US 20160065599A1

(19) United States

(12) Patent Application Publication

Hovor et al.

(10) Pub. No.: US 2016/0065599 A1

(43) Pub. Date: Mar. 3, 2016

(54) **UNSTRUCTURED SECURITY THREAT INFORMATION ANALYSIS**

(71) Applicant: **Accenture Global Services Limited**, Dublin (IE)

(72) Inventors: **Elvis Hovor**, Clarksburg, MD (US); **Shimon Modi**, Washington, DC (US); **Shubhashis Sengupta**, Karntaka (IN); **Roshni Ramesh Ramnani**, Karnataka (IN); **Annervaz Karukapadath Mohamedrasheed**, Kerala (IN)

(21) Appl. No.: **14/473,743**

(22) Filed: **Aug. 29, 2014**

Publication Classification

(51) **Int. Cl.**

H04L 29/06 (2006.01)

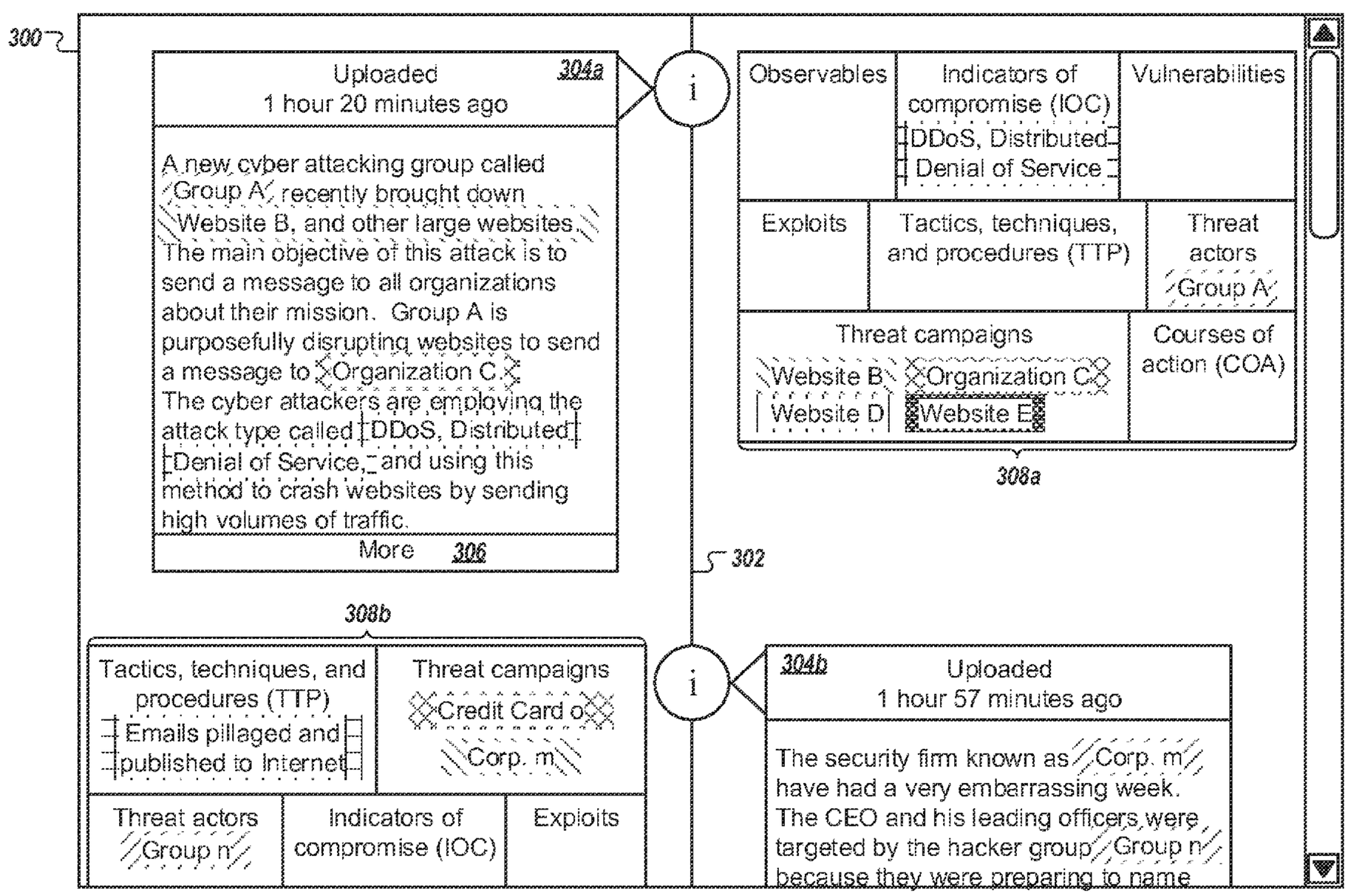
H04L 29/08 (2006.01)

(52) **U.S. Cl.**

CPC ***H04L 63/1416*** (2013.01); ***H04L 67/10*** (2013.01)

(57) **ABSTRACT**

Methods, systems, and apparatus, including computer programs encoded on computer storage media, for creating structured data using data received from unstructured textual data sources. One of the methods includes receiving unstructured textual data, identifying one or more keywords in the unstructured textual data, determining one or more patterns included in the unstructured textual data using the identified keywords, identifying one or more intelligence types that correspond with the unstructured textual data using the determined patterns, and associating, for each of the identified intelligence types, a data subset from the unstructured textual data with the respective intelligence type.



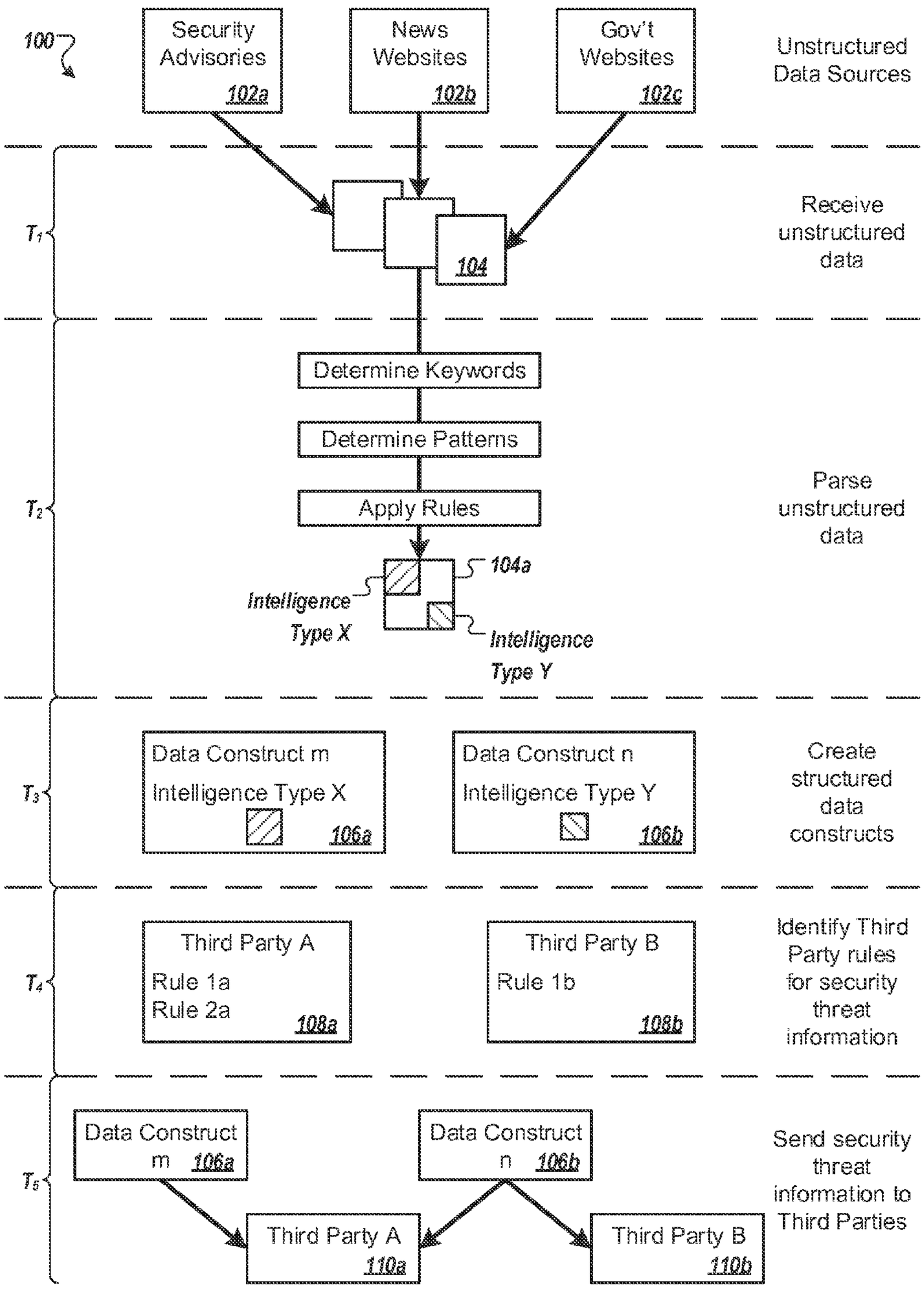

100
Security Advisories 102a
News Websites 102b
Gov't Websites 102c
Unstructured Data Sources
T1
104
Receive unstructured data
T2
Determine Keywords
Determine Patterns
Apply Rules
104a
Intelligence Type X
Intelligence Type Y
Parse unstructured data
T3
Data Construct m
Intelligence Type X
106a
Data Construct n
Intelligence Type Y
106b
Create structured data constructs
T4
Third Party A
Rule 1a
Rule 2a
108a
Third Party B
Rule 1b
108b
Identify Third Party rules for security threat information
T5
Data Construct m 106a
Data Construct n 106b
Third Party A 110a
Third Party B 110b
Send security threat information to Third Parties

FIG. 1

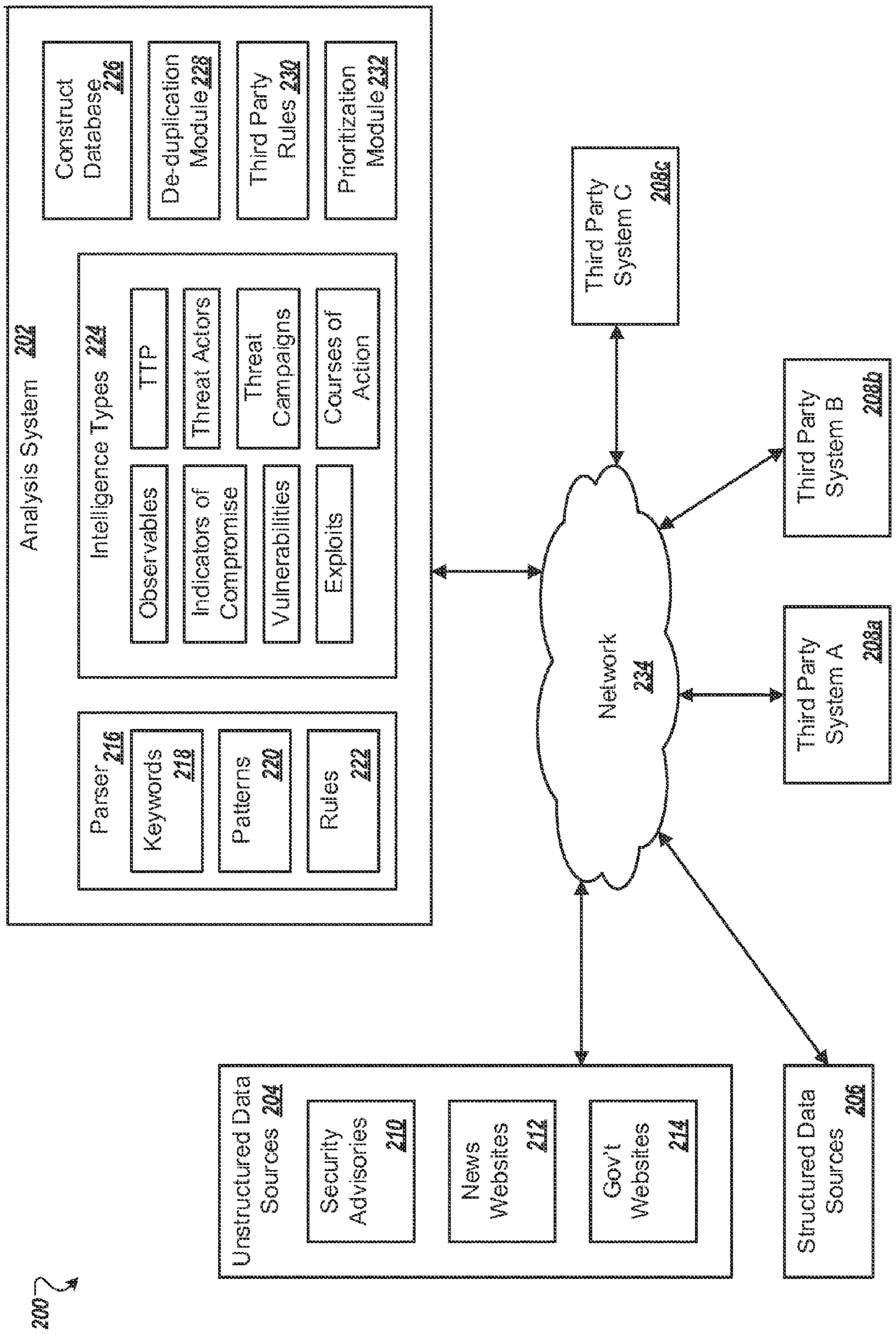

200
Analysis System 202
Construct Database 226
De-duplication Module 228
Third Party Rules 230
Prioritization Module 232
Intelligence Types 224
TTP
Threat Actors
Threat Campaigns
Courses of Action
Observables
Indicators of Compromise
Vulnerabilities
Exploits
Parser 216
Keywords 218
Patterns 220
Rules 222
Third Party System C 208c
Third Party System B 208b
Third Party System A 208a
Network 234
Unstructured Data Sources 204
Security Advisories 210
News Websites 212
Gov't Websites 214
Structured Data Sources 206

FIG. 2

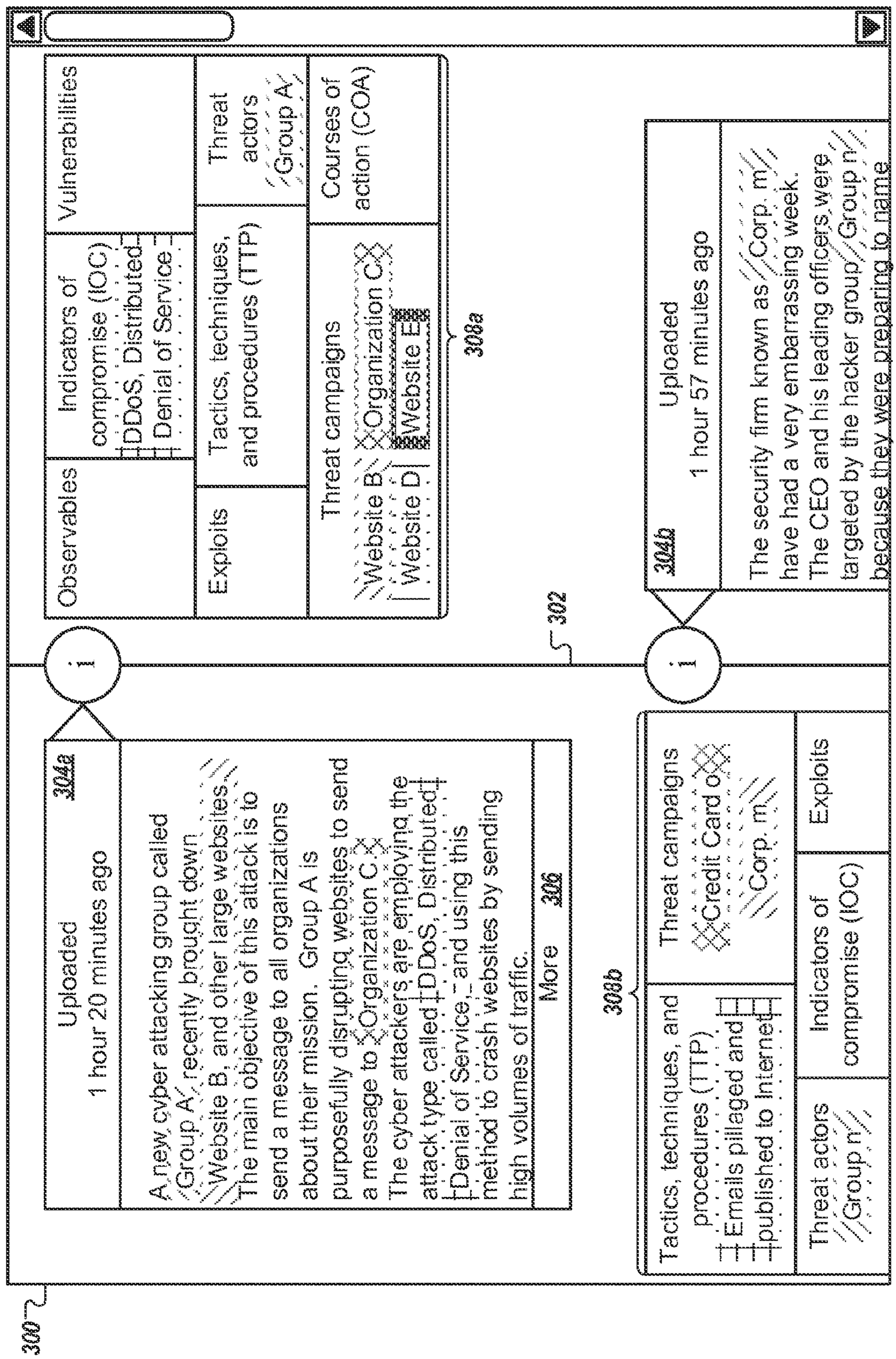

300
Uploaded
1 hour 20 minutes ago
304a
A new cyber attacking group called Group A recently brought down Website B, and other large websites. The main objective of this attack is to send a message to all organizations about their mission. Group A is purposefully disrupting websites to send a message to Organization C. The cyber attackers are employing the attack type called DDoS, Distributed Denial of Service, and using this method to crash websites by sending high volumes of traffic.
More
306
308b
Tactics, techniques, and procedures (TTP)
Emails pillaged and published to Internet
Threat actors
Group n
Indicators of compromise (IOC)
Credit Card o
Corp. m
Exploits
Observables
Indicators of compromise (IOC)
DDoS, Distributed Denial of Service
Vulnerabilities
Exploits
Tactics, techniques, and procedures (TTP)
Threat actors
Group A
Courses of action (COA)
Threat campaigns
Website B
Organization C
Website D
Website E
308a
302
304b
Uploaded
1 hour 57 minutes ago
The security firm known as Corp. m have had a very embarrassing week. The CEO and his leading officers were targeted by the hacker group Group n, because they were preparing to name

FIG. 3

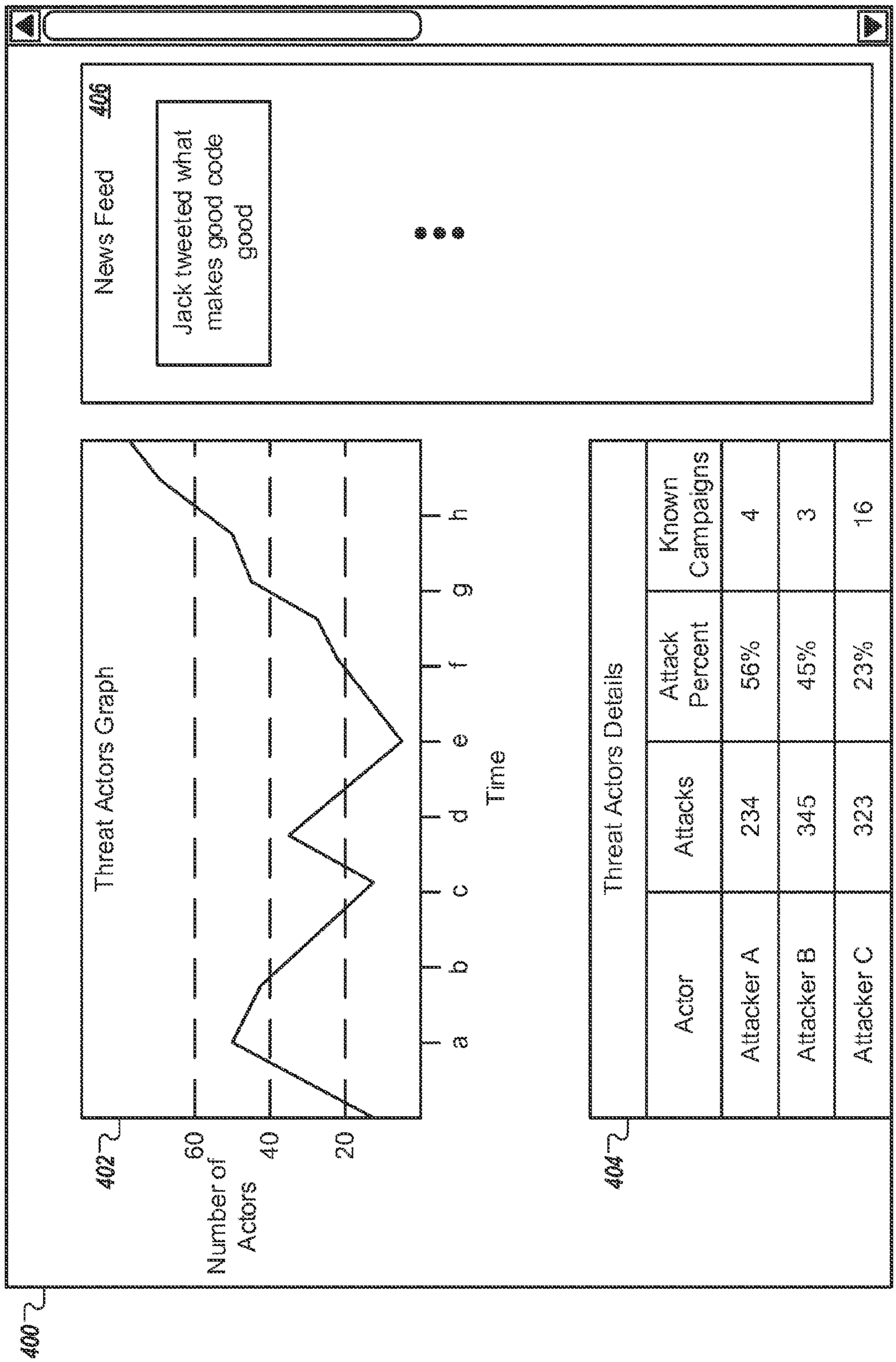

400
News Feed
406
Jack tweeted what makes good code good
...
Threat Actors Graph
402
Number of Actors
60
40
20
a
b
c
d
e
f
g
h
Time
Threat Actors Details
404
Actor
Attacks
Attack Percent
Known Campaigns
Attacker A
234
56%
4
Attacker B
345
45%
3
Attacker C
323
23%
16

FIG. 4

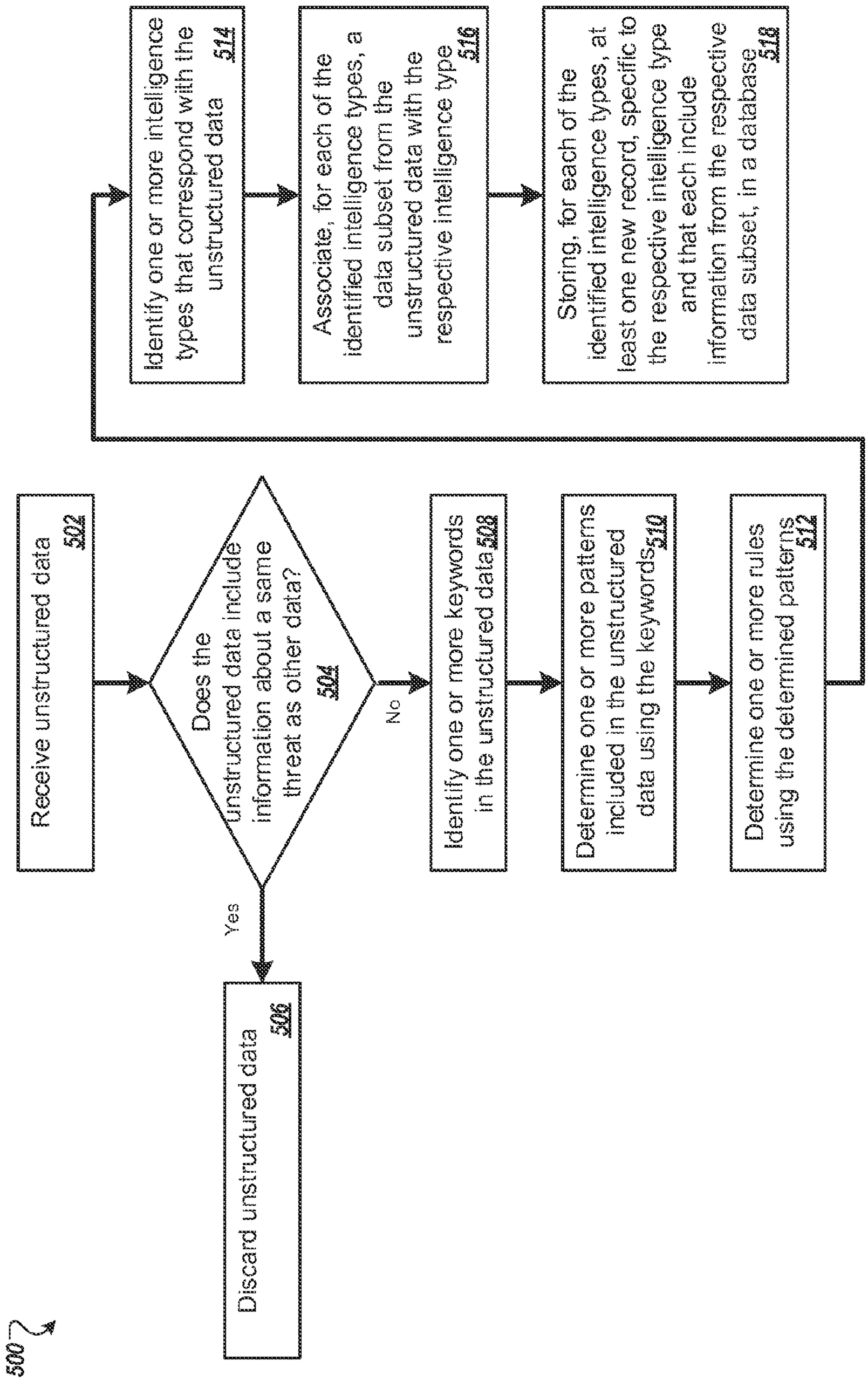

500
Receive unstructured data 502
Does the unstructured data include information about a same threat as other data? 504
Yes
No
Discard unstructured data 506
Identify one or more keywords in the unstructured data 508
Determine one or more patterns included in the unstructured data using the keywords 510
Determine one or more rules using the determined patterns 512
Identify one or more intelligence types that correspond with the unstructured data 514
Associate, for each of the identified intelligence types, a data subset from the unstructured data with the respective intelligence type 516
Storing, for each of the identified intelligence types, at least one new record, specific to the respective intelligence type and that each include information from the respective data subset, in a database 518

FIG. 5

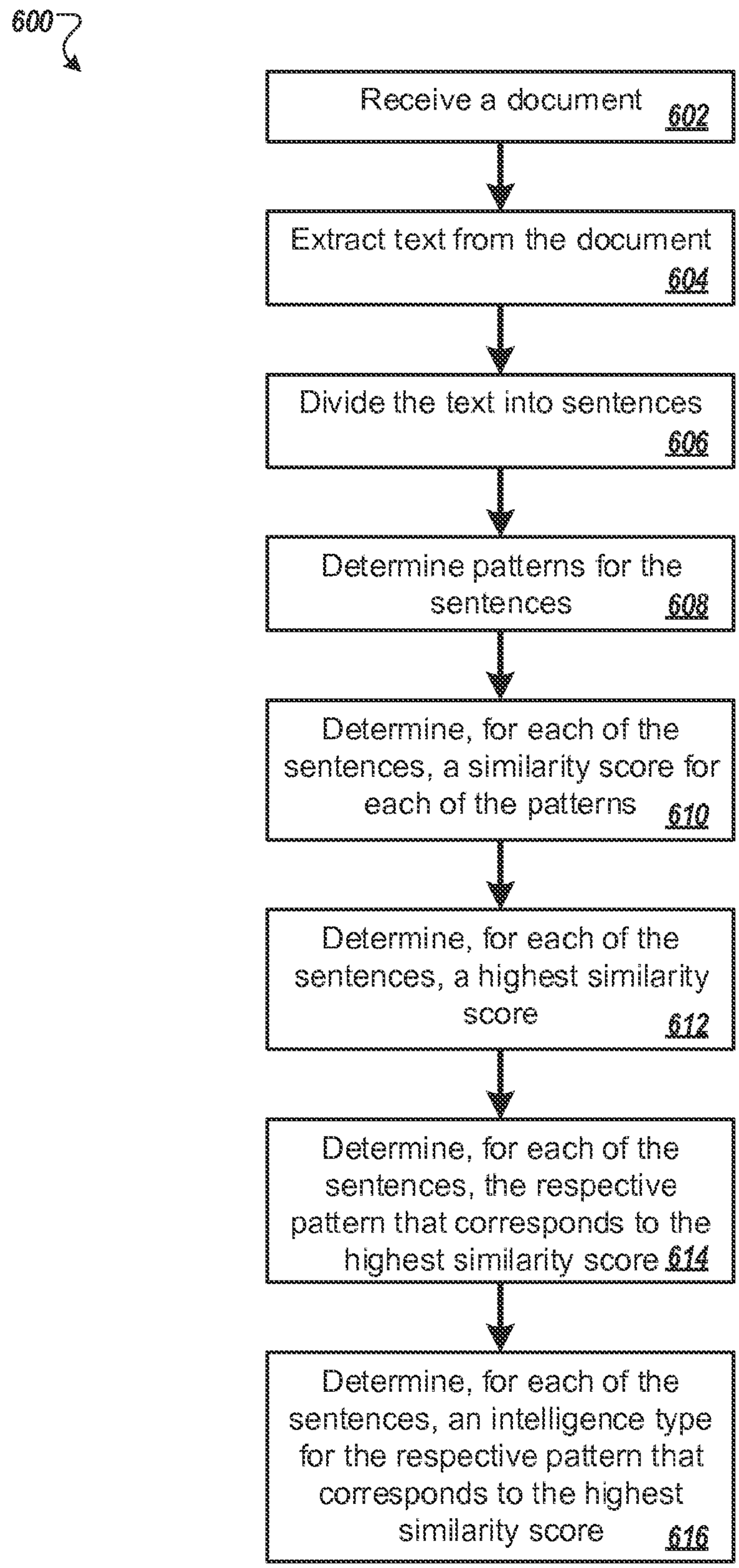

600
Receive a document 602
Extract text from the document 604
Divide the text into sentences 606
Determine patterns for the sentences 608
Determine, for each of the sentences, a similarity score for each of the patterns 610
Determine, for each of the sentences, a highest similarity score 612
Determine, for each of the sentences, the respective pattern that corresponds to the highest similarity score 614
Determine, for each of the sentences, an intelligence type for the respective pattern that corresponds to the highest similarity score 616

FIG. 6

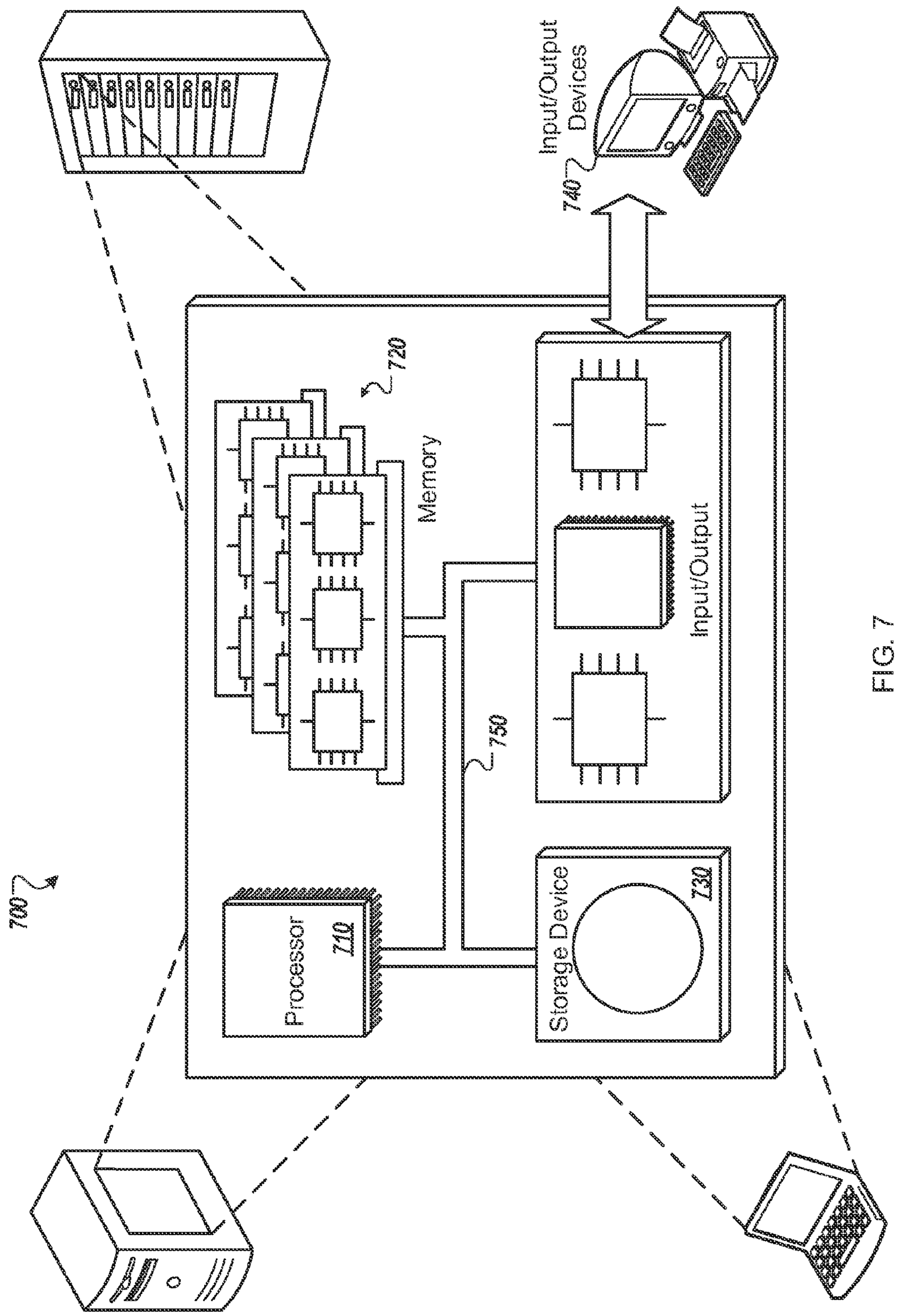

700
720
Memory
Input/Output Devices
740
Input/Output
750
Processor
710
Storage Device
730

FIG. 7

UNSTRUCTURED SECURITY THREAT INFORMATION ANALYSIS

BACKGROUND

[0001] Organizations may use security threat information to make changes to their networks, e.g., in response to perceived or actual threats. For instance, an organization may learn about a software vulnerability and install a software update that removes the vulnerability or change network settings to prevent, or reduce the likelihood, of a malicious actor exploiting the vulnerability.

SUMMARY

[0002] In general, one innovative aspect of the subject matter described in this specification can be embodied in methods that include the actions of receiving unstructured textual data, identifying one or more keywords in the unstructured textual data, determining one or more patterns included in the unstructured textual data using the identified keywords, identifying one or more intelligence types that correspond with the unstructured textual data using the determined patterns, and associating, for each of the identified intelligence types, a data subset from the unstructured textual data with the respective intelligence type. Other embodiments of this aspect include corresponding computer systems, apparatus, and computer programs recorded on one or more computer storage devices, each configured to perform the actions of the methods. A system of one or more computers can be configured to perform particular operations or actions by virtue of having software, firmware, hardware, or a combination of them installed on the system that in operation causes or cause the system to perform the actions. One or more computer programs can be configured to perform particular operations or actions by virtue of including instructions that, when executed by data processing apparatus, cause the apparatus to perform the actions.

[0003] In general, one innovative aspect of the subject matter described in this specification can be embodied in methods that include the actions of receiving unstructured textual data, parsing the unstructured textual data into a plurality of sections including a particular section, identifying one or more keywords in the particular section of the unstructured textual data, determining one or more patterns that match the particular section using the identified keywords, identifying one or more intelligence types that correspond with the particular section using the determined patterns, and associating, for each of the identified intelligence types, a data subset from the unstructured textual data, associated with the particular section, with the respective intelligence type. Other embodiments of this aspect include corresponding computer systems, apparatus, and computer programs recorded on one or more computer storage devices, each configured to perform the actions of the methods. A system of one or more computers can be configured to perform particular operations or actions by virtue of having software, firmware, hardware, or a combination of them installed on the system that in operation causes or cause the system to perform the actions. One or more computer programs can be configured to perform particular operations or actions by virtue of including instructions that, when executed by data processing apparatus, cause the apparatus to perform the actions.

[0004] In general, one innovative aspect of the subject matter described in this specification can be embodied in methods that include the actions of receiving unstructured textual data, identifying one or more keywords in the unstructured textual data, determining one or more patterns included in the unstructured textual data using the identified keywords, identifying one or more intelligence types that correspond with the unstructured textual data using the determined patterns, associating, for each of the identified intelligence types, a data subset from the unstructured textual data with the respective intelligence type, determining a rule for a third party that indicates that the third party should receive data associated with a particular intelligence type, determining that at least one of the identified intelligence types includes the particular intelligence type, and providing, for each of the at least one of the identified intelligence types, the respective data subset to the third party. Other embodiments of this aspect include corresponding computer systems, apparatus, and computer programs recorded on one or more computer storage devices, each configured to perform the actions of the methods. A system of one or more computers can be configured to perform particular operations or actions by virtue of having software, firmware, hardware, or a combination of them installed on the system that in operation causes or cause the system to perform the actions. One or more computer programs can be configured to perform particular operations or actions by virtue of including instructions that, when executed by data processing apparatus, cause the apparatus to perform the actions.

[0005] The foregoing and other embodiments can each optionally include one or more of the following features, alone or in combination. Associating, for each of the identified intelligence types, the data subset from the unstructured textual data with the respective intelligence type may include storing, for each of the identified intelligence types, at least one new record, in a database, specific to the respective intelligence type that each includes information from the respective data subset. The unstructured may include an advisory. Receiving the unstructured textual data may include receiving the unstructured textual data from a government or a security source. Identifying the intelligence types that correspond with the unstructured textual data using the determined patterns may include determining one or more rules using the determined patterns, and identifying the intelligence types that correspond with the unstructured textual data using the determined rules.

[0006] In some implementations, the method may include determining a rule for a third party that indicates that the third party should receive data associated with a particular intelligence type, determining that at least one of the identified intelligence types includes the particular intelligence type, and providing, for each of the at least one of the identified intelligence types, the respective data subset to the third party. Providing, for each of the at least one of the identified intelligence types, the respective data subset to the third party may include providing, for each of the at least one of the identified intelligence types, instructions to the third party for presentation of the respective data subset. The method may include receiving a plurality of second unstructured textual datasets that each include information about a potential security threat, determining that a first subset of the second unstructured textual datasets includes information about the same threat as a second, different subset of the second unstructured textual datasets, discarding the first subset, and identifying,

for each second unstructured textual dataset in the second subset, one or more keywords in the respective second unstructured textual dataset.

[0007] In some implementations, the method may include parsing the unstructured textual data into a plurality of sections including a particular section. Each of the plurality of sections may include a sentence or a paragraph. The data subset may include data representing the particular section.

[0008] In general, one innovative aspect of the subject matter described in this specification can be embodied in methods that include the actions of for each dataset from multiple datasets that each include information about potential security threats: identifying, by one or more computers, one or more respective intelligence types that each categorizes a subset of data in the respective dataset, and associating, for each of the identified intelligence types, each of the subsets of data, which are categorized by the respective intelligence type, with the respective intelligence type. The method includes the actions of determining, for each of the subsets, whether the respective subset does not include information about the same threat as a different subset, determining one or more rules for a third party that each indicate that the third party should receive data associated with particular types of potential security threats and priority information for the data, determining, for each of the potential security threats indicated in the rules, a group of the subsets that include information associated with the respective potential security threat, assigning, for each subset in each of the groups, a priority to the respective subset using the priority information, and providing the determined subsets to the third party using the respective priorities. Other embodiments of this aspect include corresponding computer systems, apparatus, and computer programs recorded on one or more computer storage devices, each configured to perform the actions of the methods. A system of one or more computers can be configured to perform particular operations or actions by virtue of having software, firmware, hardware, or a combination of them installed on the system that in operation causes or cause the system to perform the actions. One or more computer programs can be configured to perform particular operations or actions by virtue of including instructions that, when executed by data processing apparatus, cause the apparatus to perform the actions.

[0009] In general, one innovative aspect of the subject matter described in this specification can be embodied in methods that include the actions of for each dataset from multiple datasets that each include information about potential security threats: identifying, by at least one of the computers, one or more respective intelligence types that each categorizes a subset of data in the respective dataset, and associating, for each of the identified intelligence types, each of the subsets of data, which are categorized by the respective intelligence type, with the respective intelligence type. The method includes the actions of determining one or more rules for a third party that each indicate that the third party should receive data associated with particular types of potential security threats and priority information for the data, determining, for each of the potential security threats indicated in the rules, a group of the subsets that include information associated with the respective potential security threat, assigning, for each subset in each of the groups, a priority to the respective subset using the priority information, and providing the determined subsets to the third party using the respective priorities. Other embodiments of this aspect include corresponding computer systems, apparatus, and computer programs recorded on one or more computer storage devices, each configured to perform the actions of the methods. A system of one or more computers can be configured to perform particular operations or actions by virtue of having software, firmware, hardware, or a combination of them installed on the system that in operation causes or cause the system to perform the actions. One or more computer programs can be configured to perform particular operations or actions by virtue of including instructions that, when executed by data processing apparatus, cause the apparatus to perform the actions.

[0010] In general, one innovative aspect of the subject matter described in this specification can be embodied in methods that include the actions of for each dataset from multiple datasets that each include information about potential security threats: identifying one or more respective intelligence types that each categorizes a subset of data in the respective dataset, and associating, for each of the identified intelligence types, each of the subsets of data, which are categorized by the respective intelligence type, with the respective intelligence type. The method includes the actions of determining, for each of the subsets, whether the respective subset does not include information about the same threat as a different subset, determining, for a third party, a group of the subsets that include information requested by the third party, assigning, for each subset in the group, a priority to the respective subset using priority information for the third party, and providing the determined subsets to the third party using the respective priorities. Other embodiments of this aspect include corresponding computer systems, apparatus, and computer programs recorded on one or more computer storage devices, each configured to perform the actions of the methods. A system of one or more computers can be configured to perform particular operations or actions by virtue of having software, firmware, hardware, or a combination of them installed on the system that in operation causes or cause the system to perform the actions. One or more computer programs can be configured to perform particular operations or actions by virtue of including instructions that, when executed by data processing apparatus, cause the apparatus to perform the actions.

[0011] The foregoing and other embodiments can each optionally include one or more of the following features, alone or in combination. Providing the determined subsets to the third party using the respective priorities may include providing the determined subsets for presentation according to the respective priorities. The method may include receiving the datasets from one or more sources, and parsing each of the datasets into the subsets of data. Identifying the respective intelligence types that each categorize a subset of data in the respective dataset may include identifying the respective intelligence types that each categorize one of the parsed subsets. The method may include determining that a first subset from the subsets does not include information about the same threat as a different subset. Determining the group of the subsets that include information associated with the respective potential security threat may include determining a group that includes the first subset.

[0012] In some implementations, the method may include determining that a first subset from the subsets includes information about the same threat as a different subset. Determining the group of the subsets that include information associated with the respective potential security threat may include determining a group that does not include the first subset. The

method may include determining that the first subset includes information with an older timestamp than the corresponding different subset. Determining the group that does not include the first subset may include determining the group that does not include the first subset in response to determining that the first subset includes information with the older timestamp than the corresponding different subset. The method may include determining that the first subset includes information from a less reputable source than the corresponding different subset. Determining the group that does not include the first subset may include determining the group that does not include the first subset in response to determining that the first subset includes information from a less reputable source than the corresponding different subset. The method may include determining that content in the first subset varies from the corresponding subset by more than a threshold amount.

[0013] In some implementations, the method may include determining that a first subset from the subsets includes information about the same threat as a different subset, and merging the first subset with the corresponding different subset. The method may include determining that the first subset varies from the corresponding different subset by less than a threshold amount. Merging the first subset with the corresponding different subset may include merging the first subset with the corresponding different subset in response to determining that the first subset varies from the corresponding different subset by less than the threshold amount. The method may include determining that a first subset from the subsets includes information about the same threat as a different subset, determining that the first subset varies from the corresponding different subset by more than a threshold amount, and linking the first subset with the corresponding different subset.

[0014] In some implementations, the priority information may include priority information for the potential security threats. Determining, for each of the potential security threats indicated in the rules, the group of the subsets that include information associated with the respective potential security threats may include determining, for one of the potential security threats that identifies a particular software application used by the third party, that each of the subsets in the groups of subsets include information about the particular software application. Determining, for each of the potential security threats indicated in the rules, the group of the subsets that include information associated with the respective potential security threats may include determining, for one of the potential security threats that identifies a particular hardware device used by the third party, that each of the subsets in the groups of subsets include information about the particular hardware device.

[0015] In some implementations, the method may include determining, for each of the subsets, whether the respective subset does not include information about the same threat as a different subset. Determining, for the third party, the group of the subsets that include information requested by the third party may include determining one or more rules for the third party that each indicate that the third party should receive data associated with particular types of potential security threats and the priority information for the data. Assigning, for each subset in the group, a priority to the respective subset using priority information for the third party may include assigning, for each subset in each of the groups, a priority to the respective subset using the priority information.

[0016] The subject matter described in this specification can be implemented in particular embodiments so as to realize one or more of the following advantages. In some implementations, when a system receives data from multiple sources and stores the data in data constructs, e.g., formatted according to a standardized representation, the system reduces duplicate data sent to third parties and reduces potential confusion about related threat intelligence. In some implementations, when a system formats data received from multiple sources in data constructs, the system may improve expressiveness when describing a certain threat actor, their activities, how they operate, and possible mitigation actions. In some implementations, a system automates the process of determining duplicates and applying appropriate resolution to address duplicate data. In some implementations, a system provides users with an automatic determination of relevance of data specific to their particular infrastructure, e.g., hardware, software, or both.

[0017] In some implementations, transforming and formatting data from unstructured textual data sources into a structured collection of data may increase the quantity of documents that may be analyzed by a system, the type of documents that can be analyzed, provide more accurate correlation of threat intelligence information from multiples sources, or any combination of two or more of these. In some implementations, a system might not lose data or data granularity when creating data constructs from unstructured or structured data. In some implementations, a system may provide clients with the higher priority data before lower priority data, e.g., as determined for each client.

[0018] The details of one or more embodiments of the subject matter of this specification are set forth in the accompanying drawings and the description below. Other features, aspects, and advantages of the subject matter will become apparent from the description, the drawings, and the claims.

BRIEF DESCRIPTION OF THE DRAWINGS

[0019] FIG. **1** is an example of an environment in which a system receives and parses unstructured data to create structured data constructs.

[0020] FIG. **2** is an example of an environment in which an analysis system receives data from unstructured data sources, parses the data, creates structured data constructs from the parsed data, and provides the structured data to third party systems.

[0021] FIG. **3** is an example of a user interface for presenting security threat information.

[0022] FIG. **4** is an example of a user interface that includes threat trend information.

[0023] FIG. **5** is a flow diagram of a process for storing a record, created from unstructured data, in a database.

[0024] FIG. **6** is a flow diagram of a process for determining intelligence types for unstructured data in a document.

[0025] FIG. **7** is a block diagram of a computing system that can be used in connection with computer-implemented methods described in this document.

[0026] Like reference numbers and designations in the various drawings indicate like elements.

DETAILED DESCRIPTION

[0027] Unstructured threat intelligence gathering may automate the process of analyzing threat information from unstructured data sources, such as security advisories, secu-

rity alerts, security warnings, or blog sources, using natural language processing techniques. A system may parse unstructured data from security advisories provided by agencies like the Federal Bureau of Investigation, the Center for Internet Security or the Multi-State Information Sharing Analysis Center, or the United States Computer Emergency Readiness Team. The system may use natural language processing, such as the Apache OpenNLP library, and a machine learning library, such as Waikato Environment for Knowledge Analysis (Weka) or in the R programming language, to identify relevant threat information from these advisories and map the threat information to structured data constructs, e.g., that may be accepted by multiple different organizations.

[0028] FIG. **1** is an example of an environment **100** in which a system receives and parses unstructured data **104** to create structured data constructs. The system may receive the unstructured data, e.g., unstructured textual data, **104** at time T**1** from multiple different unstructured data sources **102***a-c*, such as data included in a security advisory **102***a*, data from a news website **102***b*, or data from a government website **102***c*. The system may receive the unstructured data **104** from a news feed, in an alert, e.g., an email, or through any other appropriate method.

[0029] At time T**2**, the system parses some of the unstructured data **104***a*. For instance, the system determines keywords, phrases, sentences, or any combination of two or more of these, in the unstructured data **104***a*. The system may determine patterns in the unstructured data **104***a* using the keywords, e.g., where each pattern corresponds with particular keywords or types of keywords and potentially locations of the keywords with respect to other keywords.

[0030] The system may use rules that correspond with the determined patterns to identify intelligence types that categorize all or parts of the unstructured data **104***a*. For example, the system may determine that co-occurrences of a set of keywords or a particular combination of keywords located within a sentence, adjacent sentences, or a paragraph, indicate that the text corresponds with a particular intelligence type, such as intelligence type X or intelligence type Y, each of which are associated with subsets of the unstructured data **104***a*.

[0031] Some examples of intelligence types may include i) observables, ii) indicators of compromise (IOC), iii) vulnerabilities, iv) exploits, v) adversary tactics, techniques, and procedures (TTP), vi) threat actors, vii) threat campaigns, and viii) courses of action (COA). Each of the intelligence types may have a corresponding data construct in the system, e.g., particular to the types of information available for or related to the respective intelligence type. For instance, the intelligence types may be formatted according to Structured Threat Information eXpression (STIX™).

[0032] Observables may be properties or measurable events pertinent to the operation of computers and networks. Some examples of observables may include information about a file, e.g., name, hash, or size; a registry key value; a service being started; or a Hypertext Transfer Protocol (HTTP) request being sent.

[0033] Indicators of compromise (IOC) may include certain observable conditions as well as contextual information about patterns of those observable conditions and how and when a pattern should be acted on. The contextual information may represent artifacts or behaviors of interest within a cyber-security context or both. The patterns of the observable conditions may be mapped to related TTP context information, include relevant metadata about confidence in the indicator's assertion, handling restrictions, valid time windows, likely impact, sightings of the information indicator, structured test mechanisms for detection, related campaigns, or suggested COA, or both related TTP context information and relevant metadata.

[0034] Vulnerabilities may include information about security vulnerabilities identified by independent research teams, internal teams, or security vendors. The security vulnerabilities may indicate particular vulnerabilities for an operating system, an application, or a version of an application, e.g., particular to a specific operating system.

[0035] Exploits may include information related to system vulnerabilities or the tools used to take advantage of a particular vulnerability by techniques of a threat actor. One example of exploit information may include identification or characterization of a vulnerability.

[0036] TTP may include information about how threat actors operate and perform their adversarial actions. An example tactic may include the use of malware to steal credit card credentials. An example technique may include sending emails to potential victims that include malicious attachments, e.g., used to capture credit card information. An example procedure may include research to identify potential targets for malicious email.

[0037] Threat actors may include characterizations of malicious actors that represent a cyber-threat and may include potential identity, location, specific techniques used by, presumed intent of and other observed historical behavior for the respective threat actor. Threat actors may be linked to TTPs, threat campaigns, or both, which were used by the respective threat actor or other threat actors that may be or are known to have been associated with the respective threat actor.

[0038] Threat campaigns may relate threat actors to particular IOC, exploits, TTP, or any combination of two or more of these. For instance, a threat campaign may indicate the particular procedures, and the indicators used to determine those procedures, used by a particular threat actor to compromise one or more organizations.

[0039] COA may include recommended actions to mitigate or remediate risks presented by IOC or motivation of threat actors or threat campaigns. For instance, a COA may be corrective, e.g., to fix an exploited vulnerability, or preventative, e.g., to fix a potential vulnerability or a vulnerability that has not yet been exploited, for the particular organization or another organization.

[0040] In some examples, some of the unstructured data **104***a*, e.g., a particular subset of the unstructured data **104***a*, may be associated with multiple intelligence types. For instance, a feed received from a first source may include a particular subset with information that can be categorized as a threat actor and as a threat campaign.

[0041] The system uses the identified intelligence types to create structured data constructs for the unstructured data **104***a* at time T**3**. For example, the system determines data constructs specific to the respective intelligence types and stores data from the subsets of the unstructured data **104***a* in the respective data constructs. The system may store the subset that corresponds with the intelligence type X in a data construct m **106***a*, e.g., specific to the intelligence type X, and the subset that corresponds with the intelligence type Y in a data construct n **106***b*, e.g., specific to the intelligence type Y.

[0042] In some examples, the system determines whether some data for different intelligence types is related to the

same threat, e.g., the same threat actor. The system may determine a relationship for the data that is related to the same threat and how that relationship should be identified, as described in more detail below. The system may determine relationships during time T**1**, time T**2**, time T**3**, or a combination of two or more of these, depending on a configuration of the system and how the relationships are determined.

[0043] Once the data constructs are created, or for data that was previously placed in data constructs and not yet sent to a specific third party, at time T**4**, the system identifies third party rules **108***a-b* for security threat information. The rules are particular to a specific third party. For example, Third Party A **110***a* may have rules **1***a* and **2***a* and Third Party B **110***b* may have rule **1***b*.

[0044] The rules for the different third parties may be the same or different. For instance, rule **1***b* may be the same as rule **2***a*, e.g., may identify the same types of intelligence types or the same subsets of intelligence types. These two rules may indicate that both third parties want to receive information about intelligence type Y, and the rule **1***a* may indicate that Third Party A would like to receive information about intelligence type X. In one example, a rule may indicate that the corresponding third party is interested in vulnerabilities for particular applications, operating systems, hardware, or two or more of these.

[0045] The rules may include a destination, e.g., a system address for the particular third party. In some examples, the rules do not include a destination, and the system determines a destination using the particular third party to which the rule corresponds. The destination may be an internet protocol (IP) address, a uniform resource identifier (URI), or another appropriate type of identifier for the third party or a system, e.g., computer, used by the third party. When two rules for different third parties are the same, e.g., identify the same types of threat intelligence data, the two rules may include different destinations that correspond with the respective third party.

[0046] At time T**5**, the system sends data constructs, identified using the third party rules **108***a-b*, to the respective third parties **110***a-b*. For instance, the system sends the data construct m **106***a* to the Third Party A **110***a* and the data construct n **106***b* to both the Third Party A **110***a* and the Third Party B **110***b*. The data constructs provided to the third parties may be prioritized, e.g., using the third party rules **108***a-b* or other data, as described in more detail below.

[0047] One or more of the third parties **110***a-b* may use the data constructs **106***a-b* to adjust rules for an intrusion detection system (IDS), an intrusion prevention system (IPS), or both. For instance, The Third Party B **110***b* may receive courses of action from the system and update rules for an intrusion prevention system using the received courses of action. In one example, the Third Party B **110***b* may receive indicators of compromise, vulnerabilities, and exploits from the system, determine which of those threats are relevant to the Third Party B **110***b*, e.g., relevant to devices associated with the Third Party B **110***b*, and request courses of action from the system for the threats that are determined to be relevant to the Third Party B **110***b*. The Third Party B **110***b* may then use the received courses of action to update corresponding systems or policies, e.g., to prevent or reduce the likelihood of those threats occurring.

[0048] Some of the third parties **100***a-b* may analyze the data in the data constructs **106***a-b*, e.g., to determine historical perspectives, trends, or both. For instance, the Third Party A **110***a* may perform frequency analysis, determine clusters of threat relationships, or identify specific actors using the data constructs **106***a-b*.

[0049] In some examples, a third party may use received data constructs to determine potential tests for the third party's systems. For instance, the third party may test a potential vulnerability and determine how to manage the potential vulnerability.

[0050] In some implementations, a third party system may determine if one or more indicators, one or more observables, or a combination of the two, are currently occurring within the third party's infrastructure and, if so, perform an appropriate corrective action. In some implementations, a network resource manager, e.g., an IDS, may detect anomalous behavior, e.g., network behavior, and determine whether this behavior maps to indicators or observables received from the system. When the network resource manager determines that the behavior maps to data received from the system, the network resource manager may request additional information about the mapped indicators or observables from the system. For instance, the network resource manager may request and receive courses of action for the mapped indicators or observables, e.g., that the system identifies using relationships between the indicators or observables and the respective courses of action.

[0051] FIG. **2** is an example of an environment **200** in which an analysis system **202** receives data from unstructured data sources **204**, parses the data, creates structured data constructs from the parsed data, and provides the structured data to third party systems **208***a-c*. The analysis system **202** may also receive structured data from structured data sources **206**, potentially merge or relate the structured data constructs created from the unstructured data with the structured data and provide the data, e.g., structured data or data from a combination of structured and unstructured data sources, to the third party systems **208***a-c*.

[0052] The unstructured data sources **204** may include security advisories **210** received by the analysis system **202**, e.g., from unstructured data sources that distribute the security advisories via the Internet. The analysis system **202** may subscribe to one or more of the unstructured data sources **204** to receive some of the security advisories **210** from those sources. In some examples, the security advisories **210** may be publically available on a news website **212** or a government website **214**.

[0053] The analysis system **202** may use a web crawler to scan documents referenced by specified uniform resource identifiers (URIs), e.g., for one or more of the unstructured data sources **204**, to identify unstructured data. In some examples, the analysis system **202** may analyze a particular document, e.g., presented in a web browser or otherwise selected by a user, to determine unstructured data. For instance, a plugin in a web browser or another application may analyze content presented in the web browser and, upon identifying unstructured threat information, send that identified information to a server associated with the analysis system **202**, e.g., that executes at least part of the analysis system **202**.

[0054] The analysis system **202** may receive unstructured data, e.g., security threat information, from other types of news websites **212**, such as a news publisher or a blog. The security threat information may include information about a known threat actor, e.g., a hacker group, that actor's activities, security breaches at an organization, vulnerabilities identified

by one or more actors, e.g., a security organization, or any combination of two or more of these.

[0055] Some examples of content retrieved from the government websites **214** include security alerts, security warnings, or other information about security threats. The content from the government websites **214** may be publically available, e.g., on the Internet. The government websites **214** may include multiple different websites for the same government, for different governments, or both.

[0056] The unstructured data received from the unstructured data sources **204** does not include labels that explicitly identify intelligence types, e.g., threat actors, vulnerabilities, or exploits. As mentioned above, some examples of unstructured data include news websites, security alerts, blogs, and forum posts. The analysis system **202**, as described in more detail below, may analyze the unstructured data to identify data that corresponds to intelligence types and store that data in structured data constructs as described in more detail below. The structured data from the structured data sources **206** is formatted according to a schema, e.g., specific to a particular one of the structured data sources **206**. Some examples of structured data include an Extensible Markup Language (XML) feed, a feed with comma-separated values (CSV), or a tab delimited feed.

[0057] For instance, the structured data may include a header or title for threat information, may have a consistent way in which the data is stored in memory or presented, e.g., that is easily readable by computers, or both. The unstructured data does not have this consistency in data representation. Some examples of headers for structured data include "IP," "frequency," "percent of total," and "count" and the structured data would include values, e.g., text, numerical, or both, that correspond with each of these headers. In one example, unstructured data may include a blog post, which may include "section headers" that identify different portions of the post, e.g., "security threat A" and "security threat B," followed by the paragraphs of text describing those portions, but is not formatted in the same way as the structured data.

[0058] For instance, the structured data sources **206** may include multiple proprietary sources. Some examples of proprietary sources may include paid providers, to which the analysis system **202**, or another system or party associated with the analysis system **202**, provides compensation for data received by the analysis system **202**, and government sources. For instance, a government agency may make threat information publically available to other systems, including the analysis system **202**, upon request or may send the threat information to the analysis system **202** upon determining new threat information. The analysis system **202** may receive data from each of the proprietary sources in a proprietary format, e.g., in a data feed formatted specific to the particular proprietary source, or an open format.

[0059] The structured data sources **206** may include a threat exchange that provides data to the analysis system **202** for free. The threat exchange may require that the analysis system **202** provides the threat exchange with threat information in order to access threat information provided to the threat exchange by other sources. For instance, the threat exchange may receive data from one or more free content providers and provide that data to the analysis system **202**, e.g., in exchange for data received from the analysis system **202** or not.

[0060] The analysis system **202** may receive open source data from some of the structured data sources **206**. For example, the analysis system **202** may receive an Extensible Markup Language (XML) feed, a feed with comma-separated values (CSV), or a tab delimited feed.

[0061] One or more of the structured data sources **206** may overlap with other types of the structured data sources **206**. For instance, one of the proprietary sources may provide an XML feed, a CSV feed, a tab delimited feed, or free threat information. In some examples, some of the proprietary sources may make data available on the threat exchange, e.g., for free or for a fee. Some of the open source data may be made available on the threat exchange.

[0062] A parser **216** in the analysis system **202** receives the unstructured data from the unstructured data sources **204** and uses keywords **218**, patterns **220**, and optionally rules **222**, to create structured data constructs using data from the unstructured data, as described in more detail below. For example, the parser **216** may receive an unstructured dataset, and identify keywords in the dataset, e.g., that may be specific to particular intelligence types or particular types of attacks.

[0063] The parser **216** uses the identified keywords, and potentially the position of those keywords in the respective dataset, to determine text patterns from the patterns **220** that correspond to the respective dataset. For instance, the parser **216** may compare strings from the unstructured data, e.g., that include some of the identified keywords, with the patterns **220** to determine whether the string matches or is similar to one of the patterns **220**.

[0064] The parser **216** may use the determined patterns **220** for the unstructured data or a subset of the unstructured data to determine intelligence types for the respective subset of the unstructured data. For instance, the rules **222** may indicate that a particular pattern or group of patterns is associated with a particular one or more of the intelligence types **224**. The parser **216** may use any appropriate algorithm to determine which intelligence types **224** are associated with the unstructured data or a subset of the unstructured data. Some examples of the intelligence types **224** include observables, indicators of compromise (IOC), vulnerabilities, exploits, adversary tactics, techniques, and procedures (TTP), threat actors, threat campaigns, and courses of action (COA), described in more detail above.

[0065] The parser **216** may use machine learning, e.g., supervised or unsupervised, when processing unstructured data. For example, the parser **216** may use an OpenNLP library when processing the unstructured data.

[0066] For each of the intelligence types for a single dataset or subset of unstructured data, the parser **216** creates a data construct specific to the respective intelligence type and stores the data construct in a construct database **226**. For example, the parser **216** may create a data construct for an observable, a threat campaign, a COA, a vulnerability, or an exploit that includes an identifier for the data construct and the subset of data from the unstructured data that corresponds with the intelligence type. In some examples, the data constructs may include an identifier for the source of the data, e.g., the particular unstructured data source **204** or structured data source **206** from which the analysis system **202** received the data, a timestamp, e.g., for when the corresponding data was generated by or received from the respective structured data source, a source reputation, a severity, a severity reputation, or a combination of two or more of these.

[0067] Some of the data constructs may include links to other data constructs. For instance, a data construct for an IOC may include one or more links to data constructs for corresponding observables and a threat actor data construct

may include one or more links to threat campaigns or TTP used by the respective actor. Some of the data constructs, e.g., for IOC, threat actors, or TTP, may include a title or a name.

[0068] In some implementations, the parser **216** may create multiple data constructs for a particular intelligence type, e.g., from a single set of unstructured data. For instance, the parser **216** may determine a subset of a particular unstructured dataset that corresponds with observables and that different portions of that subset correspond with different instances of observables. One portion of the subset may include information about an HTTP request and another portion of the subset may include information about a particular file. In this example, the parser **216** creates a first observable data construct for information about the HTTP request and a second observable data construct for information about the particular file.

[0069] A de-duplication module **228**, included in the analysis system **202**, may determine relationships between datasets received from the unstructured data sources **204**, parsed data, e.g., that has not yet been stored in the construct database **226**, data received from the structured data sources **206**, data constructs stored in the construct database **226**, or any combination of two or more of these. For instance, the de-duplication module **228** may determine whether some of the data in two data constructs is related to the same threat, includes the same information about a threat, or includes information about related threats and how that information or those threats are related. The de-duplication module **228** may merge two different data constructs, relate two different data constructs, or discard one of two related data constructs.

[0070] For example, the de-duplication module **228** or the parser **216** may merge two sets or subsets of unstructured data when creating a data construct; discard some of the data, e.g., one of the subsets, and create a single data construct; or relate two separate data constructs, one of which may be newly created and the other previously created or both of which may be newly created. For instance, a data construct may include data subsets from both a first set of unstructured data and a second set of unstructured data, both of which correspond to the same intelligence type.

[0071] The parser **216** may work together with the de-duplication module **228** during a de-duplication process. For instance, the de-duplication module **228** may determine whether two data sets or subsets, e.g., of unstructured data, are related and how they are related, and the parser **216** may use information about the relationship when creating or updating data constructs stored in the construct database **226**.

[0072] When multiple datasets, or subsets, are related to the same threat and include the same information, e.g., all of the same information, the analysis system **202** may discard the older datasets and keep the newest dataset for a construct. Similarly, when multiple subsets are related to the same threat and include most of the same information, e.g., the newest dataset includes more information or updated information, the analysis system **202** may discard the older datasets and keep the newest dataset. In some examples, when an IOC subset or data construct includes the same observables, or links to the data constructs for the same observables, as another IOC subset or data construct, the analysis system **202** determines that the two are related to the same threat.

[0073] In some implementations, the analysis system **202** may discard a dataset when two datasets are related to the same threat but the information included in the two datasets varies by more than a threshold amount, e.g., by more than a predetermined percentage or by a predetermined quantity, such as a quantity of words, phrases, sentences or paragraphs. The discarded dataset may be the older dataset or the dataset from a less reputable source.

[0074] For instance, the system may use historical analysis of information from a certain source to determine a reputation for that source, e.g., historical analysis of how actionable is the information from a certain source. In some examples, the system may determine how much data from source A overlaps with a known well-reputed source. If there is a high overlap between data from source A and the well-reputed source, the system may assign a high reputation to the source A. In one example, an analyst may dynamically assign a reputation score to a source. The analysis system **202** may use any appropriate method to determine a source's reputation.

[0075] When a subset or a data construct includes a confidence score, the element with the highest confidence score may be kept, e.g., and the other data discarded. The confidence score may represent a probability that the data in the data construct is accurate or a probability that the data in the data construct is from a reputable source, e.g., as determined by the analysis system **202**. In some examples, the analysis system **202** may discard data with a lower confidence score when there is more than a twenty percent difference between the two confidence scores.

[0076] When multiple subsets of data are related to the same threat, e.g., determined using an identifier for the threat, the analysis system **202** may discard the subsets from the less reputable sources and keep the subsets from the more reputable sources. For instance, the analysis system **202** may create a data construct using the data from the more reputable sources. In some examples, if a cost, efficacy, affected software, confidence score, victim, time, leveraged TTPs, or attributed threat actors for a data construct or a subset, e.g., a COA, an exploits, or an incident data construct, are different than an existing data construct or another subset, the analysis system **202** may use the subset or data construct from the more reputable source, e.g., when each data construct includes a "source" element.

[0077] When multiple subsets of data are related to the same threat, but the content of the subsets is different, e.g., by more than a threshold amount, the analysis system **202** may create a data construct for each of the subsets and relate each of the created data constructs with each other. For instance, each data construct may include an identifier or one or more pointers that the analysis system **202** may use to determine the related data constructs, e.g., using a "related packages" element in the respective data construct.

[0078] In some examples, if a name in a subset of data is similar to a name in another subset of data, or in an existing data construct, e.g., when the names are used to identify the data, the analysis system **202** may relate a new data construct for the subset to a data construct for the other subset, or the existing data construct. The similarity in the name may allow for small errors, e.g., one character differences, and extensions of names, e.g., where "cat" and "cat_**1**" are considered matches.

[0079] The analysis system **202** uses third party rules **230** to determine when to send data constructs to the third parties **208***a-c*, what types of data constructs to send to the third parties **208***a-c*, and how to prioritize the data constructs, e.g., an order in which to present the data constructs to the third parties **208***a-c* or for the third parties **208***a-c* to present the information to a user. For instance, the analysis system **202**

may determine to send data constructs to the third party system A **208***a* when relevant data constructs are created by the analysis system **202**, to send data constructs to the third party system B **208***b* once each day, and to send data constructs to the third party system C **208***c* once each week. The third party rules **230** may include a particular time of day at which the analysis system **202** should send corresponding data constructs to the requesting third party systems **208***a-c*.

[0080] The analysis system **202**, or a module of the analysis system **202**, may use the third party rules **230** to determine which data constructs to send to each of the third party systems **208***a-c*. For instance, a first rule for the third party system A **208***a* may indicate that the third party A wants to receive threat information that is relevant to a particular organization or industry and a second rule for the third party system B **208***b* may indicate that the third party B wants to receive threat information for a particular operating system or a particular version of an operating system.

[0081] The third party rules **230** may include a corresponding destination for the requested data constructs. For instance, the first rule for the third party system A **208***a* may include an IP address for the third party system A **208***a*. In some examples, a rule may indicate that the analysis system **202** should send a first subset of data constructs to a first destination and a second subset of data constructs to a second destination. For instance, the third party system B **208***b* may include multiple servers and the analysis system **202** provides the first subset to a first server in the third party system B **208***b* and provides the second subset to a second server in the third party system B **208***b*.

[0082] A prioritization module **232**, included in the analysis system **202**, may use the third party rules **230** to determine a priority for each of the data constructs requested by a particular one of the third party systems **208***a-c*. For instance, after the analysis system **202** determines the data constructs to provide to the third party system C **208***c*, the prioritization module **232** determines the order in which the data constructs are provided to the third party system C **208***c* or presented on a user interface on the third party system C **208***c*, e.g., based on instructions provided by the analysis system **202** to the third party system C **208***c* for the presentation of the information included in the requested data constructs.

[0083] The prioritization module **232** may analyze how recently a threat, e.g., that is represented by a corresponding data construct, was used or identified, the severity of the threat, whether a government source has released a statement about the threat, or any combination of two or more of these when prioritizing the data constructs. For instance, the prioritization module **232** may assign a data construct for a newer threat a higher priority than another data construct for an older threat. The prioritization module **232** may assign a data construct for a more severe threat a higher priority than another data construct for a less severe threat, e.g., irrespective of the newness of either of the threats.

[0084] The prioritization module **232** may determine a severity of a threat using comments included in data from the unstructured data sources **204**, from the structured data sources **206**, or from other sources. For instance, the prioritization module **232** may assign a data construct a default, e.g., low or medium, severity unless the prioritization module **232** determines that comments that correspond with a threat represented by a data construct indicate that the threat should be associated with a different, e.g., higher, severity. The analysis system **202** may determine the comments or severity of a threat using unstructured data and details about a threat using structured data. In some examples, the analysis system **202** may determine the severity of a threat and the details of the threat from the unstructured data.

[0085] The third party systems **208***a-c* may use the received data constructs, and the priority in which the data constructs are received or identified for presentation, to take corresponding actions based on the threats identified in the data constructs. For instance, the third party system A **208***a* may be an automated system that determines whether the indicated threat is relevant to the third party A, e.g., whether or not any devices for the third party A may be affected by the indicated threat. Upon determining that a particular threat is relevant to the third party A. e.g., that a device may be compromised because of the indicated threat, the automated system may request a course of action (COA) from the analysis system **202** that corresponds with the particular threat. The third party system A **208***a* may analyze the data constructs in the order the data constructs are received from the analysis system **202**, e.g., according to the priority of the data constructs. The analysis system **202** may provide data constructs to third party systems **208***a-c* using an application programming interface (API), a feed, e.g., XML or rich site summary (RSS), or any other appropriate type of method or protocol.

[0086] In some examples, the analysis system **202** provides instructions to one of the third party systems **208***a-c*, e.g., the third party system B **208***b*, for the presentation of a user interface that includes information about the requested data constructs and the threats indicated by those data constructs. The user interface presents the threats according to the priority determined by the prioritization module **232**, e.g., with higher priority threats presented before lower priority threats. When the third party system receives additional data constructs from the analysis system **202**, the third party system may merge the most recently received data constructs, or information associated with these data constructs, with a current presentation of other previously received data constructs, or information associated with those data constructs. For instance, using the instructions received from the analysis system **202**, the third party system may determine that a first newly received data construct has a higher priority than a currently presented data construct and present the first newly received data construct above the currently presented data construct in a user interface. A second newly received data construct may have a lower priority than, and be presented below, the currently presented data construct.

[0087] In these examples, a user may view the user interface and determine which of the threats are relevant to the third party C and which threats require a corresponding action to be taken. For instance, a device operated by the user may request additional information about a particular threat, such as a COA, from the analysis system **202**, e.g., when an action should be taken for a threat. Upon receipt of the COA, the device operated by the user or another device, e.g., an automated device, may perform the action identified by the COA, e.g., to prevent or reduce the likelihood of the threat.

[0088] In some implementations, the prioritization module **232** may use learning to determine how to prioritize data constructs that will be sent to the third party systems **208***a-c*. For instance, the prioritization module **232** may receive requests for COAs from the third party system C **208***c*, determine that the types of threats associated with these COAs should have a higher priority than other types of threats for which the third party system C **208***c* requested information,

and update corresponding third party rules, or a prioritization module for the third party C, to indicate which types of threats should have a higher priority.

[0089] The prioritization module **232** may use a list of hardware manufacturers, a list of device models, a list of software developers, a list of software versions, a list of firewall rules, a list of intrusion prevention system rules, a list of malware rules, information source data, whether a course of action was performed, e.g., by the respective third party system, whether a patch was applied, e.g., by the respective third party system, or any combination of two or more of these when determining a data construct priority for one of the third party systems **208***a-c*. The particular third party system **208***a-c* may include any combination of prioritization rules, and rules for data constructs of interest, when defining the corresponding third party rules **230**.

[0090] In some examples, the prioritization module **232** may filter out data constructs for particular hardware manufacturers, device models, software developers, software versions or any combination of two or more of these, determined not to be relevant to a particular one of the third party systems **208***a-c*, e.g., and not send these data constructs to the particular third party system. The prioritization module **232** may filter out data constructs for particular data sources, e.g., when a third party rule indicates that data constructs from the particular data sources should not be sent to the corresponding third party system or when the third party system should only receive data constructs for data from a predetermined list of data sources that does not include any of the particular data sources.

[0091] In some implementations, the prioritization module **232** might not send any COAs to one or more of the third party systems **208***a-c*, e.g., unless one of the third party systems **208***a-c* requests a specific COA, at which time the analysis system **202** provides the respective third party system with the specific COA. In some examples, the prioritization module **232** determines whether a particular data construct relates to a malware signature, a firewall rule, or both, that has already been added to a corresponding firewall or anti-virus configuration. If so, the analysis system **202** might not send a corresponding data construct to the respective third party system. If not, the analysis system **202** sends the corresponding data structure to the respective third party system.

[0092] A network **234**, such as a local area network (LAN), wide area network (WAN), the Internet, or a combination thereof, connects the analysis system **202**, the unstructured data sources **204**, the structured data sources **206**, and the third party systems **208***a-c*. For instance, the analysis system **202** may receive structured data from the unstructured data sources **204** and the structured data sources **206** using the network **234** and provide data constructs to one or more of the third party systems **208***a-c* using the network **234**.

[0093] FIG. **3** is an example of a user interface **300** for presenting security threat information. The user interface **300** includes a priority line **302** that indicates a priority of security threats presented in the user interface **300**. For instance, instructions for the presentation of the user interface **300** may be provided to a particular third party system that include information about different security threats and corresponding priority information, e.g., specific to the particular third party system. If another third party system were to receive the same information about the same security threats, the other third party system may receive different priority information for those security threats, e.g., when the different priority information is specific to the other third party system.

[0094] In the example shown in FIG. **3**, a first document **304***a*, presented above a second document **304***b*, has a higher priority than the second document **304***b*, e.g., and was determined to be potentially more relevant to the particular third party system. In other examples, the second document **304***b* may have a higher priority than the first document **304***a*.

[0095] The first document **304***a* includes information about a security threat, such as a note indicating that "a new cyber group called Group A recently brought down Website B, and other large websites" such as Website D and Website E. The first document **304***a* may further indicate that "the main objective of this attack is to send a message to all organizations about their mission. Group A is purposefully disrupting websites to send a message to Organization C. The cyber attackers are employing the attack type called DDoS, Distributed Denial of Service, and using this method to crash websites by sending high volumes of traffic."

[0096] The formatting of the first document **304***a* may correspond with the format of the unstructured data received by the analysis system. In some examples, the analysis system determines the formatting of the first document **304***a* using rules or style sheets, or another appropriate method.

[0097] A user of the particular third party system may view the first document **304***a* and determine whether the first document **304***a* includes information associated with the particular third party system. For instance, the user may determine whether the attack may be or is directed toward the particular third party system or another system operated by or associated with the third party, such as an organization with which the third party works or has a business relationship.

[0098] The user may select a more information option **306** in the user interface **300** to view additional details about the security threat or a list of options for the security threat. The list of options may include an option to view the intelligence types associated with the first document **304***a*. Upon selection of an option to view the intelligence types, the particular third party system may receive instructions from the analysis system for presentation of an intelligence types view **308***a* for the first document **304***a*. The intelligence types view **308***a* may indicate that the threat actors identified in the first document **304***a* include the Group A, the threat campaigns include the Website B, the Organization C, and the Websites D and E, e.g., the latter two as the other large websites mentioned in the note for the first document **304***a*. The intelligence types view **308***a* may include DDoS as an indicator of compromise for the first document **304***a*.

[0099] The user may select an option to receive a course of action for the first document, e.g., in the menu presented upon selection of the more information option **306** or in the intelligence types view **308***a*. Upon selection of the option to receive a course of action, the particular third party system may request, and receive, a corresponding course of action from the analysis system. For instance, the request may include an identifier for the first document **304***a*, or the threat campaigns or the threat actors identified in the first document **304***a*, or a combination of the two, and the analysis system may use this information to determine a corresponding course of action and provide information about the corresponding course of action to the particular third party system.

[0100] The particular third party system may present information about the course of action to the user, e.g., in the user interface **300** or another user interface. In some examples, the

particular third party system may automatically perform one or more actions in response to receipt of the requested course of action, e.g., without user input. The actions performed by the particular third party system may reduce or eliminate the likelihood that the particular third party system, or another system associated with the particular third party, may be compromised by the same threat as the security threat identified in the first document **304***a*.

[0101] After viewing the first document **304***a*, the user may select a complete option to indicate that the user viewed the first document **304***a*, and view information about the second document **304***b*. For instance, the user may view an intelligence types view **308***b* for the second document **304***b*.

[0102] FIG. **4** is an example of a user interface **400** that includes threat trend information. For instance, the analysis system may provide a third party system with security threat information and the third party system may use the received security threat information to generate the user interface **400**. In some implementations, the analysis system may provide the third party system with instructions for the presentation of the user interface **400**.

[0103] The user interface **400** includes a threat actors graph **402** with information about a quantity of threat actors over a particular period of time. The particular period of time may be the most recent year, month, week, or another time range, e.g., determined by a user of the user interface **400**.

[0104] The quantity of threat actors may be specific to a particular industry, e.g., retail, a particular organization or entity, e.g., Company A, or for all known threat actors during the particular period of time. The quantity of threat actors may include only unique instances of threat actors or may include multiple instances of the same threat actor. For example, when a particular group attacks two different organizations using different techniques, the quantity of threat actors may count this particular group once, e.g., only unique instances of threat actors, or twice, e.g., for each attack performed by the particular group. In the latter example, the threat actors graph **402** may be a threat attacks graph that shows trends in a quantity of attacks over time.

[0105] Threat actors details **404** may include details about some of or all of the threat actors identified in the threat actors graph **402**. For instance, for each threat actor included in the threat actors graph **402**, the threat actors details **404** may include the number of attacks performed by that actor, e.g., during the predetermined period of time or at any time, an attack percentage for the respective actor, e.g., a ratio of the quantity of attacks associated with the respective actor to the quantity of attacks that have been carried out by all other threat actors, and the number of threat campaigns performed by the respective threat actor, e.g., during the predetermined period of time or at any time. In some examples, a sum of all the attack percentages may be greater than 100%, e.g., when multiple actors are involved in the same attack. In some examples, a sum of all the attack percentages, e.g., by attacker A, attacker B, and attacker C, may be 100%, e.g., when only a single attacker is associated with each attack. In some implementations, a sum of all the attack percentages may be less than 100%, e.g., when some attacks are not attributed to any attacker.

[0106] The user interface **400** may include additional information, such as a news feed **406** with information about security threats, potential security threats, solutions to security threats, or other relevant information. In some examples, the user interface **400** may include information about trends for particular types of threats, such as DDoS, particular types of threat campaigns, or other types of trend information determined using the data constructs.

[0107] FIG. **5** is a flow diagram of a process **500** for storing a record, created from unstructured data, in a database. For example, the process **500** can be used by the analysis system **202** from the environment **200**.

[0108] The process receives unstructured data (**502**). For instance, the analysis system retrieves the unstructured data from a website, e.g., that contains a security advisory or other security information.

[0109] The process determines whether the unstructured data includes information about a same threat as other data (**504**). For instance, the analysis system may determine whether a subset or all of the unstructured data is the same as other unstructured or structured data received by the analysis system.

[0110] If the process determines that the unstructured data includes information about the same threat as other data, the process discards the unstructured data (**506**). For example, the analysis system removes the unstructured data from memory, e.g., permanent or short term memory, and takes no further action with respect to the unstructured data.

[0111] If the process determines that the unstructured data does not include information about the same threat as other data, the process identifies one or more keywords in the unstructured data (**508**). For instance, the analysis system determines whether one or more of the keywords or key phrases, shown in Table 1, or variations of these keywords or key phrases, e.g., a different grammatical number or tense, are included in the unstructured data.

TABLE 1

Example Keywords and Key Phrases

action	course	found	platform	statement
advisory	CVE ID	found in	prone	strategies
affect	CVSS	handling	prone to	systems
affected	denial of service	Header: Details	recommend	targets
affecting	described	Header: Summary	release	trigger
arbitrary	details	how to resolve	remediation	update
attacker	disable	impact	reported in	upgrade
attacks	discovered	improper	result	vendor
available	execute	information	result in	versions
cause	exploitation	mitigation	software	vulnerable
code	exploits	patch	solution	What can you do

[0112] The process determines one or more patterns included in the unstructured data using the keywords (**510**). For example, the analysis system determines whether one or more combinations of keywords, shown in Tables 2 and 3, occur in the unstructured data or a particular subset, e.g., sentence, paragraph, page, etc., of the unstructured data. In the examples shown in Table 3, the values in brackets "<xxx>" can be a variety of different text strings. For instance, both "threat actor A exploits" and "group Z exploits" match the pattern "<TA> exploits."

TABLE 2

Examples Patterns A		
Patterns to identify vulnerabilities	Patterns to identify COAs	Patterns to identify IOCs
vulnerabilities described advisory affect versions	solution	impact
vendor statement	update available	vulnerabilities results
vendor information versions	recommended course action	vulnerabilities cause
vulnerability found	disable	vulnerability prone
prone vulnerability	mitigation strategies	attacker execute arbitrary code
systems affected	recommendations	exploitation vulnerabilities result
affected platform	patch vulnerable software	improper handling cause resulting
attacks targets	remediation	denial of service
vulnerable systems	upgrade patched release	

TABLE 3

Examples Patterns B				
Patterns to identify Indicators of compromise (IOC)	Patterns to identify Threat Actors (TA)	Patterns to identify Exploit Target/ Vulnerabilities (ET)	Patterns to identify Adversary tactics, techniques, and procedures (TTP)	Patterns to identify Courses of Action (COA)
<IoC> Vulnerability	<TA> causes	Vulnerability <ET>	<IOC> reported in . . . <TTPs>	Solution
Cause <IoC>	<TA> exploits	CVE ID <ET>	Header: Summary <TTPs>	Update available
Prone to <IoC>	<TA> triggers	CVSS <ET>	Systems affected	Recommended course action
Result in <IoC>		<IOC> resulting/found in/discovered/affecting <exploit target>	Affected platform	Disable
Trigger <IoC>		Vulnerability found <ET>	Attacks targets	Mitigation strategies
		Prone vulnerability	Vulnerable systems	Recommendations
		Details <ET>	<attacker> <opt modal> <action verb>	Patch vulnerable software
			Header: Details <TTPs>	Remediation
				Upgrade patched release
				How to Resolve <COA>
				What can you do <COA>

[0113] The process determines one or more rules using the determined patterns (**512**). The analysis system may determine that each pattern corresponds with a rule identifying a particular intelligence type or types for the respective pattern. For instance, a rule may associate the pattern "vendor statement" with the intelligence type "vulnerabilities."

[0114] The process identifies one or more intelligence types that correspond with the unstructured data (**514**). The analysis system may determine that a particular subset of the unstructured data corresponds with one of the intelligence types, using the patterns or the rules, or that all of the unstructured data corresponds with a particular one or more of the intelligence types.

[0115] The process associates, for each of the identified intelligence types, a data subset from the unstructured data with the respective intelligence type (**516**). In some examples, the analysis system associates all of the unstructured data with the respective intelligence type. The association may include the creation of a data construct specific to the respective intelligence type that stores the information from the data subset.

[0116] The process stores, for each of the identified intelligence types, at least one new record, specific to the respective intelligence type and that each include information from the respective data subset, in a database (**518**). For example, the analysis system stores a data construct in a construct database for each of the identified intelligence types, e.g., and for each dataset or subset that corresponds to the identified intelligence type. The analysis system may store information about a particular subset of the unstructured data in multiple data constructs, e.g., when that particular subset corresponds with multiple different intelligence types.

[0117] The order of steps in the process **500** described above is illustrative only, and storing the record, created from the unstructured data, in the database can be performed in different orders. For example, the analysis system may associate the data subset with the respective intelligence type and then determine whether the data subset includes information about the same threat as other data or is otherwise related to the same threat as the other data.

[0118] In some implementations, the process **500** can include additional steps, fewer steps, or some of the steps can be divided into multiple steps. For example, the analysis system may perform the process **600** instead of steps **508** through **512**. In some examples, the process does not include steps **506**, **512**, **518**, or any two or more of these steps.

[0119] In some implementations, the process may determine rules for a third party, determine data constructs that correspond with those rules, and provide the data constructs to the third party, e.g., according to a priority. For instance, the process may determine one or more rules for a third party that each indicate that the third party should receive data associated with particular types of potential security threats and priority information for the data. The particular types of potential security threats may be specific to a particular software application, hardware device, or both. For example, a rule may indicate that data constructs for a particular operating system, for specific software for a particular operating system, or for particular software for a certain type of hardware, e.g., brand, model, or instruction set, should be sent to the third party.

[0120] The process may determine, for each of the potential security threats indicated in the rules, a group of the subsets that include information associated with the respective potential security threat. For instance, the analysis system uses the rules to determine which data constructs correspond with the rules for a particular third party, e.g., and have not already been sent to that particular third party. The analysis system may use a frequency of updates, specified by the rules, to determine when to send data constructs to the third party, e.g., and determine the data constructs that have been created or updated since the last time data constructs were sent to the particular third party.

[0121] The process may assign, for each subset in each of the groups, a priority to the respective subset using the priority information. For example, a priority module uses the priority information, which may directly or indirectly correspond with the potential security threats, to determine a priority for the data constructs, e.g., which data constructs are more likely to be of greater importance to the third party than the other data constructs. In some examples, each of the potential security threats in the rules may include corresponding priority information for the priority module to use when determining the priorities.

[0122] In some examples, the priority module determines priority information using machine learning. For instance, the priority module may determine that the third party requests additional information for certain types of potential security threats, and not others, and rank data constructs for those certain types of potential security threats with a higher priority than data constructs for the other types of potential security threats.

[0123] The process may provide the determined subsets to the third party using the respective priorities. For example, the analysis system provides each of the data constructs to a system of the third party, e.g., with each data construct in a separate message or with all of the data constructs in a message feed.

[0124] FIG. **6** is a flow diagram of a process **600** for determining intelligence types for unstructured data in a document. For example, the process **600** can be used by the analysis system **202** from the environment **200**.

[0125] The process receives a document (**602**). For instance, the analysis system receives the document from an unstructured data source, e.g., a webpage hosted on behalf of a particular organization, and stores the document in a memory. The parser may retrieve the document from the memory.

[0126] The process extracts text from the document (**604**). For example, the parser identifies the text in the document, e.g., separate from images and other content included in the document.

[0127] The process divides the text into sentences (**606**). For instance, the parser determines the logical segments in the text, such as words, sentences, and paragraphs.

[0128] The process determines patterns for the sentences (**608**). For example, using the example patterns in Tables 2 and 3 above, the parser determines whether any of the sentences include a pattern or text associated with a pattern, e.g., synonyms for words included in a pattern.

[0129] The process determines, for each of the sentences, a similarity score for each of the patterns (**610**). For instance, the parser determines that a first similarity score for a particular sentence, e.g., "The vulnerabilities described in this advisory affect Application A versions 0.9.6j and 0.9.6k," and a particular pattern, e.g., "mitigation strategies," is low, e.g., zero, and a second similarity score for another pattern, e.g., "vulnerabilities described advisory affect versions" is high, e.g., one. The parser may determine other similarity scores for the particular sentence for different patterns, e.g., "affected platform" or "vulnerable systems," that are between the first similarity score and the second similarity score.

[0130] The parser, or another system, may use any appropriate method to determine the similarity scores. For instance, a similarity score may represent the percentage of words included in a pattern that are also in the sentence or logical segment, e.g., as a value between zero and one. The similarity score may represent the percentage of words included in a pattern that are included in the sentence or have synonyms included in the sentence. In some examples, the parser may use pointwise mutual information (PMI) to determine the similarity scores.

[0131] The process determines, for each of the sentences, a highest similarity score (**612**). For example, the parser determines that the second similarity score for the particular sentence is the highest similarity score for that sentence.

[0132] The process determines, for each of the sentences, the respective pattern that corresponds to the highest similar-

ity score (**614**). For instance, the parser determines that the respective pattern for the particular sentence is "vulnerabilities described advisory affect versions."

[0133] The process determines, for each of the sentences, an intelligence type for the respective pattern that corresponds to the highest similarity score (**616**). For example, the parser determines that the particular sentence corresponds with a vulnerability using the determined pattern. The parser may make the determination using a rule or other information that correlate the respective pattern with the intelligence type.

[0134] The parser may then create a data construct, e.g., specific to vulnerabilities, for the particular sentence. The data construct may include information representing the sentence "The vulnerabilities described in this advisory affect Application A versions 0.9.6j and 0.9.6k" and additional information, such as the source of the unstructured data, a timestamp when the unstructured data was created or received, a source reputation, a severity, a severity reputation, or any two or more of these.

[0135] Table 4 below shows some additional examples of sentences and corresponding patterns. In some examples, a sentence may be associated with a particular intelligence type based on a relationship between the sentence and another sentence, e.g., the sentence may be selected on the basis of recommendation by maximum number patterns. For instance, the parser may determine that a first sentence, "The impact from vulnerabilities can expose a Denial-of-Service (DoS) condition," in a particular document or paragraph is an indicator of compromise, using the pattern "denial of service." The parser may then determine that a second sentence, "The vulnerabilities described in this advisory affect Application B versions 4.0 and current," in the same particular document or paragraph, e.g., adjacent to the first sentence, should also be an indicator of compromise based on a relationship between the two sentences, e.g., when both sentences relate to the same vulnerability.

[0136] Table 4 shows an example of a sentence, "The vulnerabilities described in this advisory affect Application B versions 4.0 and current," which may be associated with multiple intelligence types, e.g., both vulnerability and indicator of compromise. In this example, the parser creates two constructs, one for a vulnerability and one for an indicator of compromise, that both include information about the sentence, e.g., ASCII code representing the text in the sentence.

TABLE 4

Example mapping of sentences to patterns and Intelligence Types

Sentence	Pattern	Intelligence Type
The vulnerabilities described in this advisory affect Application B versions 4.0 and current.	vulnerabilities described advisory affect versions	Vulnerability
	Sentence selected on the basis of recommendation by maximum number patterns	Indicator of Compromise (IOC)
Vendor Information Application C version 3.0 and earlier Application C version 3.2.11 and earlier Application C 2.x series is not vulnerable.	vendor information versions	Vulnerability
There are no known workarounds for the vulnerabilities.	Sentence selected on the basis of recommendation by maximum number patterns	Vulnerability
The impact from vulnerabilities can expose a Denial-of-Service (DoS) condition.	denial of service	Indicator of Compromise (IOC)
An attacker can exploit this issue to cause the process Z to hang, denying service to legitimate users.	denial of service	Indicator of Compromise (IOC)
Impact Recursive nameservers are vulnerable to this defect.	impact	Indicator of Compromise (IOC)
Remediation Patch the affected software with the patches supplied by the vendor or update to version 0.9.6j.	remediation	Course of Action (COA)
Disable the Workstation Service Depending on site requirements, you may wish to disable the Workstation Service as described in document M.	disable	Course of Action (COA)
Mitigating Strategies: Block external access at the network boundary, as much as possible.	mitigation strategies	Course of Action (COA)

[0137] In some implementations, the process **600** can include additional steps, fewer steps, or some of the steps can be divided into multiple steps. For example, the process might not receive a document. In some examples, the parser receives unstructured data from the unstructured data sources.

[0138] Embodiments of the subject matter and the functional operations described in this specification can be implemented in digital electronic circuitry, in tangibly-embodied computer software or firmware, in computer hardware, including the structures disclosed in this specification and their structural equivalents, or in combinations of one or more of them. Embodiments of the subject matter described in this specification can be implemented as one or more computer programs, i.e., one or more modules of computer program instructions encoded on a tangible non-transitory program carrier for execution by, or to control the operation of, data processing apparatus. Alternatively or in addition, the program instructions can be encoded on an artificially-generated propagated signal, e.g., a machine-generated electrical, optical, or electromagnetic signal, that is generated to encode information for transmission to suitable receiver apparatus for execution by a data processing apparatus. The computer storage medium can be a machine-readable storage device, a machine-readable storage substrate, a random or serial access memory device, or a combination of one or more of them.

[0139] The term "data processing apparatus" refers to data processing hardware and encompasses all kinds of apparatus, devices, and machines for processing data, including by way of example a programmable processor, a computer, or multiple processors or computers. The apparatus can also be or further include special purpose logic circuitry, e.g., an FPGA (field programmable gate array) or an ASIC (application-specific integrated circuit). The apparatus can optionally include, in addition to hardware, code that creates an execution environment for computer programs, e.g., code that constitutes processor firmware, a protocol stack, a database management system, an operating system, or a combination of one or more of them.

[0140] A computer program, which may also be referred to or described as a program, software, a software application, a module, a software module, a script, or code, can be written in any form of programming language, including compiled or interpreted languages, or declarative or procedural languages, and it can be deployed in any form, including as a stand-alone program or as a module, component, subroutine, or other unit suitable for use in a computing environment. A computer program may, but need not, correspond to a file in a file system. A program can be stored in a portion of a file that holds other programs or data, e.g., one or more scripts stored in a markup language document, in a single file dedicated to the program in question, or in multiple coordinated files, e.g., files that store one or more modules, sub-programs, or portions of code. A computer program can be deployed to be executed on one computer or on multiple computers that are located at one site or distributed across multiple sites and interconnected by a communication network.

[0141] The processes and logic flows described in this specification can be performed by one or more programmable computers executing one or more computer programs to perform functions by operating on input data and generating output. The processes and logic flows can also be performed by, and apparatus can also be implemented as, special purpose logic circuitry, e.g., an FPGA (field programmable gate array) or an ASIC (application-specific integrated circuit).

[0142] Computers suitable for the execution of a computer program include, by way of example, general or special purpose microprocessors or both, or any other kind of central processing unit. Generally, a central processing unit will receive instructions and data from a read-only memory or a random access memory or both. The essential elements of a computer are a central processing unit for performing or executing instructions and one or more memory devices for storing instructions and data. Generally, a computer will also include, or be operatively coupled to receive data from or transfer data to, or both, one or more mass storage devices for storing data, e.g., magnetic, magneto-optical disks, or optical disks. However, a computer need not have such devices. Moreover, a computer can be embedded in another device, e.g., a mobile telephone, a personal digital assistant (PDA), a mobile audio or video player, a game console, a Global Positioning System (GPS) receiver, or a portable storage device, e.g., a universal serial bus (USB) flash drive, to name just a few.

[0143] Computer-readable media suitable for storing computer program instructions and data include all forms of non-volatile memory, media and memory devices, including by way of example semiconductor memory devices, e.g., EPROM, EEPROM, and flash memory devices; magnetic disks, e.g., internal hard disks or removable disks; magneto-optical disks; and CD-ROM and DVD-ROM disks. The processor and the memory can be supplemented by, or incorporated in, special purpose logic circuitry.

[0144] To provide for interaction with a user, embodiments of the subject matter described in this specification can be implemented on a computer having a display device, e.g., a CRT (cathode ray tube) or LCD (liquid crystal display) monitor, for displaying information to the user and a keyboard and a pointing device, e.g., a mouse or a trackball, by which the user can provide input to the computer. Other kinds of devices can be used to provide for interaction with a user as well; for example, feedback provided to the user can be any form of sensory feedback, e.g., visual feedback, auditory feedback, or tactile feedback; and input from the user can be received in any form, including acoustic, speech, or tactile input. In addition, a computer can interact with a user by sending documents to and receiving documents from a device that is used by the user; for example, by sending web pages to a web browser on a user's device in response to requests received from the web browser.

[0145] Embodiments of the subject matter described in this specification can be implemented in a computing system that includes a back-end component, e.g., as a data server, or that includes a middleware component, e.g., an application server, or that includes a front-end component, e.g., a client computer having a graphical user interface or a Web browser through which a user can interact with an implementation of the subject matter described in this specification, or any combination of one or more such back-end, middleware, or front-end components. The components of the system can be interconnected by any form or medium of digital data communication, e.g., a communication network. Examples of communication networks include a local area network (LAN) and a wide area network (WAN), e.g., the Internet.

[0146] The computing system can include clients and servers. A client and server are generally remote from each other and typically interact through a communication network. The relationship of client and server arises by virtue of computer programs running on the respective computers and having a

client-server relationship to each other. In some embodiments, a server transmits data, e.g., an HTML page, to a user device, e.g., for purposes of displaying data to and receiving user input from a user interacting with the user device, which acts as a client. Data generated at the user device, e.g., a result of the user interaction, can be received from the user device at the server.

[0147] An example of one such type of computer is shown in FIG. **7**, which shows a schematic diagram of a generic computer system **700**. The system **700** can be used for the operations described in association with any of the computer-implement methods described previously, according to one implementation. The system **700** includes a processor **710**, a memory **720**, a storage device **730**, and an input/output device **740**. Each of the components **710**, **720**, **730**, and **740** are interconnected using a system bus **750**. The processor **710** is capable of processing instructions for execution within the system **700**. In one implementation, the processor **710** is a single-threaded processor. In another implementation, the processor **710** is a multi-threaded processor. The processor **710** is capable of processing instructions stored in the memory **720** or on the storage device **730** to display graphical information for a user interface on the input/output device **740**.

[0148] The memory **720** stores information within the system **700**. In one implementation, the memory **720** is a computer-readable medium. In one implementation, the memory **720** is a volatile memory unit. In another implementation, the memory **720** is a non-volatile memory unit.

[0149] The storage device **730** is capable of providing mass storage for the system **700**. In one implementation, the storage device **730** is a computer-readable medium. In various different implementations, the storage device **730** may be a floppy disk device, a hard disk device, an optical disk device, or a tape device.

[0150] The input/output device **740** provides input/output operations for the system **700**. In one implementation, the input/output device **740** includes a keyboard and/or pointing device. In another implementation, the input/output device **740** includes a display unit for displaying graphical user interfaces.

[0151] While this specification contains many specific implementation details, these should not be construed as limitations on the scope of what may be claimed, but rather as descriptions of features that may be specific to particular embodiments. Certain features that are described in this specification in the context of separate embodiments can also be implemented in combination in a single embodiment. Conversely, various features that are described in the context of a single embodiment can also be implemented in multiple embodiments separately or in any suitable subcombination. Moreover, although features may be described above as acting in certain combinations and even initially claimed as such, one or more features from a claimed combination can in some cases be excised from the combination, and the claimed combination may be directed to a subcombination or variation of a subcombination.

[0152] Similarly, while operations are depicted in the drawings in a particular order, this should not be understood as requiring that such operations be performed in the particular order shown or in sequential order, or that all illustrated operations be performed, to achieve desirable results. In certain circumstances, multitasking and parallel processing may be advantageous. Moreover, the separation of various system modules and components in the embodiments described above should not be understood as requiring such separation in all embodiments, and it should be understood that the described program components and systems can generally be integrated together in a single software product or packaged into multiple software products.

[0153] Particular embodiments of the subject matter have been described. Other embodiments are within the scope of the following claims. For example, the actions recited in the claims can be performed in a different order and still achieve desirable results. As one example, the processes depicted in the accompanying figures do not necessarily require the particular order shown, or sequential order, to achieve desirable results. In some cases, multitasking and parallel processing may be advantageous.

What is claimed is:

1. A computer-implemented method comprising:

receiving unstructured textual data;

identifying one or more keywords in the unstructured textual data;

determining one or more patterns included in the unstructured textual data using the identified keywords;

identifying one or more intelligence types that correspond with the unstructured textual data using the determined patterns; and

associating, for each of the identified intelligence types, a data subset from the unstructured textual data with the respective intelligence type.

2. The method of claim **1**, wherein associating, for each of the identified intelligence types, the data subset from the unstructured textual data with the respective intelligence type comprises storing, for each of the identified intelligence types, at least one new record, in a database, specific to the respective intelligence type that each comprises information from the respective data subset.

3. The method of claim **1**, wherein the unstructured comprises an advisory.

4. The method of claim **1**, wherein receiving the unstructured textual data comprises receiving the unstructured textual data from a government or a security source.

5. The method of claim **1**, wherein identifying the intelligence types that correspond with the unstructured textual data using the determined patterns comprises:

determining one or more rules using the determined patterns; and

identifying the intelligence types that correspond with the unstructured textual data using the determined rules.

6. The method of claim **1**, comprising:

determining a rule for a third party that indicates that the third party should receive data associated with a particular intelligence type;

determining that at least one of the identified intelligence types comprises the particular intelligence type; and

providing, for each of the at least one of the identified intelligence types, the respective data subset to the third party.

7. The method of claim **6**, wherein providing, for each of the at least one of the identified intelligence types, the respective data subset to the third party comprises providing, for each of the at least one of the identified intelligence types, instructions to the third party for presentation of the respective data subset.

8. The method of claim **1**, comprising:

receiving a plurality of second unstructured textual datasets that each include information about a potential security threat;

determining that a first subset of the second unstructured textual datasets comprises information about the same threat as a second, different subset of the second unstructured textual datasets;

discarding the first subset; and

identifying, for each second unstructured textual dataset in the second subset, one or more keywords in the respective second unstructured textual dataset.

9. A system comprising:

one or more computers and one or more storage devices storing instructions that are operable, when executed by the one or more computers, to cause the one or more computers to perform operations comprising:

receiving unstructured textual data;

parsing the unstructured textual data into a plurality of sections including a particular section;

identifying one or more keywords in the particular section of the unstructured textual data;

determining one or more patterns that match the particular section using the identified keywords;

identifying one or more intelligence types that correspond with the particular section using the determined patterns; and

associating, for each of the identified intelligence types, a data subset from the unstructured textual data, associated with the particular section, with the respective intelligence type.

10. The system of claim **9**, wherein each of the plurality of sections comprises a sentence.

11. The system of claim **9**, wherein each of the plurality of sections comprises a paragraph.

12. The system of claim **9**, wherein the data subset comprises data representing the particular section.

13. The system of claim **9**, wherein associating, for each of the identified intelligence types, the data subset from the unstructured textual data with the respective intelligence type comprises storing, for each of the identified intelligence types, at least one new record, in a database, specific to the respective intelligence type that each comprises information from the respective data subset.

14. The system of claim **9**, the operations comprising:

determining a rule for a third party that indicates that the third party should receive data associated with a particular intelligence type;

determining that at least one of the identified intelligence types comprises the particular intelligence type; and

providing, for each of the at least one of the identified intelligence types, the respective data subset to the third party.

15. The system of claim **14**, wherein providing, for each of the at least one of the identified intelligence types, the respective data subset to the third party comprises providing, for each of the at least one of the identified intelligence types, instructions to the third party for presentation of the respective data subset.

16. A computer storage medium encoded with instructions that, when executed by a user device, cause the user device to perform operations comprising:

receiving unstructured textual data;

identifying one or more keywords in the unstructured textual data;

determining one or more patterns included in the unstructured textual data using the identified keywords;

identifying one or more intelligence types that correspond with the unstructured textual data using the determined patterns;

associating, for each of the identified intelligence types, a data subset from the unstructured textual data with the respective intelligence type;

determining a rule for a third party that indicates that the third party should receive data associated with a particular intelligence type;

determining that at least one of the identified intelligence types comprises the particular intelligence type; and

providing, for each of the at least one of the identified intelligence types, the respective data subset to the third party.

17. The computer storage medium of claim **16**, wherein providing, for each of the at least one of the identified intelligence types, the respective data subset to the third party comprises providing, for each of the at least one of the identified intelligence types, instructions to the third party for presentation of the respective data subset.

18. The computer storage medium of claim **16**, the operations comprising:

receiving a plurality of second unstructured textual datasets that each include information about a potential security threat;

determining that a first subset of the second unstructured textual datasets comprises information about the same threat as a second, different subset of the second unstructured textual datasets;

discarding the first subset; and

identifying, for each second unstructured textual dataset in the second subset, one or more keywords in the respective second unstructured textual dataset.

19. The computer storage medium of claim **16**, wherein associating, for each of the identified intelligence types, the data subset from the unstructured textual data with the respective intelligence type comprises storing, for each of the identified intelligence types, at least one new record, in a database, specific to the respective intelligence type that each comprises information from the respective data subset.

20. The computer storage medium of claim **16**, the operations comprising:

parsing the unstructured textual data into a plurality of sections including a particular section;

wherein identifying one or more keywords in the unstructured textual data comprises identifying one or more keywords in the particular section of the unstructured textual data.

* * * * *