-defined consolidated platform information **238**. Threat mitigation process **10** may then determine **606** possible security-relevant capabilities for computing platform **60** (i.e., the difference between the current security-relevant capabilities of computing platform **60** and the possible security-relevant capabilities of computing platform **60**. For example, the possible security-relevant capabilities may concern the possible security-relevant capabilities of computing platform **60** using the currently-deployed security-relevant subsystems. Additionally/alternatively, the possible security-relevant capabilities may concern the possible security-relevant capabilities of computing platform **60** using one or more supplemental security-relevant subsystems.

[0130] Referring also to FIG. **12** and as will be explained below, threat mitigation process **10** may generate **608** comparison information **650** that compares the current security-relevant capabilities of computing platform **60** to the possible security-relevant capabilities of computing platform **60** to identify security-relevant deficiencies. Comparison information **650** may include graphical comparison information, such as multi-axial graphical comparison information that simultaneously illustrates a plurality of security-relevant deficiencies.

[0131] For example, comparison information **650** may define (in this particular illustrative example) graphical comparison information that include five axes (e.g. axes **652**, **654**, **656**, **658**, **660**) that correspond to five particular types of computer threats. Comparison information **650** includes origin **662**, the point at which computing platform **60** has no protection with respect to any of the five types of computer threats that correspond to axes **652**, **654**, **656**, **658**, **660**. Accordingly, as the capabilities of computing platform **60** are increased to counter a particular type of computer threat, the data point along the corresponding axis is proportionately displaced from origin **652**.

[0132] As discussed above, threat mitigation process **10** may obtain **600** consolidated platform information to identify current security-relevant capabilities for computing platform **60**. Concerning such current security-relevant capabilities for computing platform **60**, these current security-relevant capabilities are defined by data points **664**, **666**, **668**, **670**, **672**, the combination of which define bounded area **674**. Bounded area **674** (in this example) defines the current security-relevant capabilities of computing platform **60**.

[0133] Further and as discussed above, threat mitigation process **10** may determine **606** possible security-relevant capabilities for computing platform **60** (i.e., the difference between the current security-relevant capabilities of computing platform **60** and the possible security-relevant capabilities of computing platform **60**.

[0134] As discussed above, the possible security-relevant capabilities may concern the possible security-relevant capabilities of computing platform **60** using the currently-deployed security-relevant subsystems. For example, assume that the currently-deployed security relevant subsystems are not currently being utilized to their full potential. Accordingly, certain currently-deployed security relevant subsystems may have certain features that are available but are not utilized and/or disabled. Further, certain currently-deployed security relevant subsystems may have expanded features available if additional licensing fees are paid. Therefore and concerning such possible security-relevant capabilities of computing platform **60** using the currently-deployed security-relevant subsystems, data points **676**, **678**, **680**, **682**, **684** may define bounded area **686** (which represents the full capabilities of the currently-deployed security-relevant subsystems within computing platform **60**).

[0135] Further and as discussed above, the possible security-relevant capabilities may concern the possible security-relevant capabilities of computing platform **60** using one or

more supplemental security-relevant subsystems. For example, assume that supplemental security-relevant subsystems are available for the deployment within computing platform **60**. Therefore and concerning such possible security-relevant capabilities of computing platform **60** using such supplemental security-relevant subsystems, data points **688**, **690**, **692**, **694**, **696** may define bounded area **698** (which represents the total capabilities of computing platform **60** when utilizing the full capabilities of the currently-deployed security-relevant subsystems and any supplemental security-relevant subsystems).

[0136] Naturally, the format, appearance and content of comparison information **650** may be varied greatly depending upon the design criteria and anticipated performance/use of threat mitigation process **10**. Accordingly, the appearance, format, completeness and content of comparison information **650** is for illustrative purposes only and is not intended to be a limitation of this disclosure, as other configurations are possible and are considered to be within the scope of this disclosure. For example, content may be added to comparison information **650**, removed from comparison information **650**, and/or reformatted within comparison information **650**.

[0137] Referring also to FIG. **13**, threat mitigation process **10** may be configured to generate a threat context score for computing platform **60**. As discussed above, threat mitigation process **10** may obtain **600** consolidated platform information to identify current security-relevant capabilities for computing platform **60**. This consolidated platform information may be obtained from an independent information source (e.g., such as SIEM system **230** that may provide system-defined consolidated platform information **236**) and/or may be obtained from a client information source (e.g., such as questionnaires **240** that may provide client-defined consolidated platform information **238**. As will be discussed below in greater detail, threat mitigation process **10** may determine **700** comparative platform information that identifies security-relevant capabilities for a comparative platform, wherein this comparative platform information may concern vendor customers in a specific industry (i.e., the same industry as the user/owner/operator of computing platform **60**) and/or vendor customers in any industry (i.e., not necessarily the same industry as the user/owner/operator of computing platform **60**).

[0138] Referring also to FIG. **14** and as will be discussed below, threat mitigation process **10** may generate **702** comparison information **750** that compares the current security-relevant capabilities of computing platform **60** to the comparative platform information determined **700** for the comparative platform to identify a threat context indicator for computing platform **60**, wherein comparison information **750** may include graphical comparison information **752**.

[0139] Graphical comparison information **752** (which in this particular example is a bar chart) may identify one or more of: a current threat context score **754** for a client (e.g., the user/owner/operator of computing platform **60**); a maximum possible threat context score **756** for the client (e.g., the user/owner/operator of computing platform **60**); a threat context score **758** for one or more vendor customers in a specific industry (i.e., the same industry as the user/owner/operator of computing platform **60**); and a threat context score **760** for one or more vendor customers in any industry (i.e., not necessarily the same industry as the user/owner/operator of computing platform **60**).

[0140] Naturally, the format, appearance and content of comparison information **750** may be varied greatly depending upon the design criteria and anticipated performance/use of threat mitigation process **10**. Accordingly, the appearance, format, completeness and content of comparison information **750** is for illustrative purposes only and is not intended to be a limitation of this disclosure, as other configurations are possible and are considered to be within the scope of this disclosure. For example, content may be added to comparison information **750**, removed from comparison information **750**, and/or reformatted within comparison information **750**.

[0141] Computing Platform Monitoring & Mitigation

[0142] As will be discussed below in greater detail, threat mitigation process **10** may be configured to e.g., monitor the operation and performance of computing platform **60**.

[0143] Referring also to FIG. **15**, threat mitigation process **10** may be configured to monitor the health of computing platform **60** and provide feedback to a third-party concerning the same. Threat mitigation process **10** may obtain **800** hardware performance information **244** concerning hardware (e.g., server computers, desktop computers, laptop computers, switches, firewalls, routers, gateways, WAPs, and NASs), deployed within computing platform **60**. Hardware performance information **244** may concern the operation and/or functionality of one or more hardware systems (e.g., server computers, desktop computers, laptop computers, switches, firewalls, routers, gateways, WAPs, and NASs) deployed within computing platform **60**.

[0144] Threat mitigation process **10** may obtain **802** platform performance information **246** concerning the operation of computing platform **60**. Platform performance information **246** may concern the operation and/or functionality of computing platform **60**.

[0145] When obtaining **802** platform performance information concerning the operation of computing platform **60**, threat mitigation process **10** may (as discussed above): obtain **400** system-defined consolidated platform information **236** for computing platform **60** from an independent information source (e.g., SIEM system **230**); obtain **312** client-defined consolidated platform information **238** for computing platform **60** from a client information (e.g., questionnaires **240**); and present **450** differential consolidated platform information **352** for computing platform **60** to a third-party, examples of which may include but are not limited to the user/owner/operator of computing platform **60**.

[0146] When obtaining **802** platform performance information concerning the operation of computing platform **60**, threat mitigation process **10** may (as discussed above): obtain **500** consolidated platform information for computing platform **60** to identify one or more deployed security-relevant subsystems **226** (e.g., CDN (i.e., Content Delivery Network) systems; DAM (i.e., Database Activity Monitoring) systems; UBA (i.e., User Behavior Analytics) systems; MDM (i.e., Mobile Device Management) systems; IAM (i.e., Identity and Access Management) systems; DNS (i.e., Domain Name Server) systems, antivirus systems, operating systems, data lakes; data logs; security-relevant software applications; security-relevant hardware systems; and resources external to the computing platform); process **506** the consolidated platform information (e.g., system-defined consolidated platform information **236** and/or client-defined consolidated platform information **238**) to identify one or more non-deployed security-relevant subsystems (within

computing platform **60**); generate **508** a list of ranked & recommended security-relevant subsystems (e.g., non-deployed security-relevant subsystem list **550**) that ranks the one or more non-deployed security-relevant subsystems; and provide **514** the list of ranked & recommended security-relevant subsystems (e.g., non-deployed security-relevant subsystem list **550**) to a third-party, examples of which may include but are not limited to a user/owner/operator of computing platform **60**.

[0147] When obtaining **802** platform performance information concerning the operation of computing platform **60**, threat mitigation process **10** may (as discussed above): obtain **600** consolidated platform information to identify current security-relevant capabilities for the computing platform; determine **606** possible security-relevant capabilities for computing platform **60**; and generate **608** comparison information **650** that compares the current security-relevant capabilities of computing platform **60** to the possible security-relevant capabilities of computing platform **60** to identify security-relevant deficiencies.

[0148] When obtaining **802** platform performance information concerning the operation of computing platform **60**, threat mitigation process **10** may (as discussed above): obtain **600** consolidated platform information to identify current security-relevant capabilities for computing platform **60**; determine **700** comparative platform information that identifies security-relevant capabilities for a comparative platform; and generate **702** comparison information **750** that compares the current security-relevant capabilities of computing platform **60** to the comparative platform information determined **700** for the comparative platform to identify a threat context indicator for computing platform **60**.

[0149] Threat mitigation process **10** may obtain **804** application performance information **248** concerning one or more applications (e.g., operating systems, user applications, security application, and utility application) deployed within computing platform **60**. Application performance information **248** may concern the operation and/or functionality of one or more software applications (e.g., operating systems, user applications, security application, and utility application) deployed within computing platform **60**.

[0150] Referring also to FIG. **16**, threat mitigation process **10** may generate **806** holistic platform report (e.g., holistic platform reports **850**, **852**) concerning computing platform **60** based, at least in part, upon hardware performance information **244**, platform performance information **246** and application performance information **248**. Threat mitigation process **10** may be configured to receive e.g., hardware performance information **244**, platform performance information **246** and application performance information **248** at regular intervals (e.g., continuously, every minute, every ten minutes, etc.).

[0151] As illustrated, holistic platform reports **850**, **852** may include various pieces of content such as e.g., thought clouds that identity topics/issues with respect to computing platform **60**, system logs that memorialize identified issues within computing platform **60**, data sources providing information to computing system **60**, and so on. The holistic platform report (e.g., holistic platform reports **850**, **852**) may identify one or more known conditions concerning the computing platform; and threat mitigation process **10** may effectuate **808** one or more remedial operations concerning the one or more known conditions.

[0152] For example, assume that the holistic platform report (e.g., holistic platform reports **850**, **852**) identifies that computing platform **60** is under a DoS (i.e., Denial of Services) attack. In computing, a denial-of-service attack (DoS attack) is a cyber-attack in which the perpetrator seeks to make a machine or network resource unavailable to its intended users by temporarily or indefinitely disrupting services of a host connected to the Internet. Denial of service is typically accomplished by flooding the targeted machine or resource with superfluous requests in an attempt to overload systems and prevent some or all legitimate requests from being fulfilled.

[0153] In response to detecting such a DoS attack, threat mitigation process **10** may effectuate **808** one or more remedial operations. For example and with respect to such a DoS attack, threat mitigation process **10** may effectuate **808** e.g., a remedial operation that instructs WAF (i.e., Web Application Firewall) **212** to deny all incoming traffic from the identified attacker based upon e.g., protocols, ports or the originating IP addresses.

[0154] Threat mitigation process **10** may also provide **810** the holistic report (e.g., holistic platform reports **850**, **852**) to a third-party, examples of which may include but are not limited to a user/owner/operator of computing platform **60**.

[0155] Naturally, the format, appearance and content of the holistic platform report (e.g., holistic platform reports **850**, **852**) may be varied greatly depending upon the design criteria and anticipated performance/use of threat mitigation process **10**. Accordingly, the appearance, format, completeness and content of the holistic platform report (e.g., holistic platform reports **850**, **852**) is for illustrative purposes only and is not intended to be a limitation of this disclosure, as other configurations are possible and are considered to be within the scope of this disclosure. For example, content may be added to the holistic platform report (e.g., holistic platform reports **850**, **852**), removed from the holistic platform report (e.g., holistic platform reports **850**, **852**), and/or reformatted within the holistic platform report (e.g., holistic platform reports **850**, **852**).

[0156] Referring also to FIG. **17**, threat mitigation process **10** may be configured to monitor computing platform **60** for the occurrence of a security event and (in the event of such an occurrence) gather artifacts concerning the same. For example, threat mitigation process **10** may detect **900** a security event within computing platform **60** based upon identified suspect activity. Examples of such security events may include but are not limited to: DDoS events, DoS events, phishing events, spamming events, malware events, web attacks, and exploitation events.

[0157] When detecting **900** a security event (e.g., DDoS events, DoS events, phishing events, spamming events, malware events, web attacks, and exploitation events) within computing platform **60** based upon identified suspect activity, threat mitigation process **10** may monitor **902** a plurality of sources to identify suspect activity within computing platform **60**.

[0158] For example, assume that threat mitigation process **10** detects **900** a security event within computing platform **60**. Specifically, assume that threat mitigation process **10** is monitoring **902** a plurality of sources (e.g., the various log files maintained by STEM system **230**). And by monitoring **902** such sources, assume that threat mitigation process **10** detects **900** the receipt of inbound content (via an API) from a device having an IP address located in Uzbekistan; the

subsequent opening of a port within WAF (i.e., Web Application Firewall) **212**; and the streaming of content from a computing device within computing platform **60** through that recently-opened port in WAF (i.e., Web Application Firewall) **212** and to a device having an IP address located in Moldova.

[0159] Upon detecting **900** such a security event within computing platform **60**, threat mitigation process **10** may gather **904** artifacts (e.g., artifacts **250**) concerning the above-described security event. When gathering **904** artifacts (e.g., artifacts **250**) concerning the above-described security event, threat mitigation process **10** may gather **906** artifacts concerning the security event from a plurality of sources associated with the computing platform, wherein examples of such plurality of sources may include but are not limited to the various log files maintained by SIEM system **230**, and the various log files directly maintained by the security-relevant subsystems.

[0160] Once the appropriate artifacts (e.g., artifacts **250**) are gathered **904**, threat mitigation process **10** may assign **908** a threat level to the above-described security event based, at least in part, upon the artifacts (e.g., artifacts **250**) gathered **904**.

[0161] When assigning **908** a threat level to the above-described security event, threat mitigation process **10** may assign **910** a threat level using artificial intelligence/machine learning. As discussed above and with respect to artificial intelligence/machine learning being utilized to process data sets, an initial probabilistic model may be defined, wherein this initial probabilistic model may be subsequently (e.g., iteratively or continuously) modified and revised, thus allowing the probabilistic models and the artificial intelligence systems (e.g., probabilistic process **56**) to "learn" so that future probabilistic models may be more precise and may explain more complex data sets. As further discussed above, probabilistic process **56** may define an initial probabilistic model for accomplishing a defined task (e.g., the analyzing of information **58**), wherein the probabilistic model may be utilized to go from initial observations about information **58** (e.g., as represented by the initial branches of a probabilistic model) to conclusions about information **58** (e.g., as represented by the leaves of a probabilistic model). Accordingly and through the use of probabilistic process **56**, massive data sets concerning security events may be processed so that a probabilistic model may be defined (and subsequently revised) to assign **910** a threat level to the above-described security event.

[0162] Once assigned **910** a threat level, threat mitigation process **10** may execute **912** a remedial action plan (e., remedial action plan **252**) based, at least in part, upon the assigned threat level.

[0163] For example and when executing **912** a remedial action plan, threat mitigation process **10** may allow 914 the above-described suspect activity to continue when e.g., threat mitigation process **10** assigns **908** a "low" threat level to the above-described security event (e.g., assuming that it is determined that the user of the local computing device is streaming video of his daughter's graduation to his parents in Moldova).

[0164] Further and when executing **912** a remedial action plan, threat mitigation process **10** may generate **916** a security event report (e.g., security event report **254**) based, at least in part, upon the artifacts (e.g., artifacts **250**) gathered **904**; and provide **918** the security event report (e.g., security event report **254**) to an analyst (e.g., analyst **256**) for further review when e.g., threat mitigation process **10** assigns **908** a "moderate" threat level to the above-described security event (e.g., assuming that it is determined that while the streaming of the content is concerning, the content is low value and the recipient is not a known bad actor).

[0165] Further and when executing **912** a remedial action plan, threat mitigation process **10** may autonomously execute **920** a threat mitigation plan (shutting down the stream and closing the port) when e.g., threat mitigation process **10** assigns **908** a "severe" threat level to the above-described security event (e.g., assuming that it is determined that the streaming of the content is very concerning, as the content is high value and the recipient is a known bad actor).

[0166] Additionally, threat mitigation process **10** may allow 922 a third-party (e.g., the user/owner/operator of computing platform **60**) to manually search for artifacts within computing platform **60**. For example, the third-party (e.g., the user/owner/operator of computing platform **60**) may be able to search the various information resources include within computing platform **60**, examples of which may include but are not limited to the various log files maintained by STEM system **230**, and the various log files directly maintained by the security-relevant subsystems within computing platform **60**.

[0167] Computing Platform Aggregation & Searching

[0168] As will be discussed below in greater detail, threat mitigation process **10** may be configured to e.g., aggregate data sets and allow for unified search of those data sets.

[0169] Referring also to FIG. **18**, threat mitigation process **10** may be configured to consolidate multiple separate and discrete data sets to form a single, aggregated data set. For example, threat mitigation process **10** may establish **950** connectivity with a plurality of security-relevant subsystems (e.g., security-relevant subsystems **226**) within computing platform **60**. As discussed above, examples of security-relevant subsystems **226** may include but are not limited to: CDN (i.e., Content Delivery Network) systems; DAM (i.e., Database Activity Monitoring) systems; UBA (i.e., User Behavior Analytics) systems; MDM (i.e., Mobile Device Management) systems; IAM (i.e., Identity and Access Management) systems; DNS (i.e., Domain Name Server) systems, Antivirus systems, operating systems, data lakes; data logs; security-relevant software applications; security-relevant hardware systems; and resources external to the computing platform.

[0170] When establishing **950** connectivity with a plurality of security-relevant subsystems, threat mitigation process **10** may utilize **952** at least one application program interface (e.g., API Gateway **224**) to access at least one of the plurality of security-relevant subsystems. For example, a 1^{st} API gateway may be utilized to access CDN (i.e., Content Delivery Network) system; a 2^{nd} API gateway may be utilized to access DAM (i.e., Database Activity Monitoring) system; a 3^{rd} API gateway may be utilized to access UBA (i.e., User Behavior Analytics) system; a 4^{th} API gateway may be utilized to access MDM (i.e., Mobile Device Management) system; a 5^{th} API gateway may be utilized to access IAM (i.e., Identity and Access Management) system; and a 6^{th} API gateway may be utilized to access DNS (i.e., Domain Name Server) system.

[0171] Threat mitigation process **10** may obtain **954** at least one security-relevant information set (e.g., a log file) from each of the plurality of security-relevant subsystems

(e.g., CDN system; DAM system; UBA system; MDM system; IAM system; and DNS system), thus defining plurality of security-relevant information sets **258**. As would be expected, plurality of security-relevant information sets **258** may utilize a plurality of different formats and/or a plurality of different nomenclatures. Accordingly, threat mitigation process **10** may combine **956** plurality of security-relevant information sets **258** to form an aggregated security-relevant information set **260** for computing platform **60**.

[0172] When combining **956** plurality of security-relevant information sets **258** to form aggregated security-relevant information set **260**, threat mitigation process **10** may homogenize **958** plurality of security-relevant information sets **258** to form aggregated security-relevant information set **260**. For example, threat mitigation process **10** may process one or more of security-relevant information sets **258** so that they all have a common format, a common nomenclature, and/or a common structure.

[0173] Once threat mitigation process **10** combines **956** plurality of security-relevant information sets **258** to form an aggregated security-relevant information set **260** for computing platform **60**, threat mitigation process **10** may enable **960** a third-party (e.g., the user/owner/operator of computing platform **60**) to access aggregated security-relevant information set **260** and/or enable **962** a third-party (e.g., the user/owner/operator of computing platform **60**) to search aggregated security-relevant information set **260**.

[0174] Referring also to FIG. **19**, threat mitigation process **10** may be configured to enable the searching of multiple separate and discrete data sets using a single search operation. For example and as discussed above, threat mitigation process **10** may establish **950** connectivity with a plurality of security-relevant subsystems (e., security-relevant subsystems **226**) within computing platform **60**. As discussed above, examples of security-relevant subsystems **226** may include but are not limited to: CDN (i.e., Content Delivery Network) systems; DAM (i.e., Database Activity Monitoring) systems; UBA (i.e., User Behavior Analytics) systems; MDM (i.e., Mobile Device Management) systems; IAM (i.e., Identity and Access Management) systems; DNS (i.e., Domain Name Server) systems, Antivirus systems, operating systems, data lakes; data logs; security-relevant software applications; security-relevant hardware systems; and resources external to the computing platform.

[0175] When establishing **950** connectivity with a plurality of security-relevant subsystems, threat mitigation process **10** may utilize **952** at least one application program interface (e.g., API Gateway **224**) to access at least one of the plurality of security-relevant subsystems. For example, a 1^{st} API gateway may be utilized to access CDN (i.e., Content Delivery Network) system; a 2^{nd} API gateway may be utilized to access DAM (i.e., Database Activity Monitoring) system; a 3^{rd} API gateway may be utilized to access UBA (i.e., User Behavior Analytics) system; a 4^{th} API gateway may be utilized to access MDM (i.e., Mobile Device Management) system; a 5^{th} API gateway may be utilized to access IAM (i.e., Identity and Access Management) system; and a 6^{th} API gateway may be utilized to access DNS (i.e., Domain Name Server) system.

[0176] Threat mitigation process **10** may receive **1000** unified query **262** from a third-party (e.g., the user/owner/operator of computing platform **60**) concerning the plurality of security-relevant subsystems. As discussed above, examples of security-relevant subsystems **226** may include but are not limited to: CDN (i.e., Content Delivery Network) systems; DAM (i.e., Database Activity Monitoring) systems; UBA (i.e., User Behavior Analytics) systems; MDM (i.e., Mobile Device Management) systems; IAM (i.e., Identity and Access Management) systems; DNS (i.e., Domain Name Server) systems, Antivirus systems, operating systems, data lakes; data logs; security-relevant software applications; security-relevant hardware systems; and resources external to the computing platform.

[0177] Threat mitigation process **10** may distribute **1002** at least a portion of unified query **262** to the plurality of security-relevant subsystems, resulting in the distribution of plurality of queries **264** to the plurality of security-relevant subsystems. For example, assume that a third-party (e.g., the user/owner/operator of computing platform **60**) wishes to execute a search concerning the activity of a specific employee. Accordingly, the third-party (e.g., the user/owner/operator of computing platform **60**) may formulate the appropriate unified query (e.g., unified query **262**) that defines the employee name, the computing device(s) of the employee, and the date range of interest. Unified query **262** may then be parsed to form plurality of queries **264**, wherein a specific query (within plurality of queries **264**) may be defined for each of the plurality of security-relevant subsystems and provided to the appropriate security-relevant subsystems. For example, a 1^{st} query may be included within plurality of queries **264** and provided to CDN (i.e., Content Delivery Network) system; a 2^{nd} query may be included within plurality of queries **264** and provided to DAM (i.e., Database Activity Monitoring) system; a 3^{rd} query may be included within plurality of queries **264** and provided to UBA (i.e., User Behavior Analytics) system; a 4^{th} query may be included within plurality of queries **264** and provided to MDM (i.e., Mobile Device Management) system; a 5^{th} query may be included within plurality of queries **264** and provided to IAM (i.e., Identity and Access Management) system; and a 6^{th} query may be included within plurality of queries **264** and provided to DNS (i.e., Domain Name Server) system.

[0178] Threat mitigation process **10** may effectuate **1004** at least a portion of unified query **262** on each of the plurality of security-relevant subsystems to generate plurality of result sets **266**. For example, the 1^{st} query may be executed on CDN (i.e., Content Delivery Network) system to produce a 1^{st} result set; the 2^{nd} query may be executed on DAM (i.e., Database Activity Monitoring) system to produce a 2^{nd} result set; the 3^{rd} query may be executed on UBA (i.e., User Behavior Analytics) system to produce a 3^{rd} result set; the 4^{th} query may be executed on MDM (i.e., Mobile Device Management) system to produce a 4^{th} result set; the 5^{th} query may be executed on IAM (i.e., Identity and Access Management) system to produce a 5^{th} result set; and the 6^{th} query may executed on DNS (i.e., Domain Name Server) system to produce a 6^{th} result set.

[0179] Threat mitigation process **10** may receive **1006** plurality of result sets **266** from the plurality of security-relevant subsystems. Threat mitigation process **10** may then combine **1008** plurality of result sets **266** to form unified query result **268**. When combining **1008** plurality of result sets **266** to form unified query result **268**, threat mitigation process **10** may homogenize **1010** plurality of result sets **266** to form unified query result **268**. For example, threat mitigation process **10** may process one or more discrete result sets included within plurality of result sets **266** so that the

discrete result sets within plurality of result sets **266** all have a common format, a common nomenclature, and/or a common structure. Threat mitigation process **10** may then provide **1012** unified query result **268** to the third-party (e.g., the user/owner/operator of computing platform **60**).

[0180] Referring also to FIG. **20**, threat mitigation process **10** may be configured to utilize artificial intelligence/machine learning to automatically consolidate multiple separate and discrete data sets to form a single, aggregated data set. For example and as discussed above, threat mitigation process **10** may establish **950** connectivity with a plurality of security-relevant subsystems (e.g., security-relevant subsystems **226**) within computing platform **60**. As discussed above, examples of security-relevant subsystems **226** may include but are not limited to: CDN (i.e., Content Delivery Network) systems; DAM (i.e., Database Activity Monitoring) systems; UBA (i.e., User Behavior Analytics) systems; MDM (i.e., Mobile Device Management) systems; IAM (i.e., Identity and Access Management) systems; DNS (i.e., Domain Name Server) systems, Antivirus systems, operating systems, data lakes; data logs; security-relevant software applications; security-relevant hardware systems; and resources external to the computing platform.

[0181] As discussed above and when establishing **950** connectivity with a plurality of security-relevant subsystems, threat mitigation process **10** may utilize **952** at least one application program interface (e.g., API Gateway **224**) to access at least one of the plurality of security-relevant subsystems. For example, a 1^{st} API gateway may be utilized to access CDN (i.e., Content Delivery Network) system; a 2^{nd} API gateway may be utilized to access DAM (i.e., Database Activity Monitoring) system; a 3^{rd} API gateway may be utilized to access UBA (i.e., User Behavior Analytics) system; a 4^{th} API gateway may be utilized to access MDM (i.e., Mobile Device Management) system; a 5^{th} API gateway may be utilized to access IAM (i.e., Identity and Access Management) system; and a 6^{th} API gateway may be utilized to access DNS (i.e., Domain Name Server) system.

[0182] As discussed above, threat mitigation process **10** may obtain **954** at least one security-relevant information set (e.g., a log file) from each of the plurality of security-relevant subsystems (e.g., CDN system; DAM system; UBA system; MDM system; IAM system; and DNS system), thus defining plurality of security-relevant information sets **258**. As would be expected, plurality of security-relevant information sets **258** may utilize a plurality of different formats and/or a plurality of different nomenclatures.

[0183] Threat mitigation process **10** may process **1050** plurality of security-relevant information sets **258** using artificial learning/machine learning to identify one or more commonalities amongst plurality of security-relevant information sets **258**. As discussed above and with respect to artificial intelligence/machine learning being utilized to process data sets, an initial probabilistic model may be defined, wherein this initial probabilistic model may be subsequently (e.g., iteratively or continuously) modified and revised, thus allowing the probabilistic models and the artificial intelligence systems (e.g., probabilistic process **56**) to "learn" so that future probabilistic models may be more precise and may explain more complex data sets. As further discussed above, probabilistic process **56** may define an initial probabilistic model for accomplishing a defined task (e.g., the analyzing of information **58**), wherein the probabilistic model may be utilized to go from initial observations about information **58** (e.g., as represented by the initial branches of a probabilistic model) to conclusions about information **58** (e.g., as represented by the leaves of a probabilistic model). Accordingly and through the use of probabilistic process **56**, plurality of security-relevant information sets **258** may be processed so that a probabilistic model may be defined (and subsequently revised) to identify one or more commonalities (e.g., common headers, common nomenclatures, common data ranges, common data types, common formats, etc.) amongst plurality of security-relevant information sets **258**. When processing **1050** plurality of security-relevant information sets **258** using artificial learning/machine learning to identify one or more commonalities amongst plurality of security-relevant information sets **258**, threat mitigation process **10** may utilize **1052** a decision tree (e.g., probabilistic model **100**) based, at least in part, upon one or more previously-acquired security-relevant information sets.

[0184] Threat mitigation process **10** may combine **1054** plurality of security-relevant information sets **258** to form aggregated security-relevant information set **260** for computing platform **60** based, at least in part, upon the one or more commonalities identified.

[0185] When combining **1054** plurality of security-relevant information sets **258** to form aggregated security-relevant information set **260** for computing platform **60** based, at least in part, upon the one or more commonalities identified, threat mitigation process **10** may homogenize **1056** plurality of security-relevant information sets **258** to form aggregated security-relevant information set **260**. For example, threat mitigation process **10** may process one or more of security-relevant information sets **258** so that they all have a common format, a common nomenclature, and/or a common structure.

[0186] Once threat mitigation process **10** combines **1054** plurality of security-relevant information sets **258** to form an aggregated security-relevant information set **260** for computing platform **60**, threat mitigation process **10** may enable **1058** a third-party (e.g., the user/owner/operator of computing platform **60**) to access aggregated security-relevant information set **260** and/or enable **1060** a third-party (e.g., the user/owner/operator of computing platform **60**) to search aggregated security-relevant information set **260**.

[0187] Threat Event Information Updating

[0188] As will be discussed below in greater detail, threat mitigation process **10** may be configured to be updated concerning threat event information.

[0189] Referring also to FIG. **21**, threat mitigation process **10** may be configured to receive updated threat event information for security-relevant subsystems **226**. For example, threat mitigation process **10** may receive **1100** updated threat event information **270** concerning computing platform **60**, wherein updated threat event information **270** may define one or more of: updated threat listings; updated threat definitions; updated threat methodologies; updated threat sources; and updated threat strategies. Threat mitigation process **10** may enable **1102** updated threat event information **270** for use with one or more security-relevant subsystems **226** within computing platform **60**. As discussed above, examples of security-relevant subsystems **226** may include but are not limited to: CDN (i.e., Content Delivery Network) systems; DAM (i.e., Database Activity Monitoring) systems; UBA (i.e., User Behavior Analytics) systems; MDM (i.e., Mobile Device Management) systems; IAM (i.e., Identity and Access Management) systems; DNS (i.e.,

Domain Name Server) systems, Antivirus systems, operating systems, data lakes; data logs; security-relevant software applications; security-relevant hardware systems; and resources external to the computing platform.

[0190] When enabling **1102** updated threat event information **270** for use with one or more security-relevant subsystems **226** within computing platform **60**, threat mitigation process **10** may install **1104** updated threat event information **270** on one or more security-relevant subsystems **226** within computing platform **60**.

[0191] Threat mitigation process **10** may retroactively apply **1106** updated threat event information **270** to previously-generated information associated with one or more security-relevant subsystems **226**.

[0192] When retroactively apply **1106** updated threat event information **270** to previously-generated information associated with one or more security-relevant subsystems **226**, threat mitigation process **10** may: apply **1108** updated threat event information **270** to one or more previously-generated log files (not shown) associated with one or more security-relevant subsystems **226**; apply **1110** updated threat event information **270** to one or more previously-generated data files (not shown) associated with one or more security-relevant subsystems **226**; and apply **1112** updated threat event information **270** to one or more previously-generated application files (not shown) associated with one or more security-relevant subsystems **226**.

[0193] Additionally,/alternatively, threat mitigation process **10** may proactively apply **1114** updated threat event information **270** to newly-generated information associated with one or more security-relevant subsystems **226**.

[0194] When proactively applying **1114** updated threat event information **270** to newly-generated information associated with one or more security-relevant subsystems **226**, threat mitigation process **10** may: apply **1116** updated threat event information **270** to one or more newly-generated log files (not shown) associated with one or more security-relevant subsystems **226**; apply **1118** updated threat event information **270** to one or more newly-generated data files (not shown) associated with one or more security-relevant subsystems **226**; and apply **1120** updated threat event information **270** to one or more newly-generated application files (not shown) associated with one or more security-relevant subsystems **226**.

[0195] Referring also to FIG. **22**, threat mitigation process **10** may be configured to receive updated threat event information **270** for security-relevant subsystems **226**. For example and as discussed above, threat mitigation process **10** may receive **1100** updated threat event information **270** concerning computing platform **60**, wherein updated threat event information **270** may define one or more of: updated threat listings; updated threat definitions; updated threat methodologies; updated threat sources; and updated threat strategies. Further and as discussed above, threat mitigation process **10** may enable **1102** updated threat event information **270** for use with one or more security-relevant subsystems **226** within computing platform **60**. As discussed above, examples of security-relevant subsystems **226** may include but are not limited to: CDN (i.e., Content Delivery Network) systems; DAM (i.e., Database Activity Monitoring) systems; UBA (i.e., User Behavior Analytics) systems; MDM (i.e., Mobile Device Management) systems; IAM (i.e., Identity and Access Management) systems; DNS (i.e., Domain Name Server) systems, Antivirus systems, operating systems, data lakes; data logs; security-relevant software applications; security-relevant hardware systems; and resources external to the computing platform.

[0196] As discussed above and when enabling **1102** updated threat event information **270** for use with one or more security-relevant subsystems **226** within computing platform **60**, threat mitigation process **10** may install **1104** updated threat event information **270** on one or more security-relevant subsystems **226** within computing platform **60**.

[0197] Sometimes, it may not be convenient and/or efficient to immediately apply updated threat event information **270** to security-relevant subsystems **226**. Accordingly, threat mitigation process **10** may schedule **1150** the application of updated threat event information **270** to previously-generated information associated with one or more security-relevant subsystems **226**.

[0198] When scheduling **1150** the application of updated threat event information **270** to previously-generated information associated with one or more security-relevant subsystems **226**, threat mitigation process **10** may: schedule **1152** the application of updated threat event information **270** to one or more previously-generated log files (not shown) associated with one or more security-relevant subsystems **226**; schedule **1154** the application of updated threat event information **270** to one or more previously-generated data files (not shown) associated with one or more security-relevant subsystems **226**; and schedule **1156** the application of updated threat event information **270** to one or more previously-generated application files (not shown) associated with one or more security-relevant subsystems **226**.

[0199] Additionally,/alternatively, threat mitigation process **10** may schedule **1158** the application of the updated threat event information to newly-generated information associated with the one or more security-relevant subsystems.

[0200] When scheduling **1158** the application of updated threat event information **270** to newly-generated information associated with one or more security-relevant subsystems **226**, threat mitigation process **10** may: schedule **1160** the application of updated threat event information **270** to one or more newly-generated log files (not shown) associated with one or more security-relevant subsystems **226**; schedule **1162** the application of updated threat event information **270** to one or more newly-generated data files (not shown) associated with one or more security-relevant subsystems **226**; and schedule **1164** the application of updated threat event information **270** to one or more newly-generated application files (not shown) associated with one or more security-relevant subsystems **226**.

[0201] Referring also to FIGS. **23-24**, threat mitigation process **10** may be configured to initially display analytical data, which may then be manipulated/updated to include automation data. For example, threat mitigation process **10** may display **1200** initial security-relevant information **1250** that includes analytical information (e.g., thought cloud **1252**). Examples of such analytical information may include but is not limited to one or more of: investigative information; and hunting information.

[0202] Investigative Information (a portion of analytical information): Unified searching and/or automated searching, such as e.g., a security event occurring and searches being performed to gather artifacts concerning that security event.

[0203] Hunt Information (a portion of analytical information): Targeted searching/investigations, such as the moni-

toring and cataloging of the videos that an employee has watched or downloaded over the past 30 days.

[0204] Threat mitigation process **10** may allow **1202** a third-party (e.g., the user/owner/operator of computing platform **60**) to manipulate initial security-relevant information **1250** with automation information.

[0205] Automate Information (a portion of automation): The execution of a single (and possibly simple) action one time, such as the blocking an IP address from accessing computing platform **60** whenever such an attempt is made.

[0206] Orchestrate Information (a portion of automation): The execution of a more complex batch (or series) of tasks, such as sensing an unauthorized download via an API and a) shutting down the API, adding the requesting IP address to a blacklist, and closing any ports opened for the requestor.

[0207] When allowing **1202** a third-party (e.g., the user/owner/operator of computing platform **60**) to manipulate initial security-relevant information **1250** with automation information, threat mitigation process **10** may allow **1204** a third-party (e.g., the user/owner/operator of computing platform **60**) to select the automation information to add to initial security-relevant information **1250** to generate revised security-relevant information **1250'**. For example and when allowing **1204** a third-party (e.g., the user/owner/operator of computing platform **60**) to select the automation information to add to initial security-relevant information **1250** to generate revised security-relevant information **1250'**, threat mitigation process **10** may allow **1206** the third-party (e.g., the user/owner/operator of computing platform **60**) to choose a specific type of automation information from a plurality of automation information types.

[0208] For example, the third-party (e.g., the user/owner/operator of computing platform **60**) may choose to add/initiate the automation information to generate revised security-relevant information **1250'**. Accordingly, threat mitigation process **10** may render selectable options (e.g., selectable buttons **1254**, **1256**) that the third-party (e.g., the user/owner/operator of computing platform **60**) may select to manipulate initial security-relevant information **1250** with automation information to generate revised security-relevant information **1250'**. For this particular example, the third-party (e.g., the user/owner/operator of computing platform **60**) may choose two different options to manipulate initial security-relevant information **1250**, namely: "block ip" or "search", both of which will result in threat mitigation process **10** generating **1208** revised security-relevant information **1250'** (that includes the above-described automation information).

[0209] When generating **1208** revised security-relevant information **1250'** (that includes the above-described automation information), threat mitigation process **10** may combine **1210** the automation information (that results from selecting "block IP" or "search") and initial security-relevant information **1250** to generate and render **1212** revised security-relevant information **1250'**.

[0210] When rendering **1212** revised security-relevant information **1250'**, threat mitigation process **10** may render **1214** revised security-relevant information **1250'** within interactive report **1258**.

[0211] Training Routine Generation and Execution

[0212] As will be discussed below in greater detail, threat mitigation process **10** may be configured to allow for the manual or automatic generation of training routines, as well as the execution of the same.

[0213] Referring also to FIG. **25**, threat mitigation process **10** may be configured to allow for the manual generation of testing routine **272**. For example, threat mitigation process **10** may define **1300** training routine **272** for a specific attack (e.g., a Denial of Services attack) of computing platform **60**. Specifically, threat mitigation process **10** may generate **1302** a simulation of the specific attack (e.g., a Denial of Services attack) by executing training routine **272** within a controlled test environment, an example of which may include but is not limited to virtual machine **274** executed on a computing device (e.g., computing device **12**).

[0214] When generating **1302** a simulation of the specific attack (e.g., a Denial of Services attack) by executing training routine **272** within the controlled test environment (e.g., virtual machine **274**), threat mitigation process **10** may render **1304** the simulation of the specific attack (e.g., a Denial of Services attack) on the controlled test environment (e.g., virtual machine **274**).

[0215] Threat mitigation process **10** may allow **1306** a trainee (e.g., trainee **276**) to view the simulation of the specific attack (e.g., a Denial of Services attack) and may allow **1308** the trainee (e.g., trainee **276**) to provide a trainee response (e.g., trainee response **278**) to the simulation of the specific attack (e.g., a Denial of Services attack). For example, threat mitigation process **10** may execute training routine **272**, which trainee **276** may "watch" and provide trainee response **278**.

[0216] Threat mitigation process **10** may then determine **1310** the effectiveness of trainee response **278**, wherein determining **1310** the effectiveness of the trainee response may include threat mitigation process **10** assigning **1312** a grade (e.g., a letter grade or a number grade) to trainee response **278**.

[0217] Referring also to FIG. **26**, threat mitigation process **10** may be configured to allow for the automatic generation of testing routine **272**. For example, threat mitigation process **10** may utilize **1350** artificial intelligence/machine learning to define training routine **272** for a specific attack (e.g., a Denial of Services attack) of computing platform **60**.

[0218] As discussed above and with respect to artificial intelligence/machine learning being utilized to process data sets, an initial probabilistic model may be defined, wherein this initial probabilistic model may be subsequently (e.g., iteratively or continuously) modified and revised, thus allowing the probabilistic models and the artificial intelligence systems (e.g., probabilistic process **56**) to "learn" so that future probabilistic models may be more precise and may explain more complex data sets. As further discussed above, probabilistic process **56** may define an initial probabilistic model for accomplishing a defined task (e.g., the analyzing of information **58**), wherein the probabilistic model may be utilized to go from initial observations about information **58** (e.g., as represented by the initial branches of a probabilistic model) to conclusions about information **58** (e.g., as represented by the leaves of a probabilistic model). Accordingly and through the use of probabilistic process **56**, information may be processed so that a probabilistic model may be defined (and subsequently revised) to define training routine **272** for a specific attack (e.g., a Denial of Services attack) of computing platform **60**.

[0219] When using **1350** artificial intelligence/machine learning to define training routine **272** for a specific attack (e.g., a Denial of Services attack) of computing platform **60**, threat mitigation process **10** may process **1352** security-

relevant information to define training routine **272** for specific attack (e.g., a Denial of Services attack) of computing platform **60**. Further and when using **1350** artificial intelligence/machine learning to define training routine **272** for a specific attack (e.g., a Denial of Services attack) of computing platform **60**, threat mitigation process **10** may utilize **1354** security-relevant rules to define training routine **272** for a specific attack (e.g., a Denial of Services attack) of computing platform **60**. Accordingly, security-relevant information that e.g., defines the symptoms of e.g., a Denial of Services attack and security-relevant rules that define the behavior of e.g., a Denial of Services attack may be utilized by threat mitigation process **10** when defining training routine **272**.

[0220] As discussed above, threat mitigation process **10** may generate **1302** a simulation of the specific attack (e.g., a Denial of Services attack) by executing training routine **272** within a controlled test environment, an example of which may include but is not limited to virtual machine **274** executed on a computing device (e.g., computing device **12**.

[0221] Further and as discussed above, when generating **1302** a simulation of the specific attack (e.g., a Denial of Services attack) by executing training routine **272** within the controlled test environment (e.g., virtual machine **274**), threat mitigation process **10** may render **1304** the simulation of the specific attack (e.g., a Denial of Services attack) on the controlled test environment (e.g., virtual machine **274**).

[0222] Threat mitigation process **10** may allow **1306** a trainee (e.g., trainee **276**) to view the simulation of the specific attack (e.g., a Denial of Services attack) and may allow **1308** the trainee (e.g., trainee **276**) to provide a trainee response (e.g., trainee response **278**) to the simulation of the specific attack (e.g., a Denial of Services attack). For example, threat mitigation process **10** may execute training routine **272**, which trainee **276** may "watch" and provide trainee response **278**.

[0223] Threat mitigation process **10** may utilize **1356** artificial intelligence/machine learning to revise training routine **272** for the specific attack (e.g., a Denial of Services attack) of computing platform **60** based, at least in part, upon trainee response **278**.

[0224] As discussed above, threat mitigation process **10** may then determine **1310** the effectiveness of trainee response **278**, wherein determining **1310** the effectiveness of the trainee response may include threat mitigation process **10** assigning **1312** a grade (e.g., a letter grade or a number grade) to trainee response **278**.

[0225] Referring also to FIG. **27**, threat mitigation process **10** may be configured to allow a trainee to choose their training routine. For example mitigation process **10** may allow **1400** a third-party (e.g., the user/owner/operator of computing platform **60**) to select a training routine for a specific attack (e.g., a Denial of Services attack) of computing platform **60**, thus defining a selected training routine. When allowing **1400** a third-party (e.g., the user/owner/operator of computing platform **60**) to select a training routine for a specific attack (e.g., a Denial of Services attack) of computing platform **60**, threat mitigation process **10** may allow **1402** the third-party (e.g., the user/owner/operator of computing platform **60**) to choose a specific training routine from a plurality of available training routines. For example, the third-party (e.g., the user/owner/operator of computing platform **60**) may be able to select a specific type of attack (e.g., DDoS events, DoS events, phishing events, spamming events, malware events, web attacks, and exploitation events) and/or select a specific training routine (that may or may not disclose the specific type of attack).

[0226] Once selected, threat mitigation process **10** may analyze **1404** the requirements of the selected training routine (e.g., training routine **272**) to determine a quantity of entities required to effectuate the selected training routine (e.g., training routine **272**), thus defining one or more required entities. For example, assume that training routine **272** has three required entities (e.g., an attacked device and two attacking devices). According, threat mitigation process **10** may generate **1406** one or more virtual machines (e.g., such as virtual machine **274**) to emulate the one or more required entities. In this particular example, threat mitigation process **10** may generate **1406** three virtual machines, a first VM for the attacked device, a second VM for the first attacking device and a third VM for the second attacking device. As is known in the art, a virtual machine (VM) is an virtual emulation of a physical computing system. Virtual machines may be based on computer architectures and may provide the functionality of a physical computer, wherein their implementations may involve specialized hardware, software, or a combination thereof.

[0227] Threat mitigation process **10** may generate **1408** a simulation of the specific attack (e.g., a Denial of Services attack) by executing the selected training routine (e.g., training routine **272**). When generating **1408** the simulation of the specific attack (e.g., a Denial of Services attack) by executing the selected training routine (e.g., training routine **272**), threat mitigation process **10** may render **1410** the simulation of the specific attack (e.g., a Denial of Services attack) by executing the selected training routine (e.g., training routine **272**) within a controlled test environment (e.g., such as virtual machine **274**).

[0228] As discussed above, threat mitigation process **10** may allow **1306** a trainee (e.g., trainee **276**) to view the simulation of the specific attack (e.g., a Denial of Services attack) and may allow **1308** the trainee (e.g., trainee **276**) to provide a trainee response (e.g., trainee response **278**) to the simulation of the specific attack (e.g., a Denial of Services attack). For example, threat mitigation process **10** may execute training routine **272**, which trainee **276** may "watch" and provide trainee response **278**.

[0229] Further and as discussed above, threat mitigation process **10** may then determine **1310** the effectiveness of trainee response **278**, wherein determining **1310** the effectiveness of the trainee response may include threat mitigation process **10** assigning **1312** a grade (e.g., a letter grade or a number grade) to trainee response **278**.

[0230] When training is complete, threat mitigation process **10** may cease **1412** the simulation of the specific attack (e.g., a Denial of Services attack), wherein ceasing **1412** the simulation of the specific attack (e.g., a Denial of Services attack) may include threat mitigation process **10** shutting down **1414** the one or more virtual machines (e.g., the first VM for the attacked device, the second VM for the first attacking device and the third VM for the second attacking device).

[0231] Information Routing

[0232] As will be discussed below in greater detail, threat mitigation process **10** may be configured to route information based upon whether the information is more threat-pertinent or less threat-pertinent.

[0233] Referring also to FIG. **28**, threat mitigation process **10** may be configured to route more threat-pertinent content in a specific manner. For example, threat mitigation process **10** may receive **1450** platform information (e.g., log files) from a plurality of security-relevant subsystems (e.g., security-relevant subsystems **226**). As discussed above, examples of security-relevant subsystems **226** may include but are not limited to: CDN (i.e., Content Delivery Network) systems; DAM (i.e., Database Activity Monitoring) systems; UBA (i.e., User Behavior Analytics) systems; MDM (i.e., Mobile Device Management) systems; IAM (i.e., Identity and Access Management) systems; DNS (i.e., Domain Name Server) systems, Antivirus systems, operating systems, data lakes; data logs; security-relevant software applications; security-relevant hardware systems; and resources external to the computing platform.

[0234] Threat mitigation process **10** may process **1452** this platform information (e.g., log files) to generate processed platform information. And when processing **1452** this platform information (e.g., log files) to generate processed platform information, threat mitigation process **10** may: parse **1454** the platform information (e.g., log files) into a plurality of subcomponents (e.g., columns, rows, etc.) to allow for compensation of varying formats and/or nomenclature; enrich **1456** the platform information (e.g., log files) by including supplemental information from external information resources; and/or utilize **1458** artificial intelligence/machine learning (in the manner described above) to identify one or more patterns/trends within the platform information (e.g., log files).

[0235] Threat mitigation process **10** may identify **1460** more threat-pertinent content **280** included within the processed content, wherein identifying **1460** more threat-pertinent content **280** included within the processed content may include processing **1462** the processed content to identify actionable processed content that may be used by a threat analysis engine (e.g., SIEM system **230**) for correlation purposes. Threat mitigation process **10** may route **1464** more threat-pertinent content **280** to this threat analysis engine (e.g., SIEM system **230**).

[0236] Referring also to FIG. **29**, threat mitigation process **10** may be configured to route less threat-pertinent content in a specific manner. For example and as discussed above, threat mitigation process **10** may receive **1450** platform information (e.g., log files) from a plurality of security-relevant subsystems (e.g., security-relevant subsystems **226**). As discussed above, examples of security-relevant subsystems **226** may include but are not limited to: CDN (i.e., Content Delivery Network) systems; DAM (i.e., Database Activity Monitoring) systems; UBA (i.e., User Behavior Analytics) systems; MDM (i.e., Mobile Device Management) systems; IAM (i.e., Identity and Access Management) systems; DNS (i.e., Domain Name Server) systems, Antivirus systems, operating systems, data lakes; data logs; security-relevant software applications; security-relevant hardware systems; and resources external to the computing platform

[0237] Further and as discussed above, threat mitigation process **10** may process **1452** this platform information (e.g., log files) to generate processed platform information. And when processing **1452** this platform information (e.g., log files) to generate processed platform information, threat mitigation process **10** may: parse **1454** the platform information (e.g., log files) into a plurality of subcomponents (e.g., columns, rows, etc.) to allow for compensation of varying formats and/or nomenclature; enrich **1456** the platform information (e.g., log files) by including supplemental information from external information resources; and/or utilize **1458** artificial intelligence/machine learning (in the manner described above) to identify one or more patterns/trends within the platform information (e.g., log files).

[0238] Threat mitigation process **10** may identify **1500** less threat-pertinent content **282** included within the processed content, wherein identifying **1500** less threat-pertinent content **282** included within the processed content may include processing **1502** the processed content to identify non-actionable processed content that is not usable by a threat analysis engine (e.g., SIEM system **230**) for correlation purposes. Threat mitigation process **10** may route **1504** less threat-pertinent content **282** to a long term storage system (e.g., long term storage system **284**). Further, threat mitigation process **10** may be configured to allow **1506** a third-party (e.g., the user/owner/operator of computing platform **60**) to access and search long term storage system **284**.

[0239] Automated Analysis

[0240] As will be discussed below in greater detail, threat mitigation process **10** may be configured to automatically analyze a detected security event.

[0241] Referring also to FIG. **30**, threat mitigation process **10** may be configured to automatically classify and investigate a detected security event. As discussed above and in response to a security event being detected, threat mitigation process **10** may obtain **1550** one or more artifacts (e.g., artifacts **250**) concerning the detected security event. Examples of such a detected security event may include but are not limited to one or more of: access auditing; anomalies; authentication; denial of services; exploitation; malware; phishing; spamming; reconnaissance; and web attack. These artifacts (e.g., artifacts **250**) may be obtained **1550** from a plurality of sources associated with the computing platform, wherein examples of such plurality of sources may include but are not limited to the various log files maintained by STEM system **230**, and the various log files directly maintained by the security-relevant subsystems

[0242] Threat mitigation process **10** may obtain **1552** artifact information (e.g., artifact information **286**) concerning the one or more artifacts (e.g., artifacts **250**), wherein artifact information **286** may be obtained from information resources include within (or external to) computing platform **60**.

[0243] For example and when obtaining **1552** artifact information **286** concerning the one or more artifacts (e.g., artifacts **250**), threat mitigation process **10** may obtain **1554** artifact information **286** concerning the one or more artifacts (e.g., artifacts **250**) from one or more investigation resources (such as third-party resources that may e.g., provide information on known bad actors).

[0244] Once the investigation is complete, threat mitigation process **10** may generate **1556** a conclusion (e.g., conclusion **288**) concerning the detected security event (e.g., a Denial of Services attack) based, at least in part, upon the detected security event (e.g., a Denial of Services attack), the one or more artifacts (e.g., artifacts **250**), and artifact information **286**. Threat mitigation process **10** may document **1558** the conclusion (e.g., conclusion **288**), report **1560** the conclusion (e.g., conclusion **288**) to a third-party (e.g., the user/owner/operator of computing platform **60**). Further,

threat mitigation process **10** may obtain **1562** supplemental artifacts and artifact information (if needed to further the investigation).

[0245] While the system is described above as being computer-implemented, this is for illustrative purposes only and is not intended to be a limitation of this disclosure, as other configurations are possible and are considered to be within the scope of this disclosure. For example, some or all of the above-described system may be implemented by a human being.

Concept 1:

[0246] Referring also to FIGS. **31-32** and as discussed above, threat mitigation process **10** may be configured to monitor computing platform **60** for the occurrence of a security event. Examples of such security events may include but are not limited to: DDoS events, DoS events, phishing events, spamming events, malware events, web attacks, and exploitation events. For example, threat mitigation process **10** may detect **900** a security event within the computing platform (e.g., computing platform **60**) based upon identified suspect activity.

[0247] When detecting **900** a security event (e.g., DDoS events, DoS events, phishing events, spamming events, malware events, web attacks, and exploitation events) within computing platform **60** based upon identified suspect activity, threat mitigation process **10** may monitor **902** a plurality of sources to identify suspect activity within computing platform **60**.

[0248] For example, assume that threat mitigation process **10** detects **900** a security event within computing platform **60**. Specifically, assume that threat mitigation process **10** is monitoring **902** a plurality of sources (e.g., the various log files maintained by STEM system **230**). And by monitoring **902** such sources, assume that threat mitigation process **10** detects **900** the receipt of inbound content (via an API) from a device having an IP address located in Uzbekistan; the subsequent opening of a port within WAF (i.e., Web Application Firewall) **212**; and the streaming of content from a computing device within computing platform **60** through that recently-opened port in WAF (i.e., Web Application Firewall) **212** and to a device having an IP address located in Moldova.

[0249] To aid a third-party (e.g., the user/owner/operator of computing platform **60**) in analyzing the security event (e.g., DDoS events, DoS events, phishing events, spamming events, malware events, web attacks, and exploitation events), threat mitigation process **10** may render **1600** a threat mitigation user interface (e.g., threat mitigation user interface **1650**) that identifies objects (e.g., objects **1652**, **1654**, **1656**, **1658**, **1660**, **1662**, **1664**, **1666**) within a computing platform (e.g., computing platform **60**) in response to the security event.

[0250] Threat mitigation process **10** may define **1602** the objects (e.g., objects **1652**, **1654**, **1656**, **1658**, **1660**, **1662**, **1664**, **1666**) within the computing platform (e.g., computing platform **60**) in response to the security event (e.g., DDoS events, DoS events, phishing events, spamming events, malware events, web attacks, and exploitation events).

[0251] When defining **1602** the objects (e.g., objects **1652**, **1654**, **1656**, **1658**, **1660**, **1662**, **1664**, **1666**) within the computing platform (e.g., computing platform **60**) in response to the security event, threat mitigation process **10** may gather **1604** the objects (e.g., objects **1652**, **1654**, **1656**, **1658**, **1660**, **1662**, **1664**, **1666**) within the computing platform (e.g., computing platform **60**) in response to the security event from a plurality of sources associated with the computing platform (e.g., computing platform **60**), wherein examples of such plurality of sources may include but are not limited to the various log files maintained by SIEM system **230**, and the various log files directly maintained by the security-relevant subsystems.

[0252] Examples of these objects (e.g., objects **1652**, **1654**, **1656**, **1658**, **1660**, **1662**, **1664**, **1666**) may include but are not limited to: time-based traffic map object **1652**, initial source IP address object **1654**, destination port object **1656**, destination IP address external object **1658**, service splunk object **1660**, service "CB Response" object **1662**, outbound traffic by port graphical object **1664**, and data sources graphical object **1666**

[0253] Threat mitigation process **10** may enable **1606** a third-party (e.g., the user/owner/operator of computing platform **60**) to select an object within the threat mitigation user interface (e.g., threat mitigation user interface **1650**), thus defining a selected object. For example, assume that the third-party (e.g., the user/owner/operator of computing platform **60**) selects destination port object **1656** within threat mitigation user interface (e.g., threat mitigation user interface **1650**), thus defining a selected object (i.e., destination port object **1656**).

[0254] In response to the third-party (e.g., the user/owner/operator of computing platform **60**) selecting an object (i.e., destination port object **1656**) within the threat mitigation user interface (e.g., threat mitigation user interface **1650**), threat mitigation process **10** may render **1608** an inspection window (e.g., inspection window **1668**) that defines object information concerning the selected object (i.e., destination port object **1656**). Examples of this object information may include but are not limited to: total outbound traffic **1670**; time-based outbound traffic graph **1672**; source IP address (i.e., 10.12.11.180) **1674**; destination IP address (i.e., 8.12.11.180) **1676**; and destination IP address (i.e., 12.12.11.181) **1678**.

[0255] The inspection window (e.g., inspection window **1668**) may be rendered **1608** in various ways. For example, inspection window **1668** may be a popup inspection window, such as a freestanding window that pops up next to (or on top of) threat mitigation user interface **1650**. Alternatively, inspection window **1668** may be a slide out inspection window, such as a window that slides out from behind threat mitigation user interface **1650**.

[0256] Threat mitigation process **10** may enable **1610** the third-party (e.g., the user/owner/operator of computing platform **60**) to select a portion (e.g., destination IP address 8.12.11.180) of the object information rendered within the inspection window (e.g., inspection window **1668**), thus defining a selected portion.

[0257] In response to the third-party (e.g., the user/owner/operator of computing platform **60**) selecting a portion (e.g., destination IP address 8.12.11.180) of the object information rendered within the inspection window (e.g., inspection window **1668**), threat mitigation process **10** may render **1612** detailed information (e.g., detailed information **1680** concerning the selected portion (e.g., destination IP address 8.12.11.180). Detailed information (e.g., detailed information **1680**) may be information that is highly pertinent to the selected portion (e.g., destination IP address 8.12.11.180). As the selected portion is a destination having an IP address

of 8.12.11.180, examples of this detailed information (e.g., detailed information **1680**) may include but are not limited to: the total quantity of data provided to that destination IP address; a graph of the rate at which that destination IP address has been receiving data over a period of time (e.g., the past 24 hours); a list of all devices within computing platform **60** that have provided data to this destination IP address; etc.

Concept 2:

[0258] As discussed above, threat mitigation process **10** may be configured to monitor computing platform **60** for the occurrence of a security event. Examples of such security events may include but are not limited to: DDoS events, DoS events, phishing events, spamming events, malware events, web attacks, and exploitation events. For example, threat mitigation process **10** may detect **900** a security event within the computing platform (e.g., computing platform **60**) based upon identified suspect activity

[0259] As discussed above and when detecting **900** a security event (e.g., DDoS events, DoS events, phishing events, spamming events, malware events, web attacks, and exploitation events) within computing platform **60** based upon identified suspect activity, threat mitigation process **10** may monitor **902** a plurality of sources to identify suspect activity within computing platform **60**.

[0260] As discussed above and to aid a third-party (e.g., the user/owner/operator of computing platform **60**) in analyzing the security event (e.g., DDoS events, DoS events, phishing events, spamming events, malware events, web attacks, and exploitation events), threat mitigation process **10** may render **1600** a threat mitigation user interface (e.g., threat mitigation user interface **1650**) that identifies objects (e.g., objects **1652**, **1654**, **1656**, **1658**, **1660**, **1662**, **1664**, **1666**) within a computing platform (e.g., computing platform **60**) in response to the security event (e.g., computing platform **60**).

[0261] Threat mitigation process **10** may define **1602** the objects (e.g., objects **1652**, **1654**, **1656**, **1658**, **1660**, **1662**, **1664**, **1666**) within the computing platform (e.g., computing platform **60**) in response to the security event (e.g., DDoS events, DoS events, phishing events, spamming events, malware events, web attacks, and exploitation events). When defining **1602** the objects (e.g., objects **1652**, **1654**, **1656**, **1658**, **1660**, **1662**, **1664**, **1666**) within the computing platform (e.g., computing platform **60**) in response to the security event, threat mitigation process **10** may gather **1604** the objects (e.g., objects **1652**, **1654**, **1656**, **1658**, **1660**, **1662**, **1664**, **1666**) within the computing platform (e.g., computing platform **60**) in response to the security event (e.g., computing platform **60**) from a plurality of sources associated with the computing platform (e.g., computing platform **60**), wherein examples of such plurality of sources may include but are not limited to the various log files maintained by SIEM system **230**, and the various log files directly maintained by the security-relevant subsystems.

[0262] As discussed above, examples of these objects (e.g., objects **1652**, **1654**, **1656**, **1658**, **1660**, **1662**, **1664**, **1666**) may include but are not limited to: time-based traffic map object **1652**, initial source IP address object **1654**, destination port object **1656**, destination IP address external object **1658**, service splunk object **1660**, service "CB Response" object **1662**, outbound traffic by port graphical object **1664**, and data sources graphical object **1666**

[0263] Threat mitigation process **10** may enable **1606** a third-party (e.g., the user/owner/operator of computing platform **60**) to select an object within the threat mitigation user interface (e.g., threat mitigation user interface **1650**), thus defining a selected object. For example, assume that the third-party (e.g., the user/owner/operator of computing platform **60**) selects destination port object **1656** within threat mitigation user interface (e.g., threat mitigation user interface **1650**), thus defining a selected object (i.e., destination port object **1656**).

[0264] As discussed above and in response to a third-party (e.g., the user/owner/operator of computing platform **60**) selecting an object (i.e., destination port object **1656**) within the threat mitigation user interface (e.g., threat mitigation user interface **1650**), threat mitigation process **10** may render **1608** an inspection window (e.g., inspection window **1668**) that defines object information concerning the selected object (i.e., destination port object **1656**). Examples of this object information may include but are not limited to: total outbound traffic **1670**; time-based outbound traffic graph **1672**; source IP address **1674**; destination IP address **1676**; and destination IP address **1678**.

[0265] Threat mitigation process **10** may enable **1614** a third-party (e.g., the user/owner/operator of computing platform **60**) to effectuate a specific targeted action that is based, at least in part, upon the object information (e.g., total outbound traffic **1670**; time-based outbound traffic graph **1672**; source IP address **1674**; destination IP address **1676**; and destination IP address **1678**) defined within the inspection window (e.g., inspection window **1668**).

[0266] When enabling **1614** a third-party (e.g., the user/owner/operator of computing platform **60**) to effectuate a specific targeted action that is based, at least in part, upon the object information (e.g., total outbound traffic **1670**; time-based outbound traffic graph **1672**; source IP address **1674**; destination IP address **1676**; and destination IP address **1678**) defined within the inspection window (e.g., inspection window **1668**), threat mitigation process **10** may render **1616** an action list (e.g., action list **1682**) that defines one or more targeted actions (e.g., targeted actions **1684**) that are based, at least in part, upon the object information (e.g., total outbound traffic **1670**; time-based outbound traffic graph **1672**; source IP address **1674**; destination IP address **1676**; and destination IP address **1678**) defined within the inspection window (e.g., inspection window **1668**). For example and upon the third-party (e.g., the user/owner/operator of computing platform **60**) selecting icon **1686**, threat mitigation process **10** may render **1616** action list **1682** that defines targeted actions **1684** that are based, at least in part, upon the object information defined within the inspection window (e.g., inspection window **1668**).

[0267] When enabling **1614** a third-party (e.g., the user/owner/operator of computing platform **60**) to effectuate a specific targeted action that is based, at least in part, upon the object information (e.g., total outbound traffic **1670**; time-based outbound traffic graph **1672**; source IP address **1674**; destination IP address **1676**; and destination IP address **1678**) defined within the inspection window (e.g., inspection window **1668**), threat mitigation process **10** may enable **1618** the third-party (e.g., the user/owner/operator of computing platform **60**) to select the specific targeted action from the one or more targeted actions (e.g., targeted actions **1684**) defined within the action list (e.g., action list **1682**).

[0268] As discussed above, the inspection window (e.g., inspection window **1668**) defines object information concerning the selected object (i.e., destination port object **1656**). Accordingly, action list **1682** may define one or more targeted actions (e.g., targeted actions **1684**) that are based, at least in part, upon the object information (e.g., total outbound traffic **1670**; time-based outbound traffic graph **1672**; source IP address **1674**; destination IP address **1676**; and destination IP address **1678**) defined within the inspection window (e.g., inspection window **1668**), all of which is based upon (in this example) destination port object **1656** (namely destination port **445**).

[0269] According to the object information (e.g., total outbound traffic **1670**; time-based outbound traffic graph **1672**; source IP address **1674**; destination IP address **1676**; and destination IP address **1678**) defined within the inspection window (e.g., inspection window **1668**), destination IP address **1676** (namely 8.12.11.180) is the destination IP address that is receiving the majority of the data being provided by destination port **445**. Therefore and with respect to the action list (e.g., action list **1682**) that defines one or more targeted actions (e.g., targeted actions **1684**) that are based, at least in part, upon the object information (e.g., total outbound traffic **1670**; time-based outbound traffic graph **1672**; source IP address **1674**; destination IP address **1676**; and destination IP address **1678**) defined within the inspection window (e.g., inspection window **1668**), one example of such a targeted action may include but is not limited to the ability of the third-party (e.g., the user/owner/operator of computing platform **60**) to shut down all data being provided to destination IP address **1676** (namely 8.12.11.180).

[0270] Accordingly, threat mitigation process **10** may enable **1618** the third-party (e.g., the user/owner/operator of computing platform **60**) to select this specific targeted action within the action list (e.g., action list **1682**) and shut down all data transmission to IP address 8.12.11.180).

Concept 3:

[0271] As discussed above, threat mitigation process **10** may be configured to monitor computing platform **60** for the occurrence of a security event. Examples of such security events may include but are not limited to: DDoS events, DoS events, phishing events, spamming events, malware events, web attacks, and exploitation events. For example, threat mitigation process **10** may detect **900** a security event within the computing platform (e.g., computing platform **60**) based upon identified suspect activity

[0272] As discussed above and when detecting **900** a security event (e.g., DDoS events, DoS events, phishing events, spamming events, malware events, web attacks, and exploitation events) within computing platform **60** based upon identified suspect activity, threat mitigation process **10** may monitor **902** a plurality of sources to identify suspect activity within computing platform **60**.

[0273] As discussed above and to aid a third-party (e.g., the user/owner/operator of computing platform **60**) in analyzing the security event (e.g., DDoS events, DoS events, phishing events, spamming events, malware events, web attacks, and exploitation events), threat mitigation process **10** may render **1600** a threat mitigation user interface (e.g., threat mitigation user interface **1650**) that identifies objects (e.g., objects **1652**, **1654**, **1656**, **1658**, **1660**, **1662**, **1664**, **1666**) within a computing platform (e.g., computing platform **60**) in response to the security event (e.g., computing platform **60**).

[0274] As discussed above, threat mitigation process **10** may enable **1606** a third-party (e.g., the user/owner/operator of computing platform **60**) to select an object within the threat mitigation user interface (e.g., threat mitigation user interface **1650**), thus defining a selected object. For example, assume that the third-party (e.g., the user/owner/operator of computing platform **60**) selects destination port object **1656** within threat mitigation user interface (e.g., threat mitigation user interface **1650**), thus defining a selected object (i.e., destination port object **1656**).

[0275] In response to a third-party (e.g., the user/owner/operator of computing platform **60**) selecting an object (i.e., destination port object **1656**) within the threat mitigation user interface (e.g., threat mitigation user interface **1650**), threat mitigation process **10** may render **1608** an inspection window (e.g., inspection window **1668**) that defines object information concerning the selected object (i.e., destination port object **1656**). Examples of this object information may include but are not limited to: total outbound traffic **1670**; time-based outbound traffic graph **1672**; source IP address **1674**; destination IP address **1676**; and destination IP address **1678**.

[0276] To aid a third-party (e.g., the user/owner/operator of computing platform **60**) in analyzing the security event (e.g., DDoS events, DoS events, phishing events, spamming events, malware events, web attacks, and exploitation events), threat mitigation process **10** may enable **1620** a third-party (e.g., the user/owner/operator of computing platform **60**) to gather artifacts (e.g., artifacts **1688**) concerning an object (e.g., one or more of objects **1652**, **1654**, **1656**, **1658**, **1660**, **1662**, **1664**, **1666**) within the threat mitigation user interface (e.g., threat mitigation user interface **1650**). Examples of the artifacts (e.g., artifacts **1688**) may include but are not limited to: raw data (e.g., port usage and statistics); screen shots (e.g., all or a portion of threat mitigation user interface **1650** and/or inspection window **1668**); graphics (e.g., port graphical object **1664**); notes (e.g., notes authored by the third-party); annotations (e.g., annotations made by the third-party on other artifacts); audio recordings (e.g., voice recordings made by the third-party); and video recordings (e.g., video recordings made by the third-party).

[0277] When enabling **1620** a third-party (e.g., the user/owner/operator of computing platform **60**) to gather artifacts (e.g., artifacts **1688**) concerning an object (e.g., one or more of objects **1652**, **1654**, **1656**, **1658**, **1660**, **1662**, **1664**, **1666**) within the threat mitigation user interface (e.g., threat mitigation user interface **1650**), threat mitigation process **10** may also enable **1622** the third-party (e.g., the user/owner/operator of computing platform **60**) to gather artifacts (e.g., artifacts **1688**) concerning an object (e.g., total outbound traffic **1670**; time-based outbound traffic graph **1672**; source IP address **1674**; destination IP address **1676**; and destination IP address **1678**) within the inspection window (e.g., inspection window **1668**).

[0278] Threat mitigation process **10** may enable **1624** the third-party (e.g., the user/owner/operator of computing platform **60**) to store these artifacts (e.g., artifacts **1688**) within a defined storage location. For example and upon the third-party (e.g., the user/owner/operator of computing platform **60**) selecting icon **1690**, threat mitigation process **10** may

e.g., render folder **1692** within which the artifacts (e.g., artifacts **1688**) may be stored (e.g., by dragging and dropping artifacts **1688** into folder **1692** or manually storing artifacts within folder **1692**). Additionally, threat mitigation process **10** may enable **1626** the third-party (e.g., the user/owner/operator of computing platform **60**) to provide the artifacts (e.g., artifacts **1688**) to another party (e.g., supervisors and external response teams) via e.g., email, a messaging application, an FTP server, transfer to a remote storage location, etc.

Concept 4:

[0279] As discussed above and to aid a third-party (e.g., the user/owner/operator of computing platform **60**) in analyzing the security event (e.g., DDoS events, DoS events, phishing events, spamming events, malware events, web attacks, and exploitation events), threat mitigation process **10** may render **1600** a threat mitigation user interface (e.g., threat mitigation user interface **1650**) that identifies objects (e.g., objects **1652**, **1654**, **1656**, **1658**, **1660**, **1662**, **1664**, **1666**) within a computing platform (e.g., computing platform **60**) in response to a security event (e.g., computing platform **60**). As discussed above, examples of such security events may include but are not limited to: DDoS events, DoS events, phishing events, spamming events, malware events, web attacks, and exploitation events.

[0280] As also discussed above, threat mitigation process **10** may enable **1606** a third-party (e.g., the user/owner/operator of computing platform **60**) to select an object within the threat mitigation user interface (e.g., threat mitigation user interface **1650**), thus defining a selected object. For example, assume that the third-party (e.g., the user/owner/operator of computing platform **60**) selects destination port object **1656** within threat mitigation user interface (e.g., threat mitigation user interface **1650**), thus defining a selected object (i.e., destination port object **1656**).

[0281] In response to a third-party (e.g., the user/owner/operator of computing platform **60**) selecting an object (i.e., destination port object **1656**) within the threat mitigation user interface (e.g., threat mitigation user interface **1650**), threat mitigation process **10** may render **1608** an inspection window (e.g., inspection window **1668**) that defines object information concerning the selected object (i.e., destination port object **1656**). Examples of this object information may include but are not limited to: total outbound traffic **1670**; time-based outbound traffic graph **1672**; source IP address **1674**; destination IP address **1676**; and destination IP address **1678**.

[0282] Threat mitigation process **10** may monitor **1628** actions taken by a third-party (e.g., the user/owner/operator of computing platform **60**) when investigating the security event.

[0283] For example and when monitoring **1628** actions taken by a third-party (e.g., the user/owner/operator of computing platform **60**) when investigating the security event, threat mitigation process **10** may monitor **1630** artifacts (e.g., artifacts **1688**) gathered by the third-party (e.g., the user/owner/operator of computing platform **60**) when investigating the security event. As discussed above, examples of the artifacts (e.g., artifacts **1688**) may include but are not limited to: raw data (e.g., port usage and statistics); screen shots (e.g., all or a portion of threat mitigation user interface **1650** or inspection window **1668**); graphics (e.g., port graphical object **1664**); notes (e.g., notes authored by the third-party); annotations (e.g., annotations made by the third-party on other artifacts); audio recordings (e.g., voice recordings made by the third-party); and video recordings (e.g., video recordings made by the third-party).

[0284] When monitoring **1628** actions taken by a third-party (e.g., the user/owner/operator of computing platform **60**) when investigating the security event, threat mitigation process **10** may monitor **1632** objects (e.g., objects **1652**, **1654**, **1656**, **1658**, **1660**, **1662**, **1664**, **1666**) reviewed by the third-party (e.g., the user/owner/operator of computing platform **60**) when investigating the security event.

[0285] Threat mitigation process **10** may provide **1634** suggestions (e.g., suggestions **1694**) to the third-party (e.g., the user/owner/operator of computing platform **60**) concerning additional actions to be taken by the third-party (e.g., the user/owner/operator of computing platform **60**) concerning the investigation of the security event.

[0286] For example, threat mitigation process **10** may be capable of executing a plurality of remedial plans (e.g., remedial plans **1696**), wherein each of these remedial plans (e.g., remedial plans **1696**) may have varying requirements and is designed to thwart a different kind of attack. For example, one remedial plan may be designed to thwart a denial of services attack and may require (among other things) high CPU usage, high inbound port traffic, and high login attempts; while another remedial plan may be designed to thwart a malware attack and may require the recent opening of a uPnP port within a router and repeated server login attempts through that newly-opened uPnP port.

[0287] So when providing **1634** suggestions (e.g., suggestions **1694**) to the third-party (e.g., the user/owner/operator of computing platform **60**) concerning additional actions to be taken by the third-party (e.g., the user/owner/operator of computing platform **60**) concerning the investigation of the security event, threat mitigation process **10** may provide **1636** suggestions (e.g., suggestions **1694**) to the third-party (e.g., the user/owner/operator of computing platform **60**) concerning additional objects (e.g., objects **1652**, **1654**, **1656**, **1658**, **1660**, **1662**, **1664**, **1666**) to be reviewed by the third-party (e.g., the user/owner/operator of computing platform **60**) when investigating the security event. Therefore and if computer platform **60** is experiencing repeated server login attempts, threat mitigation process **10** may provide **1636** suggestions (e.g., suggestions **1694**) to the third-party (e.g., the user/owner/operator of computing platform **60**) concerning the review of an additional object (e.g., a router object) by the third-party (e.g., the user/owner/operator of computing platform **60**) when investigating the security event.

[0288] Additionally and when providing **1634** suggestions to the third-party (e.g., the user/owner/operator of computing platform **60**) concerning additional actions to be taken by the third-party (e.g., the user/owner/operator of computing platform **60**) concerning the investigation of the security event, threat mitigation process **10** may provide **1638** suggestions (e.g., suggestions **1694**) to the third-party (e.g., the user/owner/operator of computing platform **60**) concerning additional artifacts (e.g., artifacts **1688**) to be gathered by the third-party (e.g., the user/owner/operator of computing platform **60**) when investigating the security event. Therefore and if computer platform **60** is experiencing repeated server login attempts, threat mitigation process **10** may provide **1638** suggestions (e.g., suggestions **1694**) to the third-party (e.g., the user/owner/operator of computing platform **60**)

concerning gathering an additional artifact (e.g., a uPnP port table for the above-described router) by the third-party (e.g., the user/owner/operator of computing platform **60**) when investigating the security event.

[0289] Further and when providing **1634** suggestions to the third-party (e.g., the user/owner/operator of computing platform **60**) concerning additional actions to be taken by the third-party (e.g., the user/owner/operator of computing platform **60**) concerning the investigation of the security event, threat mitigation process **10** may provide **1640** suggestions to the third-party (e.g., the user/owner/operator of computing platform **60**) concerning a remedial action (e.g., the execution of one or more of remedial plans **1696**) to be taken by the third-party (e.g., the user/owner/operator of computing platform **60**) when investigating the security event.

[0290] Therefore, threat mitigation process **10** may provide **1640** suggestions to the third-party (e.g., the user/owner/operator of computing platform **60**) advising that a specific remedial action plan (designed to thwart a malware attack) be executed by the third-party (e.g., the user/owner/operator of computing platform **60**) if the uPnP table of the above-described router shows the recently-opened uPnP port within the above-described router as the location of the repeated server login attempts.

GENERAL

[0291] As will be appreciated by one skilled in the art, the present disclosure may be embodied as a method, a system, or a computer program product. Accordingly, the present disclosure may take the form of an entirely hardware embodiment, an entirely software embodiment (including firmware, resident software, micro-code, etc.) or an embodiment combining software and hardware aspects that may all generally be referred to herein as a "circuit," "module" or "system." Furthermore, the present disclosure may take the form of a computer program product on a computer-usable storage medium having computer-usable program code embodied in the medium.

[0292] Any suitable computer usable or computer readable medium may be utilized. The computer-usable or computer-readable medium may be, for example but not limited to, an electronic, magnetic, optical, electromagnetic, infrared, or semiconductor system, apparatus, device, or propagation medium. More specific examples (a non-exhaustive list) of the computer-readable medium may include the following: an electrical connection having one or more wires, a portable computer diskette, a hard disk, a random access memory (RAM), a read-only memory (ROM), an erasable programmable read-only memory (EPROM or Flash memory), an optical fiber, a portable compact disc read-only memory (CD-ROM), an optical storage device, a transmission media such as those supporting the Internet or an intranet, or a magnetic storage device. The computer-usable or computer-readable medium may also be paper or another suitable medium upon which the program is printed, as the program can be electronically captured, via, for instance, optical scanning of the paper or other medium, then compiled, interpreted, or otherwise processed in a suitable manner, if necessary, and then stored in a computer memory. In the context of this document, a computer-usable or computer-readable medium may be any medium that can contain, store, communicate, propagate, or transport the program for use by or in connection with the instruction execution system, apparatus, or device. The computer-usable medium may include a propagated data signal with the computer-usable program code embodied therewith, either in baseband or as part of a carrier wave. The computer usable program code may be transmitted using any appropriate medium, including but not limited to the Internet, wireline, optical fiber cable, RF, etc.

[0293] Computer program code for carrying out operations of the present disclosure may be written in an object oriented programming language such as Java, Smalltalk, C++ or the like. However, the computer program code for carrying out operations of the present disclosure may also be written in conventional procedural programming languages, such as the "C" programming language or similar programming languages. The program code may execute entirely on the user's computer, partly on the user's computer, as a stand-alone software package, partly on the user's computer and partly on a remote computer or entirely on the remote computer or server. In the latter scenario, the remote computer may be connected to the user's computer through a local area network/a wide area network/the Internet (e.g., network **14**).

[0294] The present disclosure is described with reference to flowchart illustrations and/or block diagrams of methods, apparatus (systems) and computer program products according to embodiments of the disclosure. It will be understood that each block of the flowchart illustrations and/or block diagrams, and combinations of blocks in the flowchart illustrations and/or block diagrams, may be implemented by computer program instructions. These computer program instructions may be provided to a processor of a general purpose computer/special purpose computer/other programmable data processing apparatus, such that the instructions, which execute via the processor of the computer or other programmable data processing apparatus, create means for implementing the functions/acts specified in the flowchart and/or block diagram block or blocks.

[0295] These computer program instructions may also be stored in a computer-readable memory that may direct a computer or other programmable data processing apparatus to function in a particular manner, such that the instructions stored in the computer-readable memory produce an article of manufacture including instruction means which implement the function/act specified in the flowchart and/or block diagram block or blocks.

[0296] The computer program instructions may also be loaded onto a computer or other programmable data processing apparatus to cause a series of operational steps to be performed on the computer or other programmable apparatus to produce a computer-implemented process such that the instructions which execute on the computer or other programmable apparatus provide steps for implementing the functions/acts specified in the flowchart and/or block diagram block or blocks.

[0297] The flowcharts and block diagrams in the figures may illustrate the architecture, functionality, and operation of possible implementations of systems, methods and computer program products according to various embodiments of the present disclosure. In this regard, each block in the flowchart or block diagrams may represent a module, segment, or portion of code, which comprises one or more executable instructions for implementing the specified logical function(s). It should also be noted that, in some alternative implementations, the functions noted in the block may occur out of the order noted in the figures. For example,

two blocks shown in succession may, in fact, be executed substantially concurrently, or the blocks may sometimes be executed in the reverse order, depending upon the functionality involved. It will also be noted that each block of the block diagrams and/or flowchart illustrations, and combinations of blocks in the block diagrams and/or flowchart illustrations, may be implemented by special purpose hardware-based systems that perform the specified functions or acts, or combinations of special purpose hardware and computer instructions.

[0298] The terminology used herein is for the purpose of describing particular embodiments only and is not intended to be limiting of the disclosure. As used herein, the singular forms "a", "an" and "the" are intended to include the plural forms as well, unless the context clearly indicates otherwise. It will be further understood that the terms "comprises" and/or "comprising," when used in this specification, specify the presence of stated features, integers, steps, operations, elements, and/or components, but do not preclude the presence or addition of one or more other features, integers, steps, operations, elements, components, and/or groups thereof.

[0299] The corresponding structures, materials, acts, and equivalents of all means or step plus function elements in the claims below are intended to include any structure, material, or act for performing the function in combination with other claimed elements as specifically claimed. The description of the present disclosure has been presented for purposes of illustration and description, but is not intended to be exhaustive or limited to the disclosure in the form disclosed. Many modifications and variations will be apparent to those of ordinary skill in the art without departing from the scope and spirit of the disclosure. The embodiment was chosen and described in order to best explain the principles of the disclosure and the practical application, and to enable others of ordinary skill in the art to understand the disclosure for various embodiments with various modifications as are suited to the particular use contemplated.

[0300] A number of implementations have been described. Having thus described the disclosure of the present application in detail and by reference to embodiments thereof, it will be apparent that modifications and variations are possible without departing from the scope of the disclosure defined in the appended claims.

What is claimed is:

1. A computer-implemented method, executed on a computing device, comprising:
- rendering a threat mitigation user interface that identifies objects within a computing platform in response to a security event;
- monitoring actions taken by a third-party when investigating the security event; and
- providing suggestions to the third-party concerning additional actions to be taken by the third-party concerning the investigation of the security event.

2. The computer-implemented method of claim **1** wherein monitoring actions taken by a third-party when investigating the security event include:
- monitoring artifacts gathered by the third-party when investigating the security event.

3. The computer-implemented method of claim **2** wherein the artifacts include one or more of:
- raw data;
- screen shots;
- graphics;
- notes;
- annotations;
- audio recordings; and
- video recordings.

4. The computer-implemented method of claim **1** wherein monitoring actions taken by a third-party when investigating the security event include:
- monitoring objects reviewed by the third-party when investigating the security event.

5. The computer-implemented method of claim **1** wherein providing suggestions to the third-party concerning additional actions to be taken by the third-party concerning the investigation of the security event includes:
- providing suggestions to the third-party concerning additional objects to be reviewed by the third-party when investigating the security event.

6. The computer-implemented method of claim **1** wherein providing suggestions to the third-party concerning additional actions to be taken by the third-party concerning the investigation of the security event includes:
- providing suggestions to the third-party concerning additional artifacts to be gathered by the third-party when investigating the security event.

7. The computer-implemented method of claim **1** wherein providing suggestions to the third-party concerning additional actions to be taken by the third-party concerning the investigation of the security event includes:
- providing suggestions to the third-party concerning a remedial action to be taken by the third-party when investigating the security event.

8. The computer-implemented method of claim **1** further comprising:
- enabling the third-party to select an object within the threat mitigation user interface, thus defining a selected object; and
- rendering an inspection window that defines object information concerning the selected object.

9. The computer-implemented method of claim **8** wherein the inspection window is a popup inspection window.

10. The computer-implemented method of claim **8** wherein the inspection window is a slide out inspection window.

11. A computer program product residing on a computer readable medium having a plurality of instructions stored thereon which, when executed by a processor, cause the processor to perform operations comprising:
- rendering a threat mitigation user interface that identifies objects within a computing platform in response to a security event;
- monitoring actions taken by a third-party when investigating the security event; and
- providing suggestions to the third-party concerning additional actions to be taken by the third-party concerning the investigation of the security event.

12. The computer-implemented method of claim **11** wherein monitoring actions taken by a third-party when investigating the security event include:
- monitoring artifacts gathered by the third-party when investigating the security event.

13. The computer-implemented method of claim **12** wherein the artifacts include one or more of:
- raw data;
- screen shots;

graphics;
notes;
annotations;
audio recordings; and
video recordings.

14. The computer-implemented method of claim **11** wherein monitoring actions taken by a third-party when investigating the security event include:
monitoring objects reviewed by the third-party when investigating the security event.

15. The computer-implemented method of claim **11** wherein providing suggestions to the third-party concerning additional actions to be taken by the third-party concerning the investigation of the security event includes:
providing suggestions to the third-party concerning additional objects to be reviewed by the third-party when investigating the security event.

16. The computer-implemented method of claim **11** wherein providing suggestions to the third-party concerning additional actions to be taken by the third-party concerning the investigation of the security event includes:
providing suggestions to the third-party concerning additional artifacts to be gathered by the third-party when investigating the security event.

17. The computer-implemented method of claim **11** wherein providing suggestions to the third-party concerning additional actions to be taken by the third-party concerning the investigation of the security event includes:
providing suggestions to the third-party concerning a remedial action to be taken by the third-party when investigating the security event.

18. The computer-implemented method of claim **11** further comprising:
enabling the third-party to select an object within the threat mitigation user interface, thus defining a selected object; and
rendering an inspection window that defines object information concerning the selected object.

19. The computer-implemented method of claim **18** wherein the inspection window is a popup inspection window.

20. The computer-implemented method of claim **18** wherein the inspection window is a slide out inspection window.

21. A computing system including a processor and memory configured to perform operations comprising:
rendering a threat mitigation user interface that identifies objects within a computing platform in response to a security event;
monitoring actions taken by a third-party when investigating the security event; and
providing suggestions to the third-party concerning additional actions to be taken by the third-party concerning the investigation of the security event.

22. The computer-implemented method of claim **21** wherein monitoring actions taken by a third-party when investigating the security event include:
monitoring artifacts gathered by the third-party when investigating the security event.

23. The computer-implemented method of claim **22** wherein the artifacts include one or more of:
raw data;
screen shots;
graphics;
notes;
annotations;
audio recordings; and
video recordings.

24. The computer-implemented method of claim **21** wherein monitoring actions taken by a third-party when investigating the security event include:
monitoring objects reviewed by the third-party when investigating the security event.

25. The computer-implemented method of claim **21** wherein providing suggestions to the third-party concerning additional actions to be taken by the third-party concerning the investigation of the security event includes:
providing suggestions to the third-party concerning additional objects to be reviewed by the third-party when investigating the security event.

26. The computer-implemented method of claim **21** wherein providing suggestions to the third-party concerning additional actions to be taken by the third-party concerning the investigation of the security event includes:
providing suggestions to the third-party concerning additional artifacts to be gathered by the third-party when investigating the security event.

27. The computer-implemented method of claim **21** wherein providing suggestions to the third-party concerning additional actions to be taken by the third-party concerning the investigation of the security event includes:
providing suggestions to the third-party concerning a remedial action to be taken by the third-party when investigating the security event.

28. The computer-implemented method of claim **21** further comprising:
enabling the third-party to select an object within the threat mitigation user interface, thus defining a selected object; and
rendering an inspection window that defines object information concerning the selected object.

29. The computer-implemented method of claim **28** wherein the inspection window is a popup inspection window.

30. The computer-implemented method of claim **28** wherein the inspection window is a slide out inspection window.

* * * * *